Second Edition

Javvin

Network Protocols Handbook

TCP/IP
ATM
Ethernet
LAN
Frame Relay
WAN
SS7/C7
VOIP
WLAN
Security
SAN
IETF
VLAN
ISO
MAN
IEEE
VPN
IBM
ITU-T
Cisco
Apple
ANSI
Novell
Microsoft

Javvin Technologies, Inc.

Network Protocols Handbook

2nd Edition.

Copyright © 2004 - 2005 Javvin Technologies Inc. All rights reserved.

13485 Old Oak Road
Saratoga CA 95070 USA
408-872-3881
handbook@javvin.com

All rights reserved. No part of this book may be reproduced or transmitted in any form or by any means electronically or mechanically.

Warning and Disclaimer

This book is designed to provied information about the current network communication protocols. Best effort has been made to make this book as complete and accurate as possible, but no warranty or fitness is implied. The infomation is provided on an “as is” basis. The author, publisher and distributor shall not have liability nor responsibility to anyone or group with respect to any loss arising from the information contained in this book and associated materials.

Table of Contents

Network Communication Architecture and Protocols••••••••••••••••••••••••••••••1

OSI Network Architecture 7 Layers Model••2

TCP/IP Four Layers Archiitecture Model••5

Other Network Architecture Models: IBM SNA••••••••••••••••••••••••••••••••••7

Network Protocols: Definition and Overview••••••••••••••••••••••••••••••••••••9

Protocols Guide••••••••••••••••••••••••••••••••••••••

SLP: Service Location Protocol••••••

EGP: Exterior Gateway Protocol••••••

CR-LDP: Constraint-based LDP••••••

**Other Security Protocols••••••

COPS: Common Open Policy Service••••••

BACP: PPP Banwidth Allocation Control Protocol (BACP)••••••••••••••••••••••••178

BCP: PPP Briding Control Protocol••179

EAP: PPP Extensible Authentication Protocol••180

CHAP: Challenge Handshake Authentication Protocol••••••••••••••••••••••••••••••••181

LCP: PPP Link Control Protocol••182

MPPP: MultiLink Point to Point Protocol (MultiPPP)••••••••••••••••••••••••••••••••183

PPP NCP: Point to Point Protocol Network Control Protocols••••••••••••••••••••••184

PAP: Password Authentication Protocol••185

PPPoA: PPP over ATM AAL5••186

PPPoE: PPP over Ethernet••187

**Other WAN Protocols••••••••••••••••••••••••••••••••••••••

GARP: Generic Attribute Registration Protocol••207

GMRP: GARP Multicast Registration Protocol••208

GVRP: GARP VLAN Registration Protocol••209

**Wilress LAN Protocols••••••••••••••••••••••••••••••••••••••

ISO Protocols in OSI 7 Layers Model••240

**Application Layer••••••••••••••••••••••••••••••••••••••

IGRP: Interior Gateway Routing Protocol••••••

SS7/C7 Protocols: Signalling System #7 for Telephony ••••••••••••••••304

BISUP: Broadband ISDN User Part••306

DUP: Data User Part••307

ISUP: ISDN User Part••308

MAP: Mobile Application Part••310

MTP2 and MTP3: Message Transfer Part level 2 and level 3••••••••••••••••••••••••••••••••••••312

SCCP: Signalling Connection Control Part of SS7••314

TCAP: Transaction Capabilities Application Part••315

TUP: Telephone User Part••317

Other Protocols••318

Microsoft CIFS: Common Internet File System••319

Microsoft SOAP: Simple Object Access Protocol••320

Xerox IDP: Internet Datagram Protocol••321

Toshiba FANP: Flow Attribute Notification Protocol••322

Network Protocols Dictionary: From A to Z and 0 to 9••••••••••••••••••••••••323

Major Networking and Telecom Standard Organizations••••••••••••••••••••••341

Network Communication Protocols Map••••••••••••••••••••••••••••••342

Figure

Figure 1-1: Communication between computers in a network••••••

Figure 2-21: SCSI Protocol Stack Structure••••••

Preface

We are living in the IT(Information Technologies) times. The IT provides us many powerful tools that have significantly changed our way of life, work and business operations. Among all the IT advancements, Internet has the most impact in every aspect of our society for the past 20 years. From Internet, people can get instant news, communicate with others, use it as a super-encyclopedia and find anything that they are interested in via search engines at their finger tips; Company can conduct business to business(B2B), business to consumer(B2C), with great efficiency; Government can announce polices, publicize regulations, and provide administrative information and services to the general public. Internet not only provides unprecedented convenience to our daily life, but also opens up new areas of disciplines and commercial opportunities that have boosted overall economy by creating many new jobs. It is reported that Internet will become a $20 trillion industry in the near future.

The Internet has also made significant progress and rapid adoption in China. According to the 14th Statistical Survey Report on the Internet Development in China announced on Jul 20, 2004 by CNNIC(China Internet Network Information Center), there are about 87 million Internet users as counted by the end of June 30, 2004, in mainland China, second only to the US; There are about 36 million computer hosts; The number of domain names registered under CN is 382216; The number of “www” websites is 626,600. It should be also noted that China has started its CNGI(China Next Generation Internet) project at the beginning of 2000, right after US and Europe started the similar initiatives. China now is becoming one of the most important and influential members not only in the World Trade Organization, but also within the Internet community.

To build the Internet and many other networks, engineers and organizations around the world have created many technologies over the past 20 years, in which network protocol is one of the key technology areas. After years of development on the communication standards and generations of networking architecture, network communication protocols have become a very complex subject. Various standard organizations have defined many communication protocols and all major vendors have their own proprietary technologies. Yet, people in the industry are continuously proposing and designing new protocols to address new problems in the network communications. It has become a huge challenge for IT and network professionals at all levels to understand the overall picture of communication protocols and to keep up with the pace of its on-going evolutions.

Javvin Company, based on Silicon Valley in California, USA, is a network software provider. This book is one of its contributions to provide an overview of network protocols and to serve as a reference and handbook for IT and network professionals.. The book fully explains and reviews all commonly used network communication protocols, including TCP/IP, security, VOIP, WAN, LAN , MAN, SAN and ISO protocols. It also covers Cisco, Novell, IBM, Microsoft, Apple and DEC network protocols. Hundreds of hyperlinks of references for further reading and studies are available in the book. It is an excellent reference for Internet programmers, network professionals and college students who are majoring IT and networking technology. It is also useful for individuals who want to know more details about the technologies underneath the Internet. I highly recommend this book to our readers.

Ke Yan, Ph.D.

Chief Architect of Juniper Networks
Founder of NetScreen Technologies

Network Communication Architecture and Protocols

A network architecture is a blueprint of the complete computer communication network, which provides a framework and technology foundation for designing, building and managing a communication network. It typically has a layered structure. Layering is a modern network design principle which divides the communication tasks into a number of smaller parts, each part accomplishing a particular sub-task and interacting with the other parts in a small number of well-defined ways. Layering allows the parts of a communication to be designed and tested without a combinatorial explosion of cases, keeping each design relatively simple.

If a network architecture is open, no single vendor owns the technology and controls its definition and development. Anyone is free to design hardware and software based on the network architecture. The TCP/IP network architecture, which the Internet is based on, is such a open network architecture and it is adopted as a worldwide network standard and widely deployed in local area network (LAN), wide area network (WAN), small and large enterprises, and last but not the least, the Internet.

Open Systems Interconnection (OSI) network architecture, developed by International Organization for Standardization, is an open standard for communication in the network across different equipment and applications by different vendors. Though not widely deployed, the OSI 7 layer model is considered the primary network architectural model for inter-computing and inter-networking communications.

In addition to the OSI network architecture model, there exist other network architecture models by many vendors, such as IBM SNA (Systems Network Architecture), Digital Equipment Corporation (DEC; now part of HP) DNA (Digital Network Architecture), Apple computer's AppleTalk, and Novell's NetWare. Actually, the TCP/IP architecture does not exactly match the OSI model. Unfortunately, there is no universal agreement regarding how to describe TCP/IP with a layered model. It is generally agreed that TCP/IP has fewer levels (from three to five layers) than the seven layers of the OSI model.

Network architecture provides only a conceptual framework for communications between computers. The model itself does not provide specific methods of communication. Actual communication is defined by various communication protocols.

OSI Network Architecture 7 Layers Model

Open Systems Interconnection (OSI) model is a reference model developed by ISO (International Organization for Standardization) in 1984, as a conceptual framework of standards for communication in the network across different equipment and applications by different vendors. It is now considered the primary architectural model for inter-computing and internetworking communications. Most of the network communication protocols used today have a structure based on the OSI model. The OSI model defines the communications process into 7 layers, dividing the tasks involved with moving information between networked computers into seven smaller, more manageable task groups. A task or group of tasks is then assigned to each of the seven OSI layers. Each layer is reasonably self-contained so that the tasks assigned to each layer can be implemented independently. This enables the solutions offered by one layer to be updated without adversely affecting the other layers.

The OSI 7 layers model has clear characteristics at each layer. Basically, layers 7 through 4 deal with end to end communications between data source and destinations, while layers 3 to 1 deal with communications between network devices. On the other hand, the seven layers of the OSI model can be divided into two groups: upper layers (layers 7, 6 & 5) and lower layers (layers 4, 3, 2, 1). The upper layers of the OSI model deal with application issues and generally are implemented only in software. The highest layer, the application layer, is closest to the end user. The lower layers of the OSI model handle data transport issues. The physical layer and the data link layer are implemented in hardware and software. The lowest layer, the physical layer, is closest to the physical network medium (the wires, for example) and is responsible for placing data on the medium.

The specific description for each layer is as follows:

Layer 7: Application Layer

- Defines interface to user processes for communication and data transfer in network
- Provides standardized services such as virtual terminal, file and job transfer and operations

Layer 6: Presentation Layer

- Masks the differences of data formats between dissimilar systems
- Specifies architecture-independent data transfer format
- Encodes and decodes data; encrypts and decrypts data; compresses and decompresses data

Layer 5: Session Layer

- Manages user sessions and dialogues
- Controls establishment and termination of logic links between users
- Reports upper layer errors

Layer 4: Transport Layer

- Manages end-to-end message delivery in network
- Provides reliable and sequential packet delivery through error recovery and flow control mechanisms
- Provides connectionless oriented packet delivery

Layer 3: Network Layer

- Determines how data are transferred between network devices
- Routes packets according to unique network device addresses
- Provides flow and congestion control to prevent network resource depletion

Layer 2: Data Link Layer

- Defines procedures for operating the communication links
- Frames packets
- Detects and corrects packets transmit errors

Layer 1: Physical Layer

- Defines physical means of sending data over network devices
- Interfaces between network medium and devices
- Defines optical, electrical and mechanical characteristics

Information being transferred from a software application in one computer to an application in another proceeds through the OSI layers. For example, if a software application in computer A has information to pass to a software application in computer B, the application program in computer A need to pass the information to the application layer (Layer 7) of computer A, which then passes the information to the presentation layer (Layer 6), which relays the data to the session layer (Layer 5), and so on all the way down to the physical layer (Layer 1). At the physical layer, the data is placed on the physical network medium and is sent across the medium to computer B. The physical layer of computer B receives the data from the physical medium, and then its physical layer passes the information up to the data link layer (Layer 2), which relays it to the network layer (Layer 3), and so on, until it reaches the application layer (Layer 7) of computer B. Finally, the application layer of computer B passes the information to the recipient application program to complete the communication process. The following diagram illustrated this process.

The seven OSI layers use various forms of control information to communicate with their peer layers in other computer systems. This control information consists of specific requests and instructions that are exchanged between peer OSI layers. Headers and Trailers of data at each layer are the two basic forms to carry the control information.

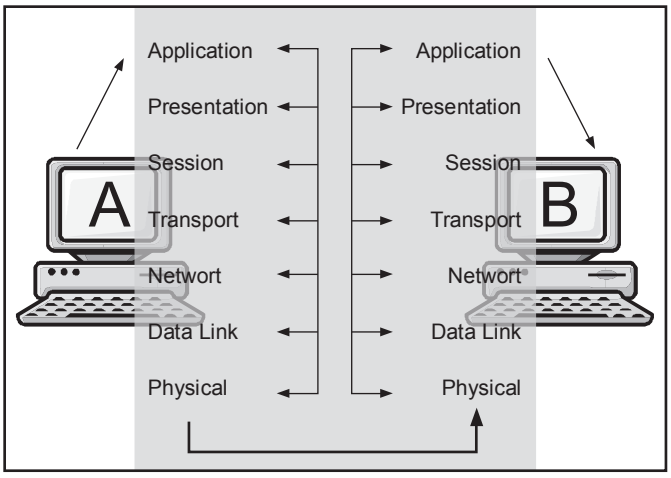


Figure 1-1: Communication between computers in a network

Headers are prepended to data that has been passed down from upper layers. Trailers are appended to data that has been passed down from upper layers. An OSI layer is not required to attach a header or a trailer to data from upper layers.

Each layer may add a Header and a Trailer to its Data, which consists of the upper layer's Header, Trailer and Data as it proceeds through the layers. The Headers contain information that specifically addresses layer-to-layer communication. Headers, trailers and data are relative concepts, depending on the layer that analyzes the information unit. For example, the Transport Header (TH) contains information that only the Transport layer sees. All other layers below the Transport layer pass the Transport Header as part of their Data. At the network layer, an information unit consists of a Layer 3 header (NH) and data.

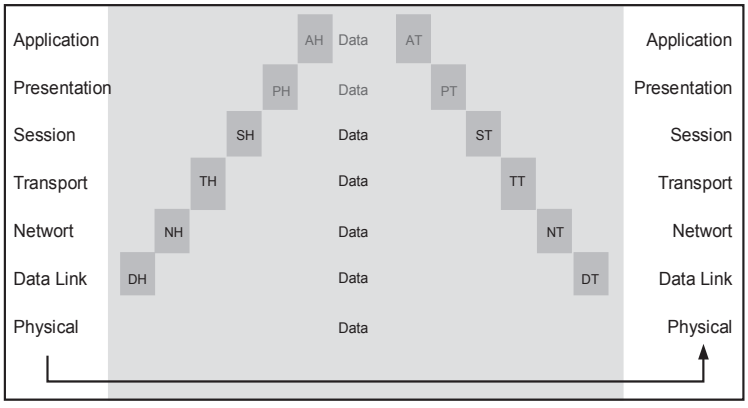


Figure 1-2: Data encapsulation at each layer

At the data link layer, however, all the information passed down by the network layer (the Layer 3 header and the data) is treated as data. In other words, the data portion of an information unit at a given OSI layer potentially can contain headers, trailers, and data from all the higher layers. This is known as encapsulation.

For example, if computer A has data from a software application to send to computer B, the data is passed to the application layer. The application layer in computer A then communicates any control information required by the application layer in computer B by prepending a header to the data. The resulting message unit, which includes a header, the data and maybe a trailer, is passed to the presentation layer, which prepends its own header containing control information intended for the presentation layer in computer B. The message unit grows in size as each layer prepends its own header and trailer containing control information to be used by its peer layer in computer B. At the physical layer, the entire information unit is transmitted through the network medium.

The physical layer in computer B receives the information unit and passes it to the data link layer. The data link layer in computer B then reads the control information contained in the header prepended by the data link layer in computer A. The header and the trailer are then removed, and the remainder of the information unit is passed to the network layer. Each layer performs the same actions: The layer reads the header and trailer from its peer layer, strips it off, and passes the remaining information unit to the next higher layer. After the application layer performs these actions, the data is passed to the recipient software application in computer B, in exactly the form in which it was transmitted by the application in computer A.

One OSI layer communicates with another layer to make use of the services provided by the second layer. The services provided by adjacent layers help a given OSI layer communicate with its peer layer in other computer systems. A given layer in the OSI model generally communicates with three other OSI layers: the layer directly above it, the layer directly below it and its peer layer in other networked computer systems. The data link layer in computer A, for example, communicates with the network layer of computer A, the physical layer of computer A and the data link layer in computer B. The following chart illustrates this example.

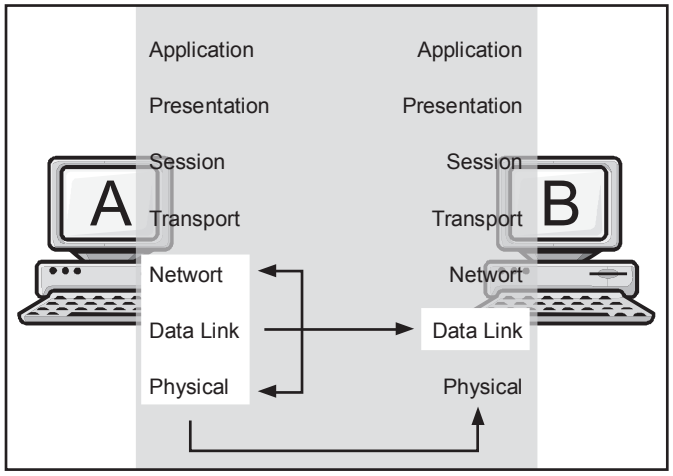


Figure 1-3: Data communication between peer layers

TCP/IP Four Layers Architecture Model

TCP/IP architecture does not exactly follow the OSI model. Unfortunately, there is no universal agreement regarding how to describe TCP/IP with a layered model. It is generally agreed that TCP/IP has fewer levels (from three to five layers) than the seven layers of the OSI model. We adopt a four layers model for the TCP/IP architecture.

TCP/IP architecture omits some features found under the OSI model, combines the features of some adjacent OSI layers and splits other layers apart. The 4-layer structure of TCP/IP is built as information is passed down from applications to the physical network layer. When data is sent, each layer treats all of the information it receives from the upper layer as data, adds control information (header) to the front of that data and then pass it to the lower layer. When data is received, the opposite procedure takes place as each layer processes and removes its header before passing the data to the upper layer.

The TCP/IP 4-layer model and the key functions of each layer is described below:

Application Layer

The Application Layer in TCP/IP groups the functions of OSI Application, Presentation Layer and Session Layer. Therefore any process above the transport layer is called an Application in the TCP/IP architecture. In TCP/IP socket and port are used to describe the path over which applications communicate. Most application level protocols are associated with one or more port number.

Transport Layer

In TCP/IP architecture, there are two Transport Layer protocols. The Transmission Control Protocol (TCP) guarantees information transmission. The User Datagram Protocol (UDP) transports datagram swithout end-to-end reliability checking. Both protocols are useful for different applications.

Network Layer

The Internet Protocol (IP) is the primary protocol in the TCP/IP Network Layer. All upper and lower layer communications must travel through IP as they are passed through the TCP/IP protocol stack. In addition, there are many supporting protocols in the Network Layer, such as ICMP, to facilitate and manage the routing process.

Network Access Layer

In the TCP/IP architecture, the Data Link Layer and Physical Layer are normally grouped together to become the Network Access layer. TCP/IP makes use of existing Data Link and Physical Layer standards rather than defining its own. Many RFCs describe how IP utilizes and interfaces with the existing data link protocols such as Ethernet, Token Ring, FDDI, HSSI, and ATM. The physical layer, which defines the hardware communication properties, is not often directly interfaced with the TCP/IP protocols in the network layer and above.

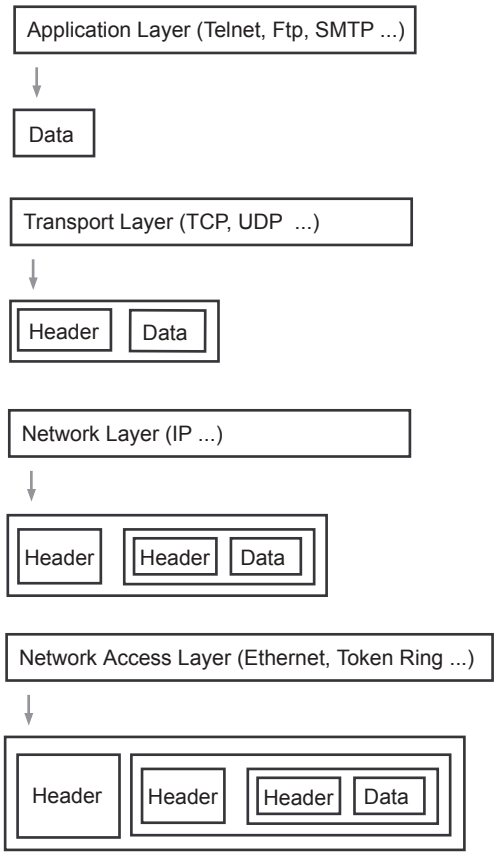


Figure 1-4: TCP/IP Protocol Stack 4 Layer Model

Other Network Architecture Models: IBM SNA

In addition to the open architectural models such as OSI 7 layers model and the TCP/IP model, there exist a few popular vendor specific network communication models, such as IBM SNA (Systems Network Architecture), Digital Equipment Corporation's (DEC, now part of HP) DNA (Digital Network Architecture). We will only provide details on the IBM SNA here.

Although it is now considered a legacy networking architecture, the IBM SNA is still widely deployed. SNA was designed around the host-to-terminal communication model that IBM's mainframes use. IBM expanded the SNA protocol to support peer-to-peer networking. This expansion was deemed Advanced Peer-to-Peer Networking (APPN) and Advanced Program-to-Program Communication (APPC). Advanced Peer-to-Peer Networking (APPN) represents IBM's second-generation SNA. In creating APPN, IBM moved SNA from a hierarchical, mainframe-centric environment to a peer-based networking environment. At the heart of APPN is an IBM architecture that supports peer-based communications, directory services, and routing between two or more APPC systems that are not directly attached.

SNA has many similarities with the OSI 7 layers reference model. However, the SNA model has only six layers and it does not define specific protocols for its physical control layer. The physical control layer is assumed to be implemented via other standards. The functions of each SNA component are described as follows:

- Data Link Control (DLC)—Defines several protocols, including the Synchronous Data Link Control (SDLC) protocol for hierarchical communication, and the Token Ring Network communication protocol for LAN communication between peers. SDLC provided a foundation for ISO HDSL and IEEE 802.2.
- Path control—Performs many OSI network layer functions, including routing and datagram segmentation and reassembly (SAR)
- Transmission control—Provides a reliable end-to-end connection service (similar to TCP), as well as encrypting and decrypting services
- Data flow control—Manages request and response processing, determines whose turn it is to communicate, groups messages and interrupts data flow on request
- Presentation services—Specifies data-transformation algorithms that translate data from one format to another, coordinate resource sharing and synchronize transaction operations
- Transaction services—Provides application services in the form of programs that implement distributed processing or management services

The following figure illustrates how the IBM SNA model maps to the ISO OSI reference model.

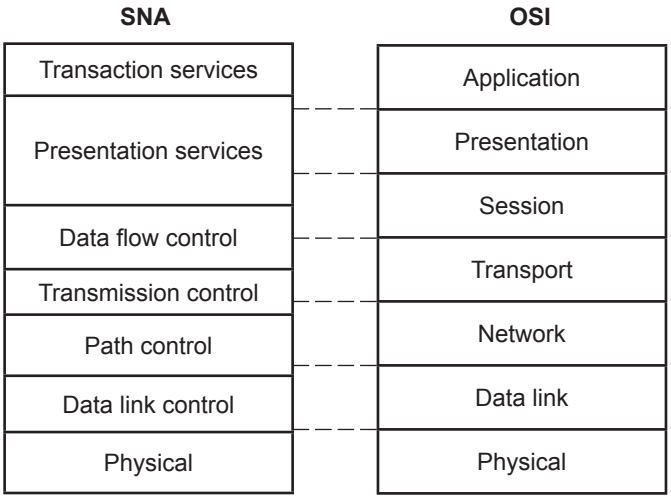


Figure 1-5: SNA vs. OSI model

A typical SNA network topology:

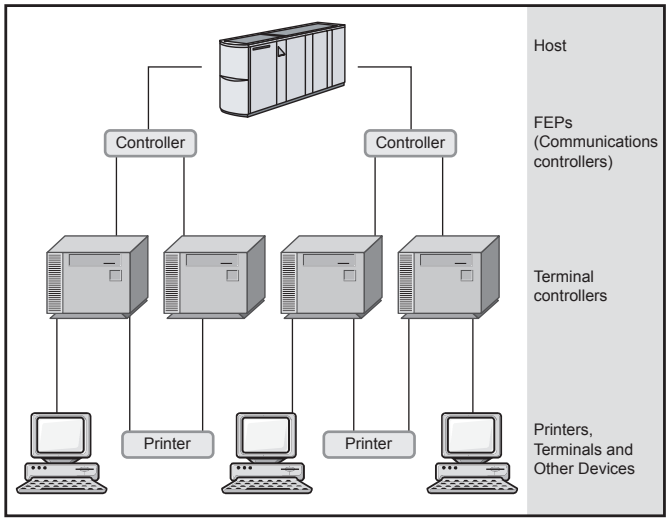


Figure 1-6: SNA Network Topology

SNA supports the following types of networks:

- A subarea network is a hierarchically organized network consisting of subarea nodes and peripheral nodes. Subarea nodes, such as hosts and communication controllers, handle general network routing. Peripheral nodes, such as terminals, attach to the network without awareness of general network routing.
- A peer network is a cooperatively organized network consisting of peer nodes that all participate in general network routing.
- A mixed network is a network that supports both host-controlled communications and peer communications.

In SNA networks, programs that exchange information across the SNA network are called transaction programs (TPs). Communication between a TP and the SNA network occurs through network accessible units or NAUs (formerly called “network addressable units”), which are unique network resources that can be accessed (through unique local addresses) by other network resources. There are three types of NAU: Physical Unit, Logic Units and Control Points.

Communication between Transaction Programs (TP) and Logic Units (LU) is shown as follows:

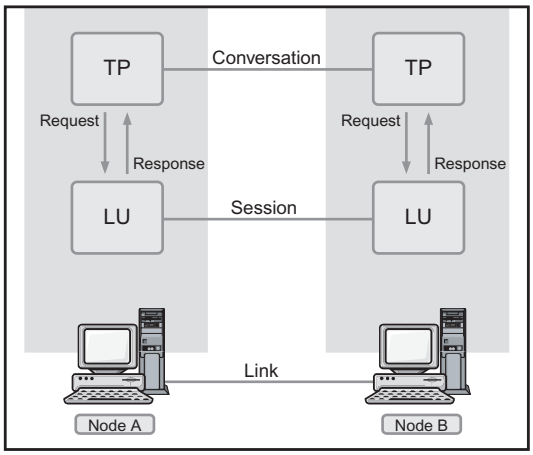


Figure 1-7: Communication between TP and LU in SNA

Network Protocol: Definition and Overview

The OSI model, and any other network communication model, provide only a conceptual framework for communication between computers, but the model itself does not provide specific methods of communication. Actual communication is defined by various communication protocols. In the context of data communication, a protocol is a formal set of rules, conventions and data structure that governs how computers and other network devices exchange information over a network. In other words, a protocol is a standard procedure and format that two data communication devices must understand, accept and use to be able to talk to each other.

In modern protocol design, protocols are “layered” according to the OSI 7 layer model or a similar layered model. Layering is a design principle which divides the protocol design into a number of smaller parts, each part accomplishing a particular sub-task, and interacting with the other parts of the protocol only in a small number of well-defined ways. Layering allows the parts of a protocol to be designed and tested without a combinatorial explosion of cases, keeping each design relatively simple. Layering also permits familiar protocols to be adapted to unusual circumstances.

The header and/or trailer at each layer reflect the structure of the protocol. Detailed rules and procedures of a protocol or protocol stack are often defined by a lengthy document. For example, IETF uses RFCs (Request for Comments) to define protocols and updates to the protocols.

A wide variety of communication protocols exist. These protocols are defined by many standard organizations throughout the world and by technology vendors over years of technology evolution and development. One of the most popular protocol suites is TCP/IP, which is the heart of Internetworking communications. The IP, the Internet Protocol, is responsible for exchanging information between routers so that the routers can select the proper path for network traffic, while TCP is responsible for ensuring the data packets are transmitted across the network reliably and error free. LAN and WAN protocols are also critical protocols in network communications. The LAN protocols suite is for the physical and data link layers communications over various LAN media such as Ethernet wires and wireless waves. The WAN protocol suite is for the lowest three layers and defines communication over various wide-area media, such as fiber optic and copper cable.

Network communication has gradually evolved – Today’s new technologies are based on accumulation over years of technologies, which may be still existing or obsolete. Because of this, the protocols which define the network communication, are highly inter-related. Many protocols rely on others for operation. For example, many routing protocols use other network protocols to exchange information between routers.

In addition to standards for individual protocols in transmission, there are now also interface standards for different layers to talk to the ones above or below (usually operating-system-specific). For example: Winsock and Berkeley sockets between layers 4 and 5, NDIS and ODI between layers 2 and 3.

The protocols for data communication cover all areas as defined in the OSI model. However, the OSI model is only loosely defined. A protocol may perform the functions of one or more of the OSI layers, which introduces complexity to understand protocols relevant to the OSI 7 layer model. In real-world protocols, there is some argument as to where the distinctions between layers are drawn; there is no one black and white answer.

To develop a complete technology that is useful for the industry, very often a group of protocols is required in the same layer or across many different layers. Different protocols often describe different aspects of a single communication; taken together, these form a protocol suite. For example, Voice over IP (VOIP), a group of protocols developed by many vendors and standard organizations, has many protocols across the 4 top layers in the OSI model.

Protocols can be implemented either in hardware or software, or a mixture of both. Typically, the lower layers are implemented in hardware, with the higher layers being implemented in software.

Protocols could be grouped into suites (or families, or stacks) by their technical functions, or origin of the protocol introduction, or both. A protocol may belong to one or multiple protocol suites, depending on how you categorize it. For example, the Gigabit Ethernet protocol IEEE 802.3z is a LAN (Local Area Network) protocol and it can also be used in MAN (Metropolitan Area Network) communications.

Most recent protocols are designed by the IETF for Internetworking communications, and the IEEE for local area networking (LAN) and metropolitan area networking (MAN). The ITU-T contributes mostly to wide area networking (WAN) and telecommunications protocols. ISO has its own suite of protocols for internetworking communications, which is mainly deployed in European countries.

Protocols Guide

TCP/IP Protocols

The TCP/IP protocol suite establishes the technical foundation of the Internet. Development of the TCP/IP started as DOD projects. Now, most protocols in the suite are developed by the Internet Engineering Task Force (IETF) under the Internet Architecture Board (IAB), an organization initially sponsored by the US government and now an open and autonomous organization. The IAB provides the coordination for the R&D underlying the TCP/IP protocols and guides the evolution of the Internet. The TCP/IP protocols are well documented in the Request For Comments (RFC), which are drafted, discussed, circulated and approved by the IETF committees. All documents are open and free and can be found online in the IETF site listed in the reference.

TCP/IP architecture does not exactly match the OSI model. Unfortunately, there is no universal agreement regarding how to describe TCP/IP with a layered model. It is generally agreed that TCP/IP has fewer levels (from three to five layers) than the seven layers of the OSI model. In this article, we force TCP/IP protocols into the OSI 7 layers structure for comparison purpose.

The TCP/IP suite's core functions are addressing and routing (IP/IPv6 in the networking layer) and transportation control (TCP, UDP in the transport layer).

IP - Internet Protocol

Addressing of network components is a critical issue for information routing and transmission in network communications. Each technology has its own convention for transmitting messages between two machines within the same network. On a LAN, messages are sent between machines by supplying the six bytes unique identifier (the "MAC" address). In an SNA network, every machine has Logical Units with their own network addresses. DECNET, AppleTalk, and Novell IPX all have a scheme for assigning numbers to each local network and to each workstation attached to the network.

On top of these local or vendor specific network addresses, IP assigns a unique number to every network device in the world, which is called an IP address. This IP address is a four bytes value in IPv4 that, by convention, is expressed by converting each byte into a decimal number (0 to 255) and separating the bytes with a period. In IPv6, the IP address has been increased to 16 bytes. Details of the IP and IPv6 protocols are presented in separate documents.

TCP - Transmission Control Protocol

TCP provides a reliable stream delivery and virtual connection service to applications through the use of sequenced acknowledgment with retransmission of packets when necessary. TCP provides stream data transfer, trans-

portation reliability, efficient flow control, full-duplex operation, and multiplexing. Check the TCP section for more details.

In the follwoing TCP/IP protocol stack table, we list all the protocols according to their functions in mapping to the OSI 7 layers network communication reference model. However, the TCP/IP architecture does not follow the OSI model closely, for example, most TCP/IP applications directly run on top of the transport layer protocols, TCP and UDP, without the presentation and session layers in between.

TCP/IP Protocol Stack

Application Layer

BOOTP: Bootstrap Protocol
DCAP: Data Link Switching Client Access Protocol
DHCP: Dynamic Host Configuration Protocol
DNS: Domain Name Systems
FTP: File Transfer Protocol
Finger: User Information Protocol
HTTP: Hypertext Transfer Protocol
S-HTTP: Secure Hypertext Transfer Protocol (S-HTTP)
IMAP & IMAP4: Internet Message Access Protocol
IPDC: IP Device Control
IRCP (IRC): Internet Relay Chat Protocol
LDAP: Lightweighted Directory Access Protocol
MIME (S-MIME): Multipurpose Internet Mail Extensions (Secure MIME)
NAT: Network Address Translation
NNTP: Network News Transfer Protocol
NTP: Network Time Protocol
POP & POP3: Post Office Protocol (version 3)
RLOGIN: Remote Login in Unix
RMON: Remote Monitoring MIBs in SNMP
SLP: Service Location Protocol
SMTP: Simple Mail Transfer Protocol
SNMP: Simple Network Management Protocol
SNTP: Simple Network Time Protocol
TELNET: TCP/IP Terminal Emulation Protocol
TFTP: Trivial File Transfer Protocol
URL: Uniform Resource Locator
X-Window: X Window or X Protocol or X System

Presentation Layer

LPP: Lightweight Presentation Protocol

Session Layer

RPC: Remote Procedure Call protocol

Transport Layer

ITOT: ISO Transport Over TCP/IP
RDP: Reliable Data Protocol
RUDP: Reliable UDP
TALI: Transport Adapter Layer Interface
TCP: Transmission Control Protocol
UDP: User Datagram Protocol
Van Jacobson: Compressed TCP

Network Layer

Routing

BGP/BGP4: Border Gateway Protocol
EGP: Exterior Gateway Protocol
IP: Internet Protocol
IPv6: Internet Protocol version 6
ICMP/ICMPv6: Internet Control Message Protocol
IRDP: ICMP Router Discovery Protocol
Mobile IP: IP Mobility Support Protocol for IPv4 & IPv6
NARP: NBMA Address Resolution Protocol
NHRP: Next Hop Resolution Protocol
OSPF: Open Shortest Path First
RIP (RIP2): Routing Information Protocol
RIPng: RIP for IPv6
RSVP: Resource ReSerVation Protocol
VRRP: Virtual Router Redundancy Protocol

Multicast

BGMP: Border Gateway Multicast Protocol
DVMRP: Distance Vector Multicast Routing Protocol
IGMP: Internet Group Management Protocol
MARS: Multicast Address Resolution Server
MBGP: Multiprotocol BGP
MOSPF: Multicast OSPF
MSDP: Multicast Source Discovery Protocol
MZAP: Multicast-Scope Zone Announcement Protocol
PGM: Pragmatic General Multicast Protocol
PIM-DM: Protocol Independent Multicast - Dense Mode
PIM-SM: Protocol Independent Multicast - Sparse Mode

MPLS Protocols

MPLS: Multi-Protocol Label Switching
CR-LDP: Constraint-Based Label Distribution Protocol
LDP: Label Distribution Protocol
RSVP-TE: Resource ReSerVation Protocol-Traffic Engineering

Data Link Layer

ARP and InARP: Address Resolution Protocol and Inverse ARP
IPCP and IPv6CP: IP Control Protocol and IPv6 Control Protocol
RARP: Reverse Address Resolution Protocol
SLIP: Serial Line IP

Related protocol suites

LAN, MAN, WAN, SAN, Security/VPN

Sponsor Source

IETF, DARPA, ISO

Application Layer Protocols

Protocol Name

BOOTP: Bootstrap Protocol

Protocol Description

The Bootstrap Protocol (BOOTP) is a UDP/IP-based protocol which allows a booting host to configure itself dynamically and without user supervision. BOOTP provides a means to notify a host of its assigned IP address, the IP address of a boot server host and the name of a file to be loaded into memory and executed. Other configuration information such as the local subnet mask, the local time offset, the addresses of default routers and the addresses of various Internet servers, can also be communicated to a host using BOOTP.

BOOTP uses two different well-known port numbers. UDP port number 67 is used for the server and UDP port number 68 is used for the BOOTP client. The BOOTP client broadcasts a single packet called a BOOTREQUEST packet that contains the client's physical network address and optionally, its IP address if known. The client could send the broadcast using the address 255.255.255.255, which is a special address called the limited broadcast address. The client waits for a response from the server. If a response is not received within a specified time interval, the client retransmits the request.

The server responds to the client's request with a BOOTREPLY packet. The request can (optionally) contain the 'generic' filename to be booted, for example, 'unix' or 'ethertip'. When the server sends the bootreply, it replaces this field with the fully qualified path name of the appropriate boot file. In determining this name, the server may consult its own database correlating the client's address and filename request, with a particular boot file customized for that client. If the bootrequest filename is a null string, then the server returns a filename field indicating the 'default' file to be loaded for that client.

In the case of clients which do not know their IP addresses, the server must also have a database relating hardware address to IP address. This client IP address is then placed into a field in the bootreply.

BOOTP is an alternative to RARP, which operates at the Data Link Layer for LAN only. BOOTP, a UDP/IP based configuration protocol, provides much more configuration information and allows dynamic configuration for an entire IP network. BOOTP and its extensions became the basis for the Dynamic Host Configuration Protocol (DHCP).

Protocol Structure

8	16	24	32bit
Op	Htype	Hlen	Hops
Xid			
Secs		Flags	
Ciaddr			
Yiaddr			
Siaddr			
Giaddr			
Chaddr (16 bytes)			
Sname (64 bytes)			
File (128 bytes)			
Option (variable)			

Op	The message operation code. Messages can be either BOOTREQUEST or BOOTREPLY.
Htype	The hardware address type.
Hlen	The hardware address length.
Xid	The transaction ID.
Secs	The seconds elapsed since the client began the address acquisition or renewal process.
Flags	The flags.
Ciaddr	The client IP address.
Yiaddr	The "Your" (client) IP address.
Siaddr	The IP address of the next server to use in bootstrap.
Giaddr	The relay agent IP address used in booting via a relay agent.
Chaddr	The client hardware address.
Sname	Optional server host name, null terminated string
File	Boot file name, null terminated string; generic name or null in DHCPDISCOVER, fully qualified directory-path name in DHCPOFFER.
Options	Optional parameters field.

Related protocols

IP, UDP, DHCP, RARP

Sponsor Source

BOOTP is defined by IETF (http://www.ietf.org) RFC951 and RFC 1542.

Reference

http://www.javvin.com/protocol/rfc951.pdf
BOOTSTRAP PROTOCOL (BOOTP)
http://www.javvin.com/protocol/rfc1542.pdf
Clarifications and Extensions for the Bootstrap Protocol
http://www.javvin.com/protocol/rfc2132.pdf
DHCP Options and BOOTP Vendor Extensions
http://www.javvin.com/protocol/rfc3396.pdf
Encoding Long Options in the (DHCPv4)

Protocol Name

DCAP: Data Link Switching Client Access Protocol

Protocol Description

The Data Link Switching Client Access Protocol (DCAP) is an application layer protocol used between workstations and routers to transport SNA/NetBIOS traffic over TCP sessions.

DCAP was introduced to address a few deficiencies in the Data Link Switching Protocol (DLSw). The implementation of the Data Link Switching Protocol (DLSw) on a large number of workstations raises the important issues of scalability and efficiency. Since DLSw is a switch-to-switch protocol, it is not efficient when implemented on workstations. DCAP addresses these issues. It introduces a hierarchical structure to resolve the scalability problems. All workstations are clients to the router (server) rather than peers to the router. This creates a client/server model. It also provides a more efficient protocol between the workstation (client) and the router (server).

In a DLSw network, each workstation needs a MAC address to communicate with an FEP attached to a LAN. When DLSw is implemented on a workstation, it does not always have a MAC address defined. For example, when a workstation connects to a router through a modem via PPP, it only consists of an IP address. In this case, the user must define a virtual MAC address. This is administratively intensive since each workstation must have a unique MAC address. DCAP uses the Dynamic Address Resolution protocol to solve this problem. The Dynamic Address Resolution protocol permits the server to dynamically assign a MAC address to a client without complex configuration.

Protocol Structure

4	8	16bit
Protocol ID	Version Number	Message Type
Packet Length		

Protocol ID	The Protocol ID is set to 1000.
Version number	The Version number is set to 0001.
Message type	The message type is the DCAP message type.
Packet length	The total packet length is the length of the packet including the DCAP header, DCAP data and user data. The minimum size of the packet is 4, which is the length of the header.

Related protocols

TCP, DLSw, NetBIOS

Sponsor Source

DCAP is defined by IETF (http://www.ietf.org) in RFC 2114.

Reference

http://www.javvin.com/protocol/rfc2114.pdf
Data Link Switching Client Access Protocol

Protocol Name

DHCP: Dynamic Host Configuration Protocol

Protocol Description

Dynamic Host Configuration Protocol (DHCP) is a communications protocol enabling network administrators manage centrally and to automate the assignment of IP addresses in a network. In an IP network, each device connecting to the Internet needs a unique IP address. DHCP lets a network administrator supervise and distribute IP addresses from a central point and automatically sends a new IP address when a computer is plugged into a different place in the network.

DHCP uses the concept of a “lease” or amount of time that a given IP address will be valid for a computer. The lease time can vary depending on how long a user is likely to require the Internet connection at a particular location. It’s especially useful in education and other environments where users change frequently. Using very short leases, DHCP can dynamically reconfigure networks in which there are more computers than there are available IP addresses.

DHCP supports static addresses for computers containing Web servers that need a permanent IP address.

DHCP is an alternative to another network IP management protocol, Bootstrap Protocol (BOOTP). DHCP is a more advanced protocol but both configuration management protocols are commonly used. Some operating systems, including Windows NT/2000, come with DHCP servers. A DHCP or BOOTP client is a program that is located in each computer so that it can be configured.

Protocol Structure

8	16	24	32bit
Op	Htype	Hlen	Hops
Xid			
Secs		Flags	
Ciaddr			
Yiaddr			
Siaddr			
Giaddr			
Chaddr (16 bytes)			
Sname (64 bytes)			
File (128 bytes)			
Option (variable)			

Op The message operation code. Messages can be either BOOTREQUEST or BOOTREPLY.
Htype The hardware address type.
Hlen The hardware address length.
Xid The transaction ID.
Secs The seconds elapsed since the client began the address acquisition or renewal process.
Flags The flags.
Ciaddr The client IP address.
Yiaddr The “Your” (client) IP address.
Siaddr The IP address of the next server to use in bootstrap.
Giaddr The relay agent IP address used in booting via a relay agent.
Chaddr The client hardware address.
Sname Optional server host name, null terminated string
File Boot file name, null terminated string; generic name or null in DHCPDISCOVER, fully qualified directory-path name in DHCPOFFER.
Options Optional parameters field. See the options documents for a list of defined options.

Related protocols

IP, BOOTP, UDP, TCP, RARP

Sponsor Source

DHCP is defined by IETF (http://www.ietf.org) RFC2131 and RFC 3396.

Reference

http://www.javvin.com/protocol/rfc2131.pdf
Dynamic Host Configuration Protocol
http://www.javvin.com/protocol/rfc3396.pdf
Encoding Long Options in the (DHCPv4)

Protocol Name

DNS: Domain Name System (Service) protocol

Protocol Description

Domain Name System (DNS) is a distributed Internet directory service. DNS is used mostly to translate between domain names and IP addresses and to control Internet email delivery. Most Internet services rely on DNS to work, and if DNS fails, web sites cannot be located and email delivery stalls.

DNS has two independent aspects:

1. It specifies the name syntax and rules for delegating authority over names. The basic syntax is:

local.group.site

2. It specifies the implementation of a distributed computing system that efficiently maps names to addresses.

In the DNS naming scheme, a decentralized and hierarchical mechanism is used by the delegating authority for parts of the namespace and distributing responsibility for mapping names and addresses. The naming scheme of DNS is used to assign network device names globally and is implemented by geographically distributed sets of severs to names to addresses.

In theory, the domain name standard in DNS protocol specifies an abstract hierarchical namespace with arbitrary values for labels. Any group can build an instance of the domain system to choose labels for all parts of its hierarchy. However most users of the DNS protocols follow the hierarchical labels used by the official Internet domain system. Some of the top level domains are: COM, EDU, GOV, NET, ORG, BIZ ... plus many country codes.

The distributed scheme of DNS allows efficient and reliable mapping of names to IP addresses. Most names can be mapped locally and a set of servers operating at multiple sites cooperatively solve the mapping problem of a large network. Because of the distributing nature, no single machine failure will prevent the DNS from operating correctly.

Protocol Structure

<table>
<tr><td colspan="3">16</td><td>21</td><td colspan="3"></td><td>28</td><td colspan="2">32bit</td></tr>
<tr><td colspan="3">ID</td><td>Q</td><td>Query</td><td>A</td><td>T</td><td>R</td><td>V</td><td>B</td><td>Rcode</td></tr>
<tr><td colspan="3">Question count</td><td colspan="7">Answer count</td></tr>
<tr><td colspan="3">Authority count</td><td colspan="7">Additional count</td></tr>
</table>

ID 16-bit field used to correlate queries and responses.

Q 1-bit field that identifies the message as a query or response.

Query 4-bit field that describes the type of message: 0 Standard query (name to address); 1 Inverse query; 2 Server status request.

A Authoritative Answer. 1-bit field. When set to 1, identifies the response as one made by an authoritative name server.

T Truncation. 1-bit field. When set to 1, indicates the message has been truncated.

R 1-bit field. Set to 1 by the resolve to request recursive service by the name server.

V 1-bit field. Signals the availability of recursive service by the name server.

B 3-bit field. Reserved for future use. Must be set to 0.

Rcode Response Code. 4-bit field that is set by the name server to identify the status of the query.

Question count 16-bit field that defines the number of entries in the question section.

Answer count 16-bit field that defines the number of resource records in the answer section.

Authority count 16-bit field that defines the number of name server resource records in the authority section.

Additional count 16-bit field that defines the number of resource records in the additional records section.

Related protocols

IP, TCP, IGMP, ICMP, SNMP, TFTP and NFS

Sponsor Source

DNS is defined by IETF (http://www.ietf.org) RFC1034 and updated by 1035, 1101, 1183, 1348, 1876, 1982, 2181, 2308, 2535

Reference

http://www.javvin.com/protocol/rfc1034.pdf
Domain Names – Concept and Facilities

Protocol Name

FTP: File Transfer Protocol

Protocol Description

File Transfer Protocol (FTP) enables file sharing between hosts. FTP uses TCP to create a virtual connection for control information and then creates a separate TCP connection for data transfers. The control connection uses an image of the TELNET protocol to exchange commands and messages between hosts.

The key functions of FTP are:

1) to promote sharing of files (computer programs and/or data);
2) to encourage indirect or implicit (via programs) use of remote computers;
3) to shield a user from variations in file storage systems among hosts; and
4) to transfer data reliably and efficiently.

FTP, though usable directly by a user at a terminal, is designed mainly for use by programs. FTP control frames are TELNET exchanges and can contain TELNET commands and option negotiation. However, most FTP control frames are simple ASCII text and can be classified as FTP commands or FTP messages. FTP messages are responses to FTP commands and consist of a response code followed by explanatory text.

Protocol Structure

Command	*Description*
ABOR	Abort data connection process.
ACCT <account>	Account for system privileges.
ALLO <bytes>	Allocate bytes for file storage on server.
APPE <filename>	Append file to file of same name on server.
CDUP <dir path>	Change to parent directory on server.
CWD <dir path>	Change working directory on server.
DELE <filename>	Delete specified file on server.
HELP <command>	Return information on specified command.
LIST <name>	List information if name is a file or list files if name is a directory.
MODE <mode>	Transfer mode (S=stream, B=block, C=compressed).
MKD <directory>	Create specified directory on server.
NLST <directory>	List contents of specified directory.
NOOP	Cause no action other than acknowledgement from server.
PASS <password>	Password for system log-in.
PASV	Request server wait for data connection.
PORT <address>	IP address and two-byte system port ID.
PWD	Display current working directory.
QUIT	Log off from the FTP server.
REIN	Reinitialize connection to log-in status.
REST <offset>	Restart file transfer from given offset.
RETR <filename>	Retrieve (copy) file from server.
RMD <directory>	Remove specified directory on server.
RNFR <old path>	Rename from old path.
RNTO <new path>	Rename to new path.
SITE <params>	Site specific parameters provided by server.
SMNT <pathname>	Mount the specified file structure.
STAT <directory>	Return information on current process or directory.
STOR <filename>	Store (copy) file to server.
STOU <filename>	Store file to server name.
STRU <type>	Data structure (F=file, R=record, P=page).
SYST	Return operating system used by server.
TYPE <data type>	Data type (A=ASCII, E=EBCDIC, I=binary).
USER <username>	User name for system log-in.

Standard FTP messages are as follows:

Response Code	*Explanatory Text*
110	Restart marker at MARK yyyy=mmmm (new file pointers).
120	Service ready in nnn minutes.
125	Data connection open, transfer starting.
150	Open connection.
200	OK.
202	Command not implemented.
211	(System status reply).
212	(Directory status reply).
213	(File status reply).
214	(Help message reply).
215	(System type reply).
220	Service ready.
221	Log off network.
225	Data connection open.
226	Close data connection.
227	Enter passive mode (IP address, port ID).
230	Log on network.
250	File action completed.
257	Path name created.
331	Password required.
332	Account name required.
350	File action pending.
421	Service shutting down.
425	Cannot open data connection.
426	Connection closed.
450	File unavailable.
451	Local error encountered.
452	Insufficient disk space.
500	Invalid command.
501	Bad parameter.
502	Command not implemented.

503	Bad command sequence.
504	Parameter invalid for command.
530	Not logged onto network.
532	Need account for storing files.
550	File unavailable.
551	Page type unknown.
552	Storage allocation exceeded.
553	File name not allowed.

Related protocols

TELNET

Sponsor Source

FTP is defined by IETF (http://www.ietf.org) in RFC 959 and updated by 2228, 2640 and 2773.

Reference

http://www.javvin.com/protocol/rfc959.pdf
FILE TRANSFER PROTOCOL (FTP)

Protocol Name

Finger: User Information Protocol

Protocol Description

The Finger user information protocol provides an interface to a remote user information program (RUIP). Finger, based on the Transmission Control Protocol, is a protocol for the exchange of user information using TCP port 79. The local host opens a TCP connection to a remote host on the Finger port. An RUIP becomes available on the remote end of the connection to process the request. The local host sends the RUIP a one line query based upon the Finger query specification and waits for the RUIP to respond. The RUIP receives and processes the query, returns an answer, then initiates the close of the connection. The local host receives the answer and the close signal and then proceeds to close its end of the connection.

Finger discloses information about users; moreover, such information may be considered sensitive. Security administrators should make explicit decisions about whether to run Finger and what information should be provided in responses. One existing implementation provides the time the user last logged in, the time he last read mail, whether unread mail was waiting for him and who the most recent unread mail was from! This makes it possible to track conversations in progress and see where someone's attention was focused. Sites that are information-security conscious should not run Finger without an explicit understanding of how much information it is giving away.

Implementations should be tested against various forms of attack. In particular, an RUIP SHOULD protect itself against malformed inputs. Vendors providing Finger with the operating system or network software should subject their implementations to penetration testing. Finger is one of the avenues for direct penetration. Like Telnet, FTP and SMTP, Finger is one of the protocols at the security perimeter of a host. Accordingly, the soundness of the implementation is paramount. The implementation should receive just as much security scrutiny during design, implementation, and testing as Telnet, FTP, or SMTP.

Protocol Structure

Any data transferred between two Finger hosts MUST be in ASCII format, with no parity, and with lines ending in CRLF (ASCII 13 followed by ASCII 10). This excludes other character formats such as EBCDIC, etc. This also means that any characters between ASCII 128 and ASCII 255 should truly be international data, not 7-bit ASCII with the parity bit set.

The Finger query specification is defined:

```
{Q1}    ::= [{W}|{W}{S}{U}]{C}
{Q2}    ::= [{W}{S}][{U}]{H}{C}
{U}     ::= username
{H}     ::= @hostname | @hostname{H}
{W}     ::= /W
{W}     ::= /W
{S}     ::= <SP> | <SP>{S}
{C}     ::= <CRLF>
```

Related protocols

TCP, TELNET, SMTP, FTP

Sponsor Source

Finger is defined by IETF (http://www.ietf.org) in RFC 1288.

Reference

http://www.javvin.com/protocol/rfc1288.pdf
FILE TRANSFER PROTOCOL (FTP)

Protocol Name

HTTP: Hypertext Transfer Protocol

Protocol Description

The Hypertext Transfer Protocol (HTTP) is an application-level protocol with the lightness and speed necessary for distributed, collaborative, hypermedia information systems. HTTP has been in use by the World-Wide Web global information initiative since 1990.

HTTP allows an open-ended set of methods to be used to indicate the purpose of a request. It builds on the discipline of reference provided by the Uniform Resource Identifier (URI), as a location (URL) or name (URN), for indicating the resource on which a method is to be applied. Messages are passed in a format similar to that used by Internet Mail and the Multipurpose Internet Mail Extensions (MIME).

HTTP is also used as a generic protocol for communication between user agents and proxies/gateways to other Internet protocols, such as SMTP, NNTP, FTP, Gopher and WAIS, allowing basic hypermedia access to resources available from diverse applications and simplifying the implementation of user agents.

The HTTP protocol is a request/response protocol. A client sends a request to the server in the form of a request method, URI, and protocol version, followed by a MIME-like message containing request modifiers, client information, and possible body content over a connection with a server. The server responds with a status line, including the message's protocol version and a success or error code, followed by a MIME-like message containing server information, entity meta information, and possible entity-body content.

The first version of HTTP, referred to as HTTP/0.9, was a simple protocol for raw data transfer across the Internet. HTTP/1.0, as defined by RFC 1945, improved the protocol by allowing messages to be in the format of MIME-like messages, containing meta information about the data transferred and modifiers on the request/response semantics. However, HTTP/1.0 does not sufficiently take into consideration the effects of hierarchical proxies, caching, the need for persistent connections, or virtual hosts. "HTTP/1.1" includes more stringent requirements than HTTP/1.0 in order to ensure reliable implementation of its features. There is a secure version of HTTP (S-HTTP) specification, which will be discussed in a separate document.

Protocol Structure

HTTP messages consist of requests from client to server and responses from server to client.

The request message has the following format:

Request Line	General header	Request header	Entity header	Message Body

The Request-Line begins with a method token, followed by the Request-URI and the protocol version, and ends with CRLF. The elements are separated by SP characters. No CR or LF is allowed except in the final CRLF sequence. The details of the general header, request header and entity header can be found in the reference documents.

The response message has the following format:

Status Line	General header	Response header	Entity header	Message Body

The Status-Code element is a 3-digit integer result code of the attempt to understand and satisfy the request. The Reason-Phrase is intended to give a short textual description of the Status-Code. The Status-Code is intended for use by automata and the Reason-Phrase is intended for the human user. The client is not required to examine or display the Reason-Phrase. The details of the general header, response header and entity header could be found in the reference documents.

Related protocols

WWW, FTP, STMP, NNTP, Gopher, WAIS, DNS, S-HTTP

Sponsor Source

HTTP is defined by IETF (http://www.ietf.org) in RFC 1945 and 2616.

Reference

http://www.javvin.com/protocol/rfc1945.pdf
Hypertext Transfer Protocol -- HTTP 1.0
http://www.javvin.com/protocol/rfc2616.pdf
Hypertext Transfer Protocol -- HTTP 1.1

Protocol Name

S-HTTP: Secure Hypertext Transfer Protocol

Protocol Description

Secure HTTP (S-HTTP) is a secure message-oriented communications protocol designed for use in conjunction with HTTP. S-HTTP is designed to coexist with HTTP's messaging model and to be easily integrated with HTTP applications.

Secure HTTP provides a variety of security mechanisms to HTTP clients and servers, providing the security service options appropriate to the wide range of potential end uses possible for the World-Wide Web (WWW). S-HTTP provides symmetric capabilities to both client and server (in that equal treatment is given to both requests and replies, as well as for the preferences of both parties) while preserving the transaction model and implementation characteristics of HTTP.

Several cryptographic message format standards may be incorporated into S-HTTP clients and servers. S-HTTP supports interoperation among a variety of implementations and is compatible with HTTP. S-HTTP aware clients can communicate with S-HTTP oblivious servers and vice-versa, although such transactions obviously would not use S-HTTP security features.

S-HTTP does not require client-side public key certificates (or public keys), as it supports symmetric key-only operation modes. This is significant because it means that spontaneous private transactions can occur without requiring individual users to have an established public key. While S-HTTP is able to take advantage of ubiquitous certification infrastructures, its deployment does not require it.

S-HTTP supports end-to-end secure transactions. Clients may be "primed" to initiate a secure transaction (typically using information supplied in message headers); this may be used to support encryption of fill-out forms, for example. With S-HTTP, no sensitive data need ever be sent over the network in the clear.

S-HTTP provides full flexibility of cryptographic algorithms, modes and parameters. Option negotiation is used to allow clients and servers to agree on transaction modes, cryptographic algorithms (RSA vs. DSA for signing, DES vs. RC2 for encrypting, etc.) and certificate selection.

S-HTTP attempts to avoid presuming a particular trust model, although its designers admit to a conscious effort to facilitate multiply-rooted hierarchical trust, and anticipate that principals may have many public key certificates. S-HTTP differs from Digest-Authentication in that it provides support for public key cryptography and consequently digital signature capability, as well as providing confidentiality. Another popular technology for secured web communication is HTTPS, which is HTTP running on top of TLS and SSL for secured web transactions.

Protocol Structure

Syntactically, Secure HTTP messages are the same as HTTP, consisting of a request or status line followed by headers and a body. However, the range of headers is different and the bodies are typically cryptographically enhanced.

S-HTTP messages, just as HTTP messages, consist of requests from client to server and responses from server to client.

The request message has the following format:

Request Line	General header	Request header	Entity header	Message Body

In order to differentiate S-HTTP messages from HTTP messages and allow for special processing, the request line should use the special "Secure" method and use the protocol designator "Secure-HTTP/1.4". Consequently, Secure-HTTP and HTTP processing can be intermixed on the same TCP port, e.g. port 80. In order to prevent leakage of potentially sensitive information Request-URI should be "*".

S-HTTP responses should use the protocol designator "Secure-HTTP/1.4".

The response message has the following format:

Status Line	General header	Response header	Entity header	Message Body

Note that the status in the Secure HTTP response line does not indicate anything about the success or failure of the unwrapped HTTP request. Servers should always use 200 OK provided that the Secure HTTP processing is successful. This prevents analysis of success or failure for any request, which the correct recipient can determine from the encapsulated data. All case variations should be accepted.

For details of the S-HTTP messages, please check the reference documents.

Related protocols

WWW, FTP, STMP, NNTP, Gopher, WAIS, HTTP, DNS

Sponsor Source

S-HTTP is defined by IETF (http://www.ietf.org) in RFC 2660.

Reference

http://www.javvin.com/protocol/rfc2660.pdf
The Secure HyperText Transfer Protocol

Protocol Name

IMAP & IMAP4: Internet Message Access Protocol (version 4)

Protocol Description

Internet Message Access Protocol (IMAP) is a method of accessing electronic mail or bulletin board messages that are kept on a mail server. IMAP permits a “client” email program to access remote message stores as if they were local. Email stored on an IMAP server can be manipulated from a desktop computer remotely, without the need to transfer messages or files back and forth between these computers.

There are several different technologies and approaches to building a distributed electronic mail infrastructure: POP (Post Office Protocol), DMSP (Distributed Mail System Protocol) and IMAP (Internet Message Access Protocol) among them. Of the three, POP is the oldest and consequently the best known. DMSP is largely limited to a single application, PCMAIL, and is known primarily for its excellent support of “disconnected” operation. IMAP offers a superset of POP and DMSP capabilities, and provides good support for all three modes of remote mailbox access: offline, online, and disconnected.

In the online mode, the IMAP mail client does not copy mails in a shared server all at once and then delete them. It is an interactive client-server model, where the client can ask the server for headers or the bodies of specified messages, or to search for messages meeting certain criteria. Messages in the mail repository can be marked with various status flags (e.g. “deleted” or “answered”) and they stay in the repository until explicitly removed by the user. IMAP is designed to permit manipulation of remote mailboxes as if they were local. Depending on the IMAP client implementation and the mail architecture desired by the system manager, the user may save messages directly on the client machine or save them on the server, or be given the choice of doing either.

IMAP includes operations for creating, deleting and renaming mailboxes; checking for new messages; permanently removing messages; setting and clearing flags; server-based and MIME parsing, and searching; and selective fetching of message attributes, texts, and portions thereof for efficiency. IMAP allows clients to access messages (both new and saved) from more than one computer. This feature has become extremely important as reliance on electronic messaging and use of multiple computers has increased.

The current version of IMAP is version 4 revision 1(IMAP4 rev1). Key features for IMAP4 include:

- Fully compatible with Internet messaging standards, e.g. MIME.
- Allows message access and management from more than one computer.
- Allows access without reliance on less efficient file access protocols.
- Provides support for “online”, “offline”, and “disconnected” access modes.
- Supports concurrent access to shared mailboxes.
- Client software needs no knowledge about the server’s file store format.

Protocol Structure

IMAP key commands:

APPEND
AUTHENTICATE
CAPABILITY
CHECK
CLOSE
COPY
CREATE
DELETE
DELETEACL
EXAMINE
EXPUNGE
FETCH
GETACL
GETQUOTA
GETQUOTAROOT
LIST
LISTRIGHTS
LOGIN
LOGOUT
LSUB
MYRIGHTS
NOOP
RENAME
SEARCH
SELECT
SETACL
SETQUOTA
STARTTLS
STATUS
STORE
SUBSCRIBE
UID
UNSELECT
UNSUBSCRIBE
X<atom>

Related protocols

SMTP, TCP, POP, POP3, MIME, DMSP

Sponsor Source

IMAP is defined by IETF (http://www.ietf.org)

Reference

http://www.javvin.com/protocol/rfc3501.pdf
INTERNET MESSAGE ACCESS PROTOCOL - VERSION 4rev1

Protocol Name

IRCP: Internet Relay Chat Protocol

Protocol Description

Internet Relay Chat Protocol (IRCP), which is well-suited to running on many machines in a distributed fashion, enables teleconferencing on the Internet. The IRC protocol has been developed on systems using the TCP/IP network protocol, although there is no requirement that this remain the only environment in which it operates. The IRC protocol is a text-based protocol, with the simplest client being any socket program capable of connecting to the server.

A typical setup in IRCP involves a single process (the server) forming a central point for clients (or other servers) to connect to, performing the required message delivery/multiplexing and other functions. The server forms the backbone of IRC, providing a point to which clients may connect to talk to each other, and a point for other servers to connect to, forming an IRC network. The only network configuration allowed for IRC servers is that of a spanning tree where each server acts as a central node for the rest of the net it sees.

To allow a reasonable amount of order to be kept within the IRC network, a special class of clients (operators) is allowed to perform general maintenance functions on the network. Another concept in the IRCP is a channel, which is a named group of one or more clients which will all receive messages addressed to that channel.

IRCP allows communications between two clients, one to many(all) clients, client to server, and server to server. This protocol provides the technical foundation for most of the Internet instant message and chat systems.

Protocol Structure

IRCP is a text-based protocol with many commands. The key commands are:

User <username> <hostname> <servername> <realname>: is used at the beginning of connection to specify the username, hostname, servername and realname of a new user.
Pass <password>: is used to set a 'connection password'.
Nick <nickname> <hopcount>: is used to give user a nickname or change the previous one.
Server <servername> <hopcount> <info>: is used to tell a server that the other end of a new connection is a server
Oper <user> <password>: request to get operator privileges
Quit <quit message>: a client session is ended with a quit message.
Squit <server> <comment>: is needed to tell about quitting or dead servers.
Join <channel>: is used by client to start listening to a specific channel.
Topic <channel>: is used to change or view the topic of a channel.
Names <channel>: is used to list all nicknames that are visible to a user on any channel.
List <channel>: is used to list channels and their topics.
Kick <channel> <user> <comment>: can be used to forcibly remove a user from a channel.

Related protocols

IP, IPv6, TCP

Sponsor Source

IRCP is defined by IETF (http://www.ietf.org) in RFC 1459 and updated by RFC 2810, 2811, 2812, 2813.

Reference

http://www.javvin.com/protocol/rfc1459.pdf
Internet Relay Chat Protocol.

Protocol Name

LDAP: Lightweight Directory Access Protocol (version 3)

Protocol Description

Lightweight Directory Access Protocol (LDAP)is designed to provide access to the X.500 Directory while not incurring the resource requirements of the Directory Access Protocol (DAP). LDAP is specifically targeted at simple management applications and browser applications that provide simple read/write interactive access to the X.500 Directory, and is intended to be a complement to the DAP itself.

Key aspects of LDAP version 3 are:

- All protocol elements of LDAPv2 are supported.
- The protocol is carried directly over TCP or other transport, bypassing much of the session/presentation overhead of X.500 DAP.
- Most protocol data elements can be encoded as ordinary strings.
- Referrals to other servers may be returned.
- SASL mechanisms may be used with LDAP to provide association security services.
- Attribute values and Distinguished Names have been internationalized through the use of the ISO 10646 character set.
- The protocol can be extended to support new operations, and controls may be used to extend existing operations.
- The schema is published in the directory for use by clients.

The general model adopted by LDAP is one of clients performing protocol operations against servers. In this model, a client transmits a protocol request to a server, describing the operation to be performed. The server is then responsible for performing the necessary operation(s) in the directory. Upon completion of the operation(s), the server returns a response, containing any results or errors to the requesting client.

In LDAP versions 1 and 2, no provision was made for protocol servers returning referrals to clients. However, for improved performance and distribution LDAP v3 permits servers to return to clients referrals to other servers. This allows servers to offload the work of contacting other servers to progress operations.

Protocol Structure

LDAP messages are PDUs mapped directly onto the TCP byte stream and use port 389. The LDAP messages do not have their own header and are text messages based on ANS.1. For the purposes of protocol exchanges, all protocol operations are encapsulated in a common envelope, the LDAPMessage, The function of the LDAPMessage is to provide an envelope containing common fields required in all protocol exchanges. At this time, the only common fields are the message ID and the controls.

Related protocols

TCP, DAP

Sponsor Source

LDAP is defined by IETF (http://www.ietf.org) in RFC 2251, 2252, 2253, 2254, 2255, 2256, 2829, 2830 and 3377.

Reference

http://www.javvin.com/protocol/rfc2251.pdf
Lightweight Directory Access Protocol (v3) The specification of the LDAP on-the-wire protocol
http://www.javvin.com/protocol/rfc2252.pdf
Lightweight Directory Access Protocol (v3): Attribute Syntax Definitions
http://www.javvin.com/protocol/rfc2253.pdf
Lightweight Directory Access Protocol (v3): UTF-8 String Representation of Distinguished Names
http://www.javvin.com/protocol/rfc2254.pdf
The String Representation of LDAP Search Filters
http://www.javvin.com/protocol/rfc2255.pdf
The LDAP URL Format
http://www.javvin.com/protocol/rfc2256.pdf
A Summary of the X.500(96) User Schema for use with LDAPv3
http://www.javvin.com/protocol/rfc2829.pdf
Authentication Methods for LDAP
http://www.javvin.com/protocol/rfc2830.pdf
Lightweight Directory Access Protocol (v3): Extension for Transport Layer Security
http://www.javvin.com/protocol/rfc3377.pdf
Lightweight Directory Access Protocol (v3): Technical Specification

Protocol Name

MIME (S-MIME): Multipurpose Internet Mail Extensions and Secure MIME

Protocol Description

MIME, an acronym for Multipurpose Internet Mail Extensions, specifies how messages must be formatted so that they can be exchanged between different email systems. MIME is a very flexible format, permitting one to include virtually any type of file or document in an email message. MIME messages can contain text, images, audio, video, or other application-specific data. Specifically, MIME allows mail messages to contain:

- Multiple objects in a single message.
- Text having unlimited line length or overall length.
- Character sets other than ASCII, allowing non-English language messages.
- Multi-font messages.
- Binary or application-specific files.
- Images, Audio, Video and multi-media messages.

A MIME multipart message contains a boundary in the Content-type: header; this boundary, which must not occur in any of the parts, is placed between the parts, and at the beginning and end of the body of the message.

A secure version of MIME, S/MIME (Secure/Multipurpose Internet Mail Extensions), is defined to support encryption of email messages. Based on the MIME standard, S/MIME provides the following cryptographic security services for electronic messaging applications: authentication, message integrity and non-repudiation of origin and privacy and data security.

S/MIME can be used by traditional mail user agents (MUAs) to add cryptographic security services to mail that is sent, and to interpret cryptographic security services in mail that is received. However, S/MIME is not restricted to mail; it can be used with any transport mechanism that transports MIME data, such as HTTP. As such, S/MIME takes advantage of the object-based features of MIME and allows secure messages to be exchanged in mixed-transport systems.

Further, S/MIME can be used in automated message transfer agents that use cryptographic security services that do not require any human intervention, such as the signing of software-generated documents and the encryption of FAX messages sent over the Internet.

Protocol Structure

Definition of MIME header fields is as follows:

```
entity-headers := [ content CRLF ]
                  [ encoding CRLF ]
                  [ id CRLF ]
                  [ description CRLF ]
                  *( MIME-extension-field CRLF )

MIME-message-headers := entity-headers
                        fields
                        version CRLF
                        ; The ordering of the header
                        ; fields implied by this BNF
                        ; definition should be ignored.

MIME-part-headers := entity-headers
                     [ fields ]
                     ; Any field not beginning with
                     ; "content-" can have no defined
                     ; meaning and may be ignored.
                     ; The ordering of the header
                     ; fields implied by this BNF
                     ; definition should be ignored.
```

The message format and procedure of S/MIME can be found in the reference documents.

Related protocols

POP3, SMTP

Sponsor Source

MIME is defined by IETF (http://www.ietf.org) in RFC 2045, 2046, 2047, 2048, 2049. S/MIME version 3 is defined in RFC 2632, 2633 etc.

Reference

http://www.javvin.com/protocol/rfc2045.pdf
Multipurpose Internet Mail Extensions (MIME) Part One: Format of Internet Message Bodies
http://www.javvin.com/protocol/rfc2046.pdf
Multipurpose Internet Mail Extensions (MIME) Part Two: Media Types
http://www.javvin.com/protocol/rfc2047.pdf
MIME (Multipurpose Internet Mail Extensions) Part Three: Message Header Extensions for Non-ASCII Text
http://www.javvin.com/protocol/rfc2048.pdf
Multipurpose Internet Mail Extensions (MIME) Part Four: Registration Procedures.
http://www.javvin.com/protocol/rfc2049.pdf
Multipurpose Internet Mail Extensions
http://www.javvin.com/protocol/rfc2632.pdf
S/MIME Version 3 Certificate Handling
http://www.javvin.com/protocol/rfc2633.pdf
S/MIME Version 3 Message Specification

Protocol Name

NAT: Network Address Translation

Protocol Description

Basic Network Address Translation (Basic NAT) is a method by which IP addresses are mapped from one group to another, transparent to end users. Network Address Port Translation, or NAPT, is a method by which many network addresses and their TCP/UDP ports are translated into a single network address and its TCP/UDP ports. Together, these two operations, referred to as traditional NAT, provide a mechanism to connect a realm with private addresses to an external realm with globally unique registered addresses.

The need for IP Address translation arises when a network's internal IP addresses cannot be used outside the network either for privacy reasons or because they are invalid for use outside the network. Network topology outside a local domain can change in many ways. Customers may change providers, company backbones may be reorganized, or providers may merge or split. Whenever external topology changes with time, address assignment for nodes within the local domain must also change to reflect the external changes. Changes of this type can be hidden from users within the domain by centralizing changes to a single address translation router. Basic Address Translation allows hosts in a private network to transparently access the external network and enable access to selected local hosts from the outside. Organizations with a network setup predominantly for internal use and with a need for occasional external access are good candidates for this scheme.

There are limitations to using the translation method. It is mandatory that all requests and responses pertaining to a session be routed via the same NAT router. One way to ascertain this would be to have NAT based on a border router that is unique to a stub domain, where all IP packets either originated from the domain or are destined for the domain. There are other ways to ensure this with multiple NAT devices.

The NAT solution has the disadvantage of taking away the end-to-end significance of an IP address, and making up for this with an increased state in the network. As a result, with a NAT device enroute, end-to-end IP network level security assured by IPSec cannot be assumed to apply to end hosts. The advantage of this approach, however, is that it can be installed without changes to hosts or routers.

Protocol Structure

NAT is a procedure, not a structured protocol.

Related protocols

IP, IPv6, TCP, UDP, NATP

Sponsor Source

NAT is defined by IETF (http://www.ietf.org) in RFC 3022.

Reference

http://www.javvin.com/protocol/rfc3022.pdf
Traditional IP Network Address Translator (Traditional NAT)

Protocol Name

NNTP: Network News Transfer Protocol

Protocol Description

Network News Transfer Protocol (NNTP) specifies a protocol for the distribution, inquiry, retrieval, and posting of news articles using a reliable stream (such as TCP port 119) server-client model. NNTP is designed so that news articles need only be stored on one (presumably central) server host, and subscribers on other hosts attached to the network may read news articles using stream connections to the news host. The Network News Transfer Protocol (NNTP) established the technical foundation for the widely used Newsgroups.

NNTP is modeled after the USENET news system. However, NNTP makes few demands upon the structure, content or storage of news articles and thus it can easily be adapted to other non-USENET news systems. Using NNTP, hosts exchanging news articles have an interactive mechanism for deciding which articles are to be transmitted.

A host desiring new news, or which has new news to send, will typically contact one or more of its neighbors using NNTP. The client host will then inquire as to which new articles have arrived in all or some of the newsgroups that it desires to receive, using the NEWNEWS command. It will receive a list of new articles from the server, and can request transmission of those articles that it desires and does not already have. Finally, the client can advise the server of those new articles which the client has recently received. The server will indicate those articles that it has already obtained copies of and which articles should be sent to add to its collection. In this manner, only those articles which are not duplicates and which are desired are transferred.

Protocol Structure

NNTP uses commands and responses for communications. Commands consist of a command word, which in some cases may be followed by a parameter. NNTP has many commands.

The following are the key commands:

Article <message ID> Displays the header, a blank line, then the body (text) of the specified article.

Message-id Optional field, is the message id of an article as shown in that article's header. If it is blank, the current article is assumed.

Head Identical to the ARTICLE command except that it returns only the header lines.

Status Similar to the ARTICLE command except that no text is returned.

Group <ggg> The required parameter ggg is the name of the newsgroup to be selected. A list of valid newsgroups may be obtained from the LIST command. The successful selection response will return the article numbers of the first and last articles in the group, and an estimate of the number of articles on file in the group.

Body Identical to the ARTICLE command except that it returns only the text body of the article.

List Returns a list of valid newsgroups and associated information.

NewsGroups A list of newsgroups created since <date and time> will be listed in the same format as the LIST command.

NewNews A list of message-ids of articles posted to or received by the specified newsgroup since "date" will be listed.

Next The internally maintained "current article pointer" is advanced to the next article in the current newsgroup.

Post If posting is allowed, response code 340 is returned to indicate that the article to be posted should be sent.

Quit The server process acknowledges the QUIT command and then closes the connection to the client.

Related protocols

TCP

Sponsor Source

NNTP is defined by IETF (http://www.ietf.org) in RFC 977.

Reference

http://www.javvin.com/protocol/rfc977.pdf
Network News Transfer Protocol

Protocol Name

NTP: Network Time Protocol

Protocol Description

Network Time Protocol (NTP) is a time synchronization system for computer clocks through the Internet network. It provides the mechanisms to synchronize time and coordinate time distribution in a large, diverse internet operating at rates from mundane to light wave. It uses a returnable time design in which a distributed sub network of time servers, operating in a self-organizing, hierarchical master-slave configuration, synchronizes logical clocks within the sub network and to national time standards via wire or radio. The servers can also redistribute reference time via local routing algorithms and time daemons.

NTP is designed to produce three products: clock offset, roundtrip delay and dispersion, all of which are relative to a selected reference clock. Clock offset represents the amount to adjust the local clock to bring it into correspondence with the reference clock. Roundtrip delay provides the capability to launch a message to arrive at the reference clock at a specified time. Dispersion represents the maximum error of the local clock relative to the reference clock. Since most host time servers will synchronize via another peer time server, there are two components in each of these three products, those determined by the peer relative to the primary reference source of standard time and those measured by the host relative to the peer. Each of these components is maintained separately in the protocol in order to facilitate error control and management of the subnet itself. They provide not only precision measurements of offset and delay, but also definitive maximum error bounds, so that the user interface can determine not only the time, but the quality of the time as well.

NTP evolved from the Time Protocol and the ICMP Timestamp message but is specifically designed to maintain accuracy and robustness, even when used over typical Internet paths involving multiple gateways, highly dispersive delays and unreliable nets. NTP version 3 is the current version but previous superseded versions are compatible.

Protocol Structure

2	5	8	16	24	32bit
LI	VN	Mode	Stratum	Poll	Precision
Root Delay					
Root Dispersion					
Reference Identifier					
Reference timestamp (64)					
Originate Timestamp (64)					
Receive Timestamp (64)					
Transmit Timestamp (64)					
Key Identifier (optional) (32)					
Message digest (optional) (128)					

LI Leap Indicator warning of impending leap-second to be inserted at the end of the last day of the current month.

VN Version number indicating the version number.

Mode The mode: This field can contain the following values:

0	Reserved.
1	Symmetric active.
3	Client.
4	Server.
5	Broadcast.
6	NTP control message.

Stratum An integer identifying the stratum level of the local clock.

Poll Signed integer indicating the maximum interval between successive messages, in seconds to the nearest power of 2.

Precision
Signed integer indicating the precision of the local clock, in seconds to the nearest power of 2.

Root Delay
Signed fixed-point number indicating the total roundtrip delay to the primary reference source, in seconds with fraction point between bits 15 and 16.

Root Dispersion
Unsigned fixed-point number indicating the nominal error relative to the primary reference source, in seconds with fraction point between bits 15 and 16.

Reference Identifier
Identifying the particular reference source.

Originate Timestamp
This is the time at which the request departed the client for the server, in 64-bit timestamp format.

Receive Timestamp
This is the time at which the request arrived at the server, in 64-bit timestamp format.

Transmit Timestamp
This is the time at which the reply departed the server for the client, in 64-bit timestamp format.

Authenticator (optional)
When the NTP authentication scheme is implemented, the Key Identifier and Message Digest fields contain the message authentication code (MAC) information defined.

Related protocols

ICMP, SNTP

Sponsor Source

NTP is defined by IETF (http://www.ietf.org) in RFC 1305.

Reference

http://www.javvin.com/protocol/rfc1305.pdf
Network Time Protocol (Version 3) Specification, Implementation.

Protocol Name

POP and POP3: Post Office Protocol (version 3)

Protocol Description

The Post Office Protocol is designed to allow a workstation to dynamically access a mail drop on a server host. POP3 is the version 3 (the latest version) of the Post Office Protocol. POP3 allows a workstation to retrieve mail that the server is holding for it. POP3 transmissions appear as data messages between stations. The messages are either command or reply messages.

There are several different technologies and approaches to building a distributed electronic mail infrastructure: POP (Post Office Protocol), DMSP (Distributed Mail System Protocol), and IMAP (Internet Message Access Protocol) among them. Of the three, POP is the oldest and consequently the best known. DMSP is largely limited to a single application, PCMAIL, and is known primarily for its excellent support of “disconnected” operation. IMAP offers a superset of POP and DMSP capabilities, and provides good support for all three modes of remote mailbox access: offline, online, and disconnected.

POP was designed to support “offline” mail processing, in which mail is delivered to a server, and a personal computer user periodically invokes a mail “client” program that connects to the server and downloads all of the pending mail to the user’s own machine. The offline access mode is a kind of store-and-forward service, intended to move mail (on demand) from the mail server (drop point) to a single destination machine, usually a PC or Mac. Once delivered to the PC or Mac, the messages are then deleted from the mail server.

POP3 is not designed to provide extensive manipulation operations of mail on the server; which are done by a more advanced (and complex) protocol IMAP4. POP3 uses TCP as the transport protocol.

Protocol Structure

POP3 messages are ASCII messages sent between client and servers. POP3 Command Summary:

Commands	Description
USER	Name of user
PASS	User’s password
STAT	Information on messages in the server
RETR	Number of message to get
DELE	Number of message to delete
LIST	Number of message to show
TOP <messageID> <nombredelignes>	Print X lines of the message starting from the beginning (header included)
QUIT	Exit to POP3’s server

Optional POP3 Commands:

APOP name digest valid in the AUTHORIZATION state
TOP msg n valid in the TRANSACTION state
UIDL [msg]

POP3 Replies:

+OK
-ERR

Related protocols

SMTP, IMAP4, TCP, POP

Sponsor Source

POP3 is defined by IETF (http://www.ietf.org) in RFC 1939.

Reference

http://www.javvin.com/protocol/rfc1939.pdf
Post Office Protocol - Version 3

Protocol Name

rlogin: Remote Login in UNIX Systems

Protocol Description

rlogin (remote login) is a UNIX command that allows an authorized user to login to other UNIX machines (hosts) on a network and to interact as if the user were physically at the host computer. Once logged in to the host, the user can do anything that the host has given permission for, such as read, edit, or delete files.

Each remote machine may have a file named /etc/hosts.equiv containing a list of trusted hostnames with which it shares usernames. Users with the same username on both the local and remote machine may rlogin from the machines listed in the remote machine's /etc/hosts.equiv file without supplying a password. Individual users may set up a similar private equivalence list with the file .rhosts in their home directories. Each line in this file contains two names: a host name and a username separated by a space. An entry in a remote user's .rhosts file permits the user named username who is logged into hostname to log in to the remote machine as the remote user without supplying a password. If the name of the local host is not found in the /etc/hosts.equiv file on the remote machine and the local username and hostname are not found in the remote user's .rhosts file, then the remote machine will prompt for a password. Hostnames listed in /etc/hosts.equiv and .rhosts files must be the official hostnames listed in the host's database; nicknames may not be used in either of these files. For security reasons, the .rhosts file must be owned by either the remote user or by root.

The remote terminal type is the same as your local terminal type (as given in your environment TERM variable). The terminal or window size is also copied to the remote system if the server supports the option, and changes in size are reflected as well. All echoing takes place at the remote site, so that (except for delays) the remote login is transparent. Flow control using <CTRL-S> and <CTRL-Q> and flushing of input and output on interrupts are handled properly.

A secure version of rlogin (slogin) was combined with two other UNIX utilities, ssh and scp, in the Secure Shell suite, an interface and protocol created to replace the earlier utilities.

Protocol Structure

rlogin command is:

rlogin [-8EL] [-ec] [-l username] hostname

OPTION Flags

-8EL	Allows an 8-bit data path at all times. Otherwise, unless the start and stop characters on the remote host are not Ctrl-S and Ctrl-Q, the rlogin command uses a 7-bit data path and parity bits are stripped.
-e Character	Changes the escape character. Substitute the character you choose for Character.
-f	Causes the credentials to be forwarded. This flag will be ignored if Kerberos 5 is not the current authentication method. Authentication will fail if the current DCE credentials are not marked forwardable.
-F	Causes the credentials to be forwarded. In addition, the credentials on the remote system will be marked forwardable (allowing them to be passed to another remote system). This flag will be ignored if Kerberos 5 is not the current authentication method. Authentication will fail if the current DCE credentials are not marked forwardable.
-k realm	Allows the user to specify the realm of the remote station if it is different from the local systems realm. For these purposes, a realm is synonymous with a DCE cell. This flag will be ignored if Kerberos 5 is not the current authentication method.
-l User	Changes the remote user name to the one you specify. Otherwise, your local user name is used at the remote host.

Hostname — The remote machine on which rlogin establishes the remote login session.

Related protocols

FTP, TELNET

Sponsor Source

rlogin is a UNIX command.

Reference

http://www.javvin.com/protocol/rfc1282.pdf
BSD Rlogin

Protocol Name

RMON: Remote Monitoring MIBs (RMON1 and RMON2)

Protocol Description

Remote Monitoring (RMON) is a standard monitoring specification that enables various network monitors and console systems to exchange network-monitoring data. RMON provides network administrators with more freedom in selecting network-monitoring probes and consoles with features that meet their particular networking needs.

RMON was originally developed to address the problem of managing LAN segments and remote sites from a central location. The RMON is an extension of the SNMP MIB. Within an RMON network monitoring data is defined by a set of statistics and functions and exchanged between various different monitors and console systems. Resultant data is used to monitor network utilization for network planning and performance-tuning, as well as assisting in network fault diagnosis.

There are 2 versions of RMON: RMONv1 and RMONv2. RMONv1, which can now be found on most modern network hardware, defined 9 MIB groups for basic network monitoring. RMON2 is an extension of RMON that focuses on higher layers of traffic above the medium access-control(MAC) layer. RMON2 has an emphasis on IP traffic and application-level traffic. RMON2 allows network management applications to monitor packets on all network layers. This is different from RMONv1, which only allows network monitoring at MAC layer or below.

RMON solutions are comprised of two components: a probe (or an agent or a monitor), and a management station. Agents store network information within their RMON MIB and are normally found as embedded software on network hardware such as routers and switches although they can be a program running on a PC. Agents can only see the traffic that flows through them so they must be placed on each LAN segment or WAN link that is to be monitored. Clients, or management stations, communicate with the RMON agent or probe, using SNMP to obtain and correlate RMON data.

There are a number of variations to the RMON MIB. For example, the Token Ring RMON MIB provides objects specific to managing Token Ring networks. The SMON MIB extends RMON by providing RMON analysis for switched networks.

Protocol Structure

The monitoring focus of RMON1 and RMON 2 in the network layers:

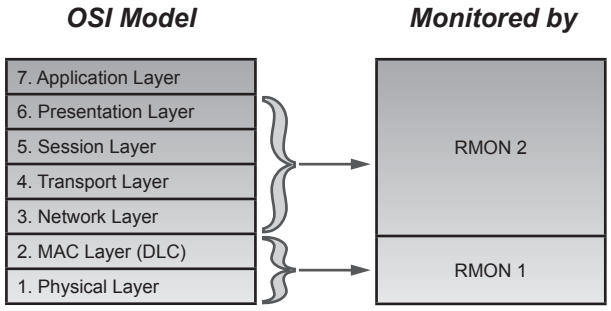


Figure 2-1: RMON Monitoring Layers

RMON 1 MIB Group	Function	Elements
Statistics	Contains statistics measured by the probe for each monitored interface on this device.	Packets dropped, packets sent, bytes sent (octets), broadcast packets, multicast packets, CRC errors, runts, giants, fragments, jabbers, collisions, and counters for packets ranging from 64 to 128, 128 to 256, 256 to 512, 512 to 1024, and 1024 to 1518 bytes.
History	Records periodic statistical samples from a network and stores for retrieval.	Sample period, number of samples, items sampled.
Alarm	Periodically takes statistical samples and compares them with set thresholds for events generation.	Includes the alarm table and requires the implementation of the event group. Alarm type, interval, starting threshold, stop threshold.
Host	Contains statistics associated with each host discovered on the network.	Host address, packets, bytes received and transmitted, as well as broadcast, multicast, and error packets.
HostTopN	Prepares tables that describe the top hosts.	Statistics, host(s), sample start and stop periods, rate base, duration.
Matrix	Stores and retrieves statistics for conversations between sets of two addresses.	Source and destination address pairs and packets, bytes, and errors for each pair.
Filters	Enables packets to be matched by a filter equation for capturing or events.	Bit-filter type (mask or not mask), filter expression (bit level), conditional expression (and, or not) to other filters.
Packet Capture	Enables packets to be captured after they flow through a channel.	Size of buffer for captured packets, full status (alarm), number of captured packets.
Events	Controls the generation and notification of events from this device.	Event type, description, last time event sent
Token Ring	Support of Token Ring	(not used often)

RMON 2 MIB Group	Functions
Protocol Directory	The Protocol Directory is a simple and interoperable way for an RMON2 application to establish which protocols a particular RMON2 agent implements. This is especially important when the application and the agent are from different vendors
Protocol Distribution	Mapping the data collected by a probe to the correct protocol name that can then be displayed to the network manager.
Address mapping	Address translation between MAC-layer addresses and network-layer addresses which are much easier to read and remember. Address translation not only helps the network manager, it supports the SNMP management platform and will lead to improved topology maps.
Network Layer host	Network host (IP layer) statistics
Network layer matrix	Stores and retrieves network layer (IP layer) statistics for conversations between sets of two addresses.
Application layer host	Application host statistic
Application layer matrix	Stores and retrieves application layer statistics for conversations between sets of two addresses.
User history	This feature enables the network manager to configure history studies of any counter in the system, such as a specific history on a particular file server or a router-to-router connection
Probe configuration	This RMON2 feature enables one vendor's RMON application to remotely configure another vendor's RMON probe.

Related protocols

SNMP, SMI

Sponsor Source

RMON is defined by IETF (http://www.ietf.org) with a group of RFCs shown in the reference links.

Reference

http://www.javvin.com/protocol/rfc2819.pdf
Remote Network Monitoring Management Information Base
http://www.javvin.com/protocol/rfc2021.pdf
Remote Network Monitoring Management Information Base Version 2 using SMIv2
http://www.javvin.com/protocol/rfc1157.pdf
A Simple Network Management Protocol

Protocol Name

SLP: Service Location Protocol

Protocol Description

The Service Location Protocol (SLP) provides a scalable framework for the discovery and selection of network services. Using this protocol, computers using the Internet no longer need so much static configuration for network services for network-based applications. This is especially important as computers become more portable and users less tolerant or less able to fulfill the demands of network system administration.

Traditionally, users find services by using the name of a network host (a human readable text string), which is an alias for a network address. SLP (Service Location Protocol) eliminates the need for a user to know the name of a network host supporting a service. Rather, the user names the service and supplies a set of attributes, which describe the service. SLP (Service Location Protocol) allows the user to bind this description to the network address of the service.

SLP (Service Location Protocol) provides a dynamic configuration mechanism for applications in local area networks. It is not a global resolution system for the entire Internet; rather it is intended to serve enterprise networks with shared services. Applications are modeled as clients that need to find servers attached to the enterprise network at a possibly distant location. For cases where there are many different clients and/or services available, the protocol is adapted to make use of nearby Directory Agents that offer a centralized repository for advertised services.

The basic operation in SLP is that a client attempts to discover the location for a service. In small installations, each service is configured to respond individually to each client. In larger installations, each service will register its service with one or more directory agents and clients contact the directory agent to fulfill a request for service location information. This is intended to be similar to URL specifications and make use of URL technology.

Protocol Structure

Service Location Protocol Header

8	16	32bit
Version	Function	Length
O M U A F rsvd	Dialect	Language Code
Char encoding		XID

Version The current version is version 1

Function The function field describes the operation of the Service location datagram. The following message types exist:

Function Value	*Message Type*	*Abbreviation*
1	Service Request	SrvReq
2	Service Reply	SrvRply
3	Service Registration	SrvReg
4	Service Deregister	SrvDereg
5	Service Acknowledge	SrvAck
6	Attribute Request	AttrRgst
7	Attribute Reply	AttrRply
8	DA Advertisement	DAADvert
9	Service Type Request	SrvTypeRqst
10	Service Type Reply	SrvTypeRply

Length Number of bytes in the message including the Service location header.

O The overflow bit.

M The monolingual bit.

U RL Authentication bit present.

A Attribute authentication bit present.

F If the F bit is set in a Service Acknowledgement, the directory agent has registered the service as a new entry.

Rsvd These bits are reserved and must have a value of 0.

Dialect To be use by future versions of the SLP. Must be set to zero.

Language Code The language encoded in this field indicates the language in which the remainder of the message should be interpreted.

Character Encoding The characters making up strings within the remainder of this message may be encoded in any standardized encoding

XID Transaction Identifier. Allows matching replies to individual requests.

Related protocols

TCP, UDP, DHCP

Sponsor Source

SLP is defined by IETF (http://www.ietf.org) in RFC 2165.

Reference

http://www.javvin.com/protocol/rfc2165.pdf
Service Location Protocol

Protocol Name

SMTP: Simple Mail Transfer Protocol

Protocol Description

Simple Mail Transfer Protocol (SMTP) is a protocol designed to transfer electronic mail reliably and efficiently. SMTP is a mail service modeled on the FTP file transfer service. SMTP transfers mail messages between systems and provides notification regarding incoming mail.

SMTP is independent of the particular transmission subsystem and requires only a reliable ordered data stream channel. An important feature of SMTP is its capability to transport mail across networks, usually referred to as “SMTP mail relaying”. A network consists of the mutually-TCP-accessible hosts on the public Internet, the mutually-TCP-accessible hosts on a firewall-isolated TCP/IP Intranet, or hosts in some other LAN or WAN environment utilizing a non-TCP transport-level protocol. Using SMTP, a process can transfer mail to another process on the same network or to some other network via a relay or gateway process accessible to both networks.

In this way, a mail message may pass through a number of intermediate relay or gateway hosts on its path from sender to ultimate recipient. The Mail eXchanger mechanisms of the domain name system are used to identify the appropriate next-hop destination for a message being transported.

Protocol Structure

SMTP commands are ASCII messages sent between SMTP hosts. Possible commands are as follows:

Command	Description
DATA	Begins message composition.
EXPN <string>	Returns names on the specified mail list.
HELO <domain>	Returns identity of mail server.
HELP <command>	Returns information on the specified command.
MAIL FROM <host>	Initiates a mail session from host.
NOOP	Causes no action, except acknowledgement from server.
QUIT	Terminates the mail session.
RCPT TO <user>	Designates who receives mail.
RSET	Resets mail connection.
SAML FROM <host>	Sends mail to user terminal and mailbox.
SEND FROM <host>	Sends mail to user terminal.
SOML FROM <host>	Sends mail to user terminal or mailbox.
TURN	Switches role of receiver and sender.
VRFY <user>	Verifies the identity of a user.

Related protocols

POP3, IMAP4, TCP, POP, FTP

Sponsor Source

SMTP is defined by IETF (http://www.ietf.org) in RFC 2821.

Reference

http://www.javvin.com/protocol/rfc2821.pdf
Simple Mail Transfer Protocol

Protocol Name

SNMP: Simple Network Management Protocol

Protocol Description

SNMP, an application layer protocol, is the standard protocol developed to manage nodes (servers, workstations, routers, switches and hubs, etc.) on an IP network. SNMP enables network administrators to manage network performance, find and solve network problems, and plan for network growth. Network management systems learn of problems by receiving traps or change notices from network devices implementing SNMP.

An SNMP managed network consists of three key components: managed devices, agents, and network-management systems (NMSs). A managed device is a network node that contains an SNMP agent and that resides on a managed network. Managed devices collect and store management information and make this information available to NMSs using SNMP. Managed devices, sometimes called network elements, can be routers and access servers, switches and bridges, hubs, computer hosts, or printers. An agent is a network management software module that resides in a managed device. An agent has local knowledge of management information and translates that information into a form compatible with SNMP. An NMS executes applications that monitor and control managed devices. NMSs provide the bulk of the processing and memory resources required for network management. One or more NMSs must exist on any managed network.

Currently, there are three versions of SNMP defined: SNMP v1, SNMP v2 and SNMP v3. Both versions 1 and 2 have a number of features in common, but SNMPv2 offers enhancements, such as additional protocol operations. SNMP Version 3 (SNMPv3) adds security and remote configuration capabilities to the previous versions. To solve the incompatible issues among different versions of SNMP, RFC 3584 defines the coexistence strategies.

SNMP also includes a group of extensions as defined by RMON, RMON 2, MIB, MIB2, SMI, OIDs, and Enterprise OIDs.

Protocol Structure

SNMP is an application protocol, which is encapsulated in UDP. The general SNMP message format for all versions is shown below:

Version	Community	PDU

- Version -- SNMP version number. Both the manager and agent must use the same version of SNMP. Messages containing different version numbers are discarded without further processing.
- Community -- Community name used for authenticating the manager before allowing access to the agent.
- PDU (Protocol Data Unit) -- The PDU types and formats for SNMPv1, v2 and v3 will be explained in the corresponding sections.

Related protocols

SNMPv1, SNMPv2, SNMPv3, UDP, RMON, SMI, OIDs

Sponsor Source

SNMP is defined by IETF (http://www.ietf.org) with a group of RFCs shown in the reference links.

Reference

http://www.javvin.com/protocol/rfc1155.pdf
Structure and Identification of Management Information for TCP/IP based internets
http://www.javvin.com/protocol/rfc1156.pdf
Management Information Base Network
http://www.javvin.com/protocol/rfc1157.pdf
A Simple Network Management Protocol
http://www.javvin.com/protocol/rfc1441.pdf
Introduction to SNMPv2
http://www.javvin.com/protocol/rfc2579.pdf
Textual Conventions for SNMPv2
http://www.javvin.com/protocol/rfc2580.pdf
Conformance Statements for SNMPv2
http://www.javvin.com/protocol/rfc2578.pdf
Structure of Management Information for SNMPv2
http://www.javvin.com/protocol/rfc3416.pdf
Protocol Operations for SNMPv2
http://www.javvin.com/protocol/rfc3417.pdf
Transport Mappings for SNMPv2
http://www.javvin.com/protocol/rfc3418.pdf
Management Information Base for SNMPv2
http://www.javvin.com/protocol/rfc3410.pdf
Introduction and Applicability Statements for Internet Standard Management Framework
http://www.javvin.com/protocol/rfc3411.pdf
Architecture for Describing SNMP Frameworks
http://www.javvin.com/protocol/rfc3412.pdf
Message Processing and Dispatching for the SNMP
http://www.javvin.com/protocol/rfc3413.pdf
SNMP Applications
http://www.javvin.com/protocol/rfc3414.pdf
User-based Security Model (USM) for SNMPv3
http://www.javvin.com/protocol/rfc3415.pdf
View-based Access Control Model for the SNMP
http://www.javvin.com/protocol/rfc3584.pdf
Coexistence between SNMP v1, v2 and v3

Protocol Name

SNMPv1: Simple Network Management Protocol version one

Protocol Description

SNMP is the protocol developed to manage nodes (servers, workstations, routers, switches and hubs etc.) on an IP network. SNMP enables network administrators to manage network performance, find and solve network problems and plan for network growth. Network management systems learn of problems by receiving traps or change notices from network devices implementing SNMP.

Currently, there are three versions of SNMP defined: SNMP v1, SNMP v2 and SNMP v3. In this document, we provide information primarily for SNMPv1. SNMPv1 is a simple request/response protocol. The network-management system issues a request, and managed devices return responses. This behavior is implemented by using one of four protocol operations: Get, GetNext, Set, and Trap. The Get operation is used by the NMS to retrieve the value of one or more object instances from an agent. If the agent responding to the Get operation cannot provide values for all the object instances in a list, it does not provide any values. The GetNext operation is used by the NMS to retrieve the value of the next object instance in a table or a list within an agent. The Set operation is used by the NMS to set the values of object instances within an agent. The Trap operation is used by agents to asynchronously inform the NMS of a significant event.

For information on SNMP, SNMPv2 and SNMPv3, please check the corresponding pages.

Protocol Structure

SNMP is an application protocol, which is encapsulated in UDP. The general SNMP message format for all versions is shown below:

Version	Community	PDU

- Version -- SNMP version number. Both the manager and agent must use the same version of SNMP. Messages containing different version numbers are discarded without further processing.
- Community -- Community name used for authenticating the manager before allowing access to the agent.
- PDU for SNMPv1 -- There are five different PDU types: GetRequest, GetNextRequest, GetResponse, SetRequest, and Trap. A general description of each of these is given in the next section.

The format for GetRequest, GetNext Request, GetResponse and SetRequest PDUs is shown here.

PDU type	Request ID	Error status	Error index	Object 1, value 1	Object 2, value 2	…

- PDU type—Specifies the type of PDU transmitted: 0 GetRequest, 1 GetNextRequest, 2 GetResponse and 3 SetRequest.
- Request ID—Associates SNMP requests with responses.
- Error status—Indicates one of a number of errors and error types. Only the response operation sets this field. Other operations set this field to zero.
- Error index—Associates an error with a particular object instance. Only the response operation sets this field. Other operations set this field to zero.
- Variable bindings—Serves as the data field of the SNMPv1 PDU. Each variable binding associates a particular object instance with its current value (with the exception of Get and GetNext requests, for which the value is ignored).

The format of the Trap PDU is shown below:

PDU type	Enterp	Agent Addr	Gen Trap	Spec Trap	Time Stamp	Obj 1, Val 1	Obj 1, Val 1	…

- PDU type --Specifies the type of PDU (4=Trap).
- Enterprise -- Identifies the management enterprise under whose registration authority the trap was defined.
- Agent address- - IP address of the agent, used for further identification.
- Generic trap type -- Field describing the event being reported. The following seven values are defined:
- Specific trap type -- Used to identify a non-generic trap when the Generic Trap Type is enterprise specific.
- Timestamp -- Value of the sysUpTime object, representing the amount of time elapsed between the last (re-)initialization and the generation of that Trap.

Related protocols

SNMPv1, SNMPv2, SNMPv3, UDP, RMON, SMI, OIDs

Sponsor Source

SNMPv1 is defined by IETF (http://www.ietf.org) in RFC 1157 plus a few supporting RFCs shown in the reference links.

Reference

http://www.javvin.com/protocol/rfc1157.pdf
A Simple Network Management Protocol
http://www.javvin.com/protocol/rfc1155.pdf
Structure and Identification of Management Information for TCP/IP based internets
http://www.javvin.com/protocol/rfc1156.pdf
Management Information Base Network

Protocol Name

SNMPv2: Simple Network Management Protocol version 2

Protocol Description

SNMP is the protocol developed to manage nodes (servers, workstations, routers, switches and hubs etc.) on an IP network. SNMP enables network administrators to manage network performance, find and solve network problems, and plan for network growth. Network management systems learn of problems by receiving traps or change notices from network devices implementing SNMP.

Currently, there are three versions of SNMP defined: SNMP v1, SNMP v2 and SNMP v3. In this document, we provide information primarily for SNMPv2. SNMP version 2 (SNMPv2) is an evolution of SNMPv1. The Get, GetNext, and Set operations used in SNMPv1 are exactly the same as those used in SNMPv2. However, SNMPv2 adds and enhances some protocol operations. The SNMPv2 Trap operation, for example, serves the same function as that used in SNMPv1 but uses a different message format and is designed to replace the SNMPv1 Trap.

SNMPv2 also defines two new operations: GetBulk and Inform. The GetBulk operation is used by the NMS to efficiently retrieve large blocks of data, such as multiple rows in a table. GetBulk fills a response message with as much of the requested data as will fit. The Inform operation allows one NMS to send trap information to another NMS and to then receive a response. In SNMPv2, if the agent responding to GetBulk operations cannot provide values for all the variables in a list, it provides partial results.

For information on SNMP, SNMPv1 and SNMPv3, please check the corresponding pages.

Protocol Structure

SNMP is an application protocol, which is encapsulated in UDP. The general SNMP message format for all versions is shown below:

Version	Community	PDU

- Version -- SNMP version number. Both the manager and agent must use the same version of SNMP. Messages containing different version numbers are discarded without further processing.
- Community -- Community name used for authenticating the manager before allowing access to the agent.
- PDU (Protocol Data Unit) - The PDU types and formats are different for SNMPv1, v2 and v3, which will be explained in the corresponding sections.

For SNMPv2, Get, GetNext, Inform, Response, Set, and Trap PDUs have the following format:

PDU type	Request ID	Error status	Error index	Object 1, value 1	Object 2, value 2	…

- PDU type—Identifies the type of PDU transmitted (Get, GetNext, Inform, Response, Set, or Trap).
- Request ID—Associates SNMP requests with responses.
- Error status—Indicates one of a number of errors and error types. Only the response operation sets this field. Other operations set this field to zero.
- Error index—Associates an error with a particular object instance. Only the response operation sets this field. Other operations set this field to zero.
- Variable bindings—Serves as the data field (value 1, value 2…) of the SNMPv2 PDU. Each variable binding associates a particular object instance with its current value (with the exception of Get and GetNext requests, for which the value is ignored).

SNMPv2 GetBulk PDU Format

PDU type	Request ID	Non repeaters	Max repetitions	Obj 1, Val 1	Obj 1, Val 1	…

- PDU type—Identifies the PDU as a GetBulk operation.
- Request ID—Associates SNMP requests with responses.
- Non repeaters—Specifies the number of object instances in the variable bindings field that should be retrieved no more than once from the beginning of the request. This field is used when some of the instances are scalar objects with only one variable.
- Max repetitions—Defines the maximum number of times that other variables beyond those specified by the Non repeaters field should be retrieved.
- Variable bindings—Serves as the data field (Obj 1, Obj 2 …) of the SNMPv2 PDU. Each variable binding associates a particular object instance with its current value (with the exception of Get and GetNext requests, for which the value is ignored).

Related protocols

SNMPv1, SNMPv2, SNMPv3, UDP, RMON, SMI, OIDs

Sponsor Source

SNMPv2 is defined by IETF (http://www.ietf.org) in RFC 1441 originally plus by a group of supporting and updating RFCs

shown in the list below.

Reference

http://www.javvin.com/protocol/rfc1441.pdf
Introduction to SNMPv2
http://www.javvin.com/protocol/rfc2579.pdf
Textual Conventions for SNMPv2
http://www.javvin.com/protocol/rfc2580.pdf
Conformance Statements for SNMPv2
http://www.javvin.com/protocol/rfc2578.pdf
Structure of Management Information for SNMPv2
http://www.javvin.com/protocol/rfc3416.pdf
Protocol Operations for SNMPv2
http://www.javvin.com/protocol/rfc3417.pdf
Transport Mappings for SNMPv2
http://www.javvin.com/protocol/rfc3418.pdf
Management Information Base for SNMPv2

Protocol Name

SNMPv3: Simple Network Management Protocol version three

Protocol Description

SNMP is the protocol developed to manage nodes (servers, workstations, routers, switches and hubs etc.) on an IP network. SNMP enables network administrators to manage network performance, find and solve network problems and plan for network growth. Network management systems learn of problems by receiving traps or change notices from network devices implementing SNMP. Currently, there are three versions of SNMP defined: SNMP v1, SNMP v2 and SNMP v3. In this document, we provide information primarily for SNMPv3.

SNMP Version 3 (SNMPv3) adds security and remote configuration capabilities to the previous versions. The SNMPv3 architecture introduces the User-based Security Model (USM) for message security and the View-based Access Control Model (VACM) for access control. The architecture supports the concurrent use of different security, access control and message processing models. More specifically:

Security
authentication and privacy
authorization and access control
Administrative Framework
naming of entities
people and policies
usernames and key management
notification destinations
proxy relationships
remotely configurable via SNMP operations

SNMPv3 also introduces the ability to dynamically configure the SNMP agent using SNMP SET commands against the MIB objects that represent the agent's configuration. This dynamic configuration support enables addition, deletion, and modification of configuration entries either locally or remotely.

For information on SNMP, SNMPv1 and SNMPv2, please check the corresponding pages.

Protocol Structure

SNMPv3 message format:

Msg Processed by MPM (Msg Processing Model)					
Version	ID	Msg Size	Msg Flag	Security Model	
Msg Processed by USM (User Security Module)					
Authoritative Engin ID	Authoritative Boots	Authoritative Engine Time	User name	Authentication parameters	Privacy Parameter
Scoped PDU					
Context engine ID	Context name	PDU			

- Version -- snmv3(3).
- ID -- A unique identifier used between two SNMP entities to coordinate request and response messages
- Msg Size -- Maximum size of a message in octets supported by the sender of the message
- Msg Flags -- An octet string containing three flags in the least significant three bits: reportableFlag, privFlag, authFlag.
- Security Model -- An identifier to indicate which security model was used by the sender and therefore which security model must be used by the receiver to process this message.
- AuthoritativeEngineID -- The snmpEngineID of the authoritative SNMP engine involved in the exchange of this message. Thus, this value refers to the source for a Trap, Response, or Report, and to the destination for a Get, GetNext, GetBulk, Set, or Inform.
- AuthoritativeEngineBoots -- The snmpEngineBoots value of the authoritative SNMP engine involved in the exchange of this message.
- AuthoritativeEngineTime -- The snmpEngineTime value of the authoritative SNMP engine involved in the exchange of this message.
- User Name --The user (principal) on whose behalf the message is being exchanged.
- AuthenticationParameters -- Null if authentication is not being used for this exchange. Otherwise, this is an authentication parameter.
- PrivacyParameters -- Null if privacy is not being used for this exchange. Otherwise, this is a privacy parameter.
- PDU (Protocol Data Unit) -- The PDU types for SNMPv3 are the same as for SNMPv2.

Related protocols

SNMPv1, SNMPv2, SNMPv3, UDP, RMON, SMI, OIDs

Sponsor Source

SNMPv3 is defined by IETF (http://www.ietf.org) in RFC 3411 plus a group of supporting RFCs shown in the reference links.

Reference

http://www.javvin.com/protocol/rfc3410.pdf
Introduction and Applicability Statements for Internet Standard Management Framework
http://www.javvin.com/protocol/rfc3411.pdf
Architecture for Describing SNMP Frameworks
http://www.javvin.com/protocol/rfc3412.pdf
Message Processing and Dispatching for the SNMP
http://www.javvin.com/protocol/rfc3413.pdf
SNMP Applications
http://www.javvin.com/protocol/rfc3414.pdf
User-based Security Model (USM) for SNMPv3
http://www.javvin.com/protocol/rfc3415.pdf
View-based Access Control Model for the SNMP
http://www.javvin.com/protocol/rfc3584.pdf
Coexistence between SNMP v1, v2 and v3

Protocol Name

SNTP: Simple Network Time Protocol

Protocol Description

The Simple Network Time Protocol (SNTP) Version 4 is an adaptation of the Network Time Protocol (NTP) used to synchronize computer clocks in the Internet. SNTP can be used when the ultimate performance of the full NTP implementation is not needed or justified. When operating with current and previous NTP and SNTP versions, SNTP Version 4 involves no changes to the NTP specification or known implementations, but rather a clarification of certain design features of NTP which allow operation in a simple, stateless remote-procedure call (RPC) mode with accuracy and reliability expectations similar to the UDP/TIME protocol.

It is strongly recommended that SNTP be used only at the extremities of the synchronization subnet. SNTP clients should operate only at the leaves (highest stratum) of the subnet and in configurations where no NTP or SNTP client is dependent on another SNTP client for synchronization. SNTP servers should operate only at the root (stratum 1) of the subnet and then only in configurations where no other source of synchronization other than a reliable radio or modem time service is available. The full degree of reliability ordinarily expected of primary servers is possible only using the redundant sources, diverse subnet paths and crafted algorithms of a full NTP implementation. This extends to the primary source of synchronization itself in the form of multiple radio or modem sources and backup paths to other primary servers should all sources fail or the majority delivers incorrect time. Therefore, the use of SNTP rather than NTP in primary servers should be carefully considered.

The only significant protocol change in SNTP Version 4 over previous versions of NTP and SNTP is a modified header interpretation to accommodate Internet Protocol Version 6 (IPv6) and OSI addressing. However, SNTP Version 4 includes certain optional extensions to the basic Version 3 model, including an anycast mode and an authentication scheme designed specifically for multicast and anycast modes.

Protocol Structure

SNTP message has the same format as the NTP:

2	5	8	16	24	32bit
LI	VN	Mode	Stratum	Poll	Precision
Root Delay					
Root Dispersion					
Reference Identifier					
Reference timestamp (64)					
Originate Timestamp (64)					
Receive Timestamp (64)					
Transmit Timestamp (64)					
Key Identifier (optional) (32)					
Message digest (optional) (128)					

LI Leap Indicator warning of impending leap-second to be inserted at the end of the last day of the current month.

VN Version number indicating the version number.

Mode - The mode: This field can contain the following values:

0 Reserved.
1 Symmetric active.
3 Client.
4 Server.
5 Broadcast.
6 NTP control message.

Stratum
An integer identifying the stratum level of the local clock.

Poll
Signed integer indicating the maximum interval between successive messages, in seconds to the nearest power of 2.

Precision
Signed integer indicating the precision of the local clock, in seconds to the nearest power of 2.

Root Delay
Signed fixed-point number indicating the total roundtrip delay to the primary reference source, in seconds with fraction point between bits 15 and 16.

Root Dispersion
Unsigned fixed-point number indicating the nominal error relative to the primary reference source, in seconds with fraction point between bits 15 and 16.

Reference Identifier
Identifying the particular reference source.

Originate Timestamp
This is the time at which the request departed the client for the server, in 64-bit timestamp format.

Receive Timestamp
This is the time at which the request arrived at the server, in 64-bit timestamp format.

Transmit Timestamp
This is the time at which the reply departed the server for the client, in 64-bit timestamp format.

Authenticator (optional)
When the NTP authentication scheme is implemented, the Key Identifier and Message Digest fields contain the message authentication code (MAC) information defined.

Related protocols

NTP, UDP

Sponsor Source

SNTP is defined by IETF (http://www.ietf.org) in RFC 2030.

Reference

http://www.javvin.com/protocol/rfc2030.pdf
Simple Network Time Protocol (SNTP) Version 4 for IPv4, IPv6 and OSI.

Protocol Name

TELNET: Terminal emulation protocol of TCP/IP

Protocol Description

TELNET is the terminal emulation protocol in a TCP/IP environment. TELNET uses the TCP as the transport protocol to establish connection between server and client. After connecting, TELNET server and client enter a phase of option negotiation that determines the options that each side can support for the connection. Each connected system can negotiate new options or renegotiate old options at any time. In general, each end of the TELNET connection attempts to implement all options that maximize performance for the systems involved.

When a TELNET connection is first established, each end is assumed to originate and terminate at a "Network Virtual Terminal", or NVT. An NVT is an imaginary device which provides a standard, network-wide, intermediate representation of a canonical terminal. This eliminates the need for "server" and "user" hosts to keep information about the characteristics of each other's terminals and terminal handling conventions.

The principle of negotiated options takes cognizance of the fact that many hosts will wish to provide additional services over and above those available within an NVT and many users will have sophisticated terminals and would like to have elegant, rather than minimal, services.

Option requests are likely to flurry back and forth when a TELNET connection is first established, as each party attempts to get the best possible service from the other party. Beyond that, however, options can be used to dynamically modify the characteristics of the connection to suit changing local conditions.

Modern Telnet is a versatile terminal emulation due to the many options that have evolved over the past twenty years. Options give TELNET the ability to transfer binary data, support byte macros, emulate graphics terminals, and convey information to support centralized terminal management.

Protocol Structure

TELNET commands are ASCII text. The following are the TELNET commands:

Commands	Code No. Dec	Hex	Description
data			All terminal input/output data.
End subNeg	240	FO	End of option subnegotiation command.
No Operation	241	F1	No operation command.
Data Mark	242	F2	End of urgent data stream.
Break	243	F3	Operator pressed the Break key or the Attention key.
Int process	244	F4	Interrupt current process.
Abort output	245	F5	Cancel output from current process.
You there?	246	F6	Request acknowledgment.
Erase char	247	F7	Request that operator erase the previous character.
Erase line	248	F8	Request that operator erase the previous line.
Go ahead!	249	F9	End of input for half-duplex connections.
SubNegotiate	230	FA	Begin option subnegotiation.
Will Use	231	FB	Agreement to use the specified option.
Won't Use	232	FC	Reject the proposed option.
Start use	233	FD	Request to start using specified option.
Stop Use	234	FE	Demand to stop using specified option.
LAC	235	FF	Interpret as command.

Related protocols

TCP, IP, SMTP, FTP

Sponsor Source

TELNET is defined by IETF (http://www.ietf.org) in RFC 854.

Reference

http://www.javvin.com/protocol/rfc854.pdf
TELNET PROTOCOL SPECIFICATION

Protocol Name

TFTP: Trivial File Transfer Protocol

Protocol Description

Trivial File Transfer Protocol (TFTP) is a simple protocol to transfer files. It has been implemented on top of the Internet User Datagram protocol (UDP). TFTP is designed to be small and easy to implement and, therefore, lacks most of the features of a regular FTP. TFTP only reads and writes files (or mail) from/to a remote server. It cannot list directories, and currently has no provisions for user authentication.

Three modes of transfer are currently supported by TFPT: netASCII, that is 8 bit ASCII; octet (this replaces the “binary” mode of previous versions of this document.) i.e. raw 8-bit bytes; mail, netASCII characters sent to a user rather than a file. Additional modes can be defined by pairs of cooperating hosts.

In TFTP, any transfer begins with a request to read or write a file, which also serves to request a connection. If the server grants the request, the connection is opened and the file is sent in fixed length blocks of 512 bytes. Each data packet contains one block of data and must be acknowledged by an acknowledgment packet before the next packet can be sent. A data packet of less than 512 bytes signals termination of a transfer. If a packet gets lost in the network, the intended recipient will timeout and may retransmit his last packet (which may be data or an acknowledgment), thus causing the sender of the lost packet to retransmit that lost packet. The sender has to keep just one packet on hand for retransmission, since the lock step acknowledgment guarantees that all older packets have been received. Notice that both machines involved in a transfer are considered senders and receivers. One sends data and receives acknowledgments, the other sends acknowledgments and receives data.

The current version of TFTP is version 2.

Protocol Structure

The basic TFTP header structure:

2bytes	String	2bytes	String	2bytes
Opcode	Filename	0	Mode	0

Opcode – Operation code or commands. The following are TFTP commands:

Opcode	Command	Description
1	Read Request	Request to read a file.
2	Write Request	Request to write to a file.
3	File Data	Transfer of file data.
4	Data Acknowledge	Acknowledgement of file data.
5	Error	Error indication.

Filename — the name of file to be transferred.

Mode — Datamode. The format of the file data that the protocol is to transfer. It could be NetASCII Standard ASCII, Octet Eight-bit binary data, or Mail Standard ASCII.

Related protocols

UDP, FTP

Sponsor Source

TFTP is defined by IETF (http://www.ietf.org) in RFC 1350.

Reference

http://www.javvin.com/protocol/rfc1350.pdf
The TFTP Protocol (Revision 2)

Protocol Name

URL: Uniform Resource Locator

Protocol Description

URL is the syntax and semantics for a compact string representation of a resource available via the Internet. For example, we use URL to locate web addresses and FTP site addresses. The generic syntax for URLs provides a framework for new schemes to be established using protocols other than those defined in this document.

URLs are used to `locate' resources, by providing an abstract identification of the resource location. Having located a resource, a system may perform a variety of operations on the resource, as might be characterized by such words as `access', `update', `replace', `find attributes'. In general, only the `access' method needs to be specified for any URL scheme.

Protocol Structure

URLs are sequences of characters, i.e., letters, digits, and special characters. URLs are written as follows:

<scheme>:<scheme-specific-part>

A URL contains the name of the scheme being used (<scheme>) followed by a colon and then a string (the <scheme-specific-part>) whose interpretation depends on the scheme.

Scheme names consist of a sequence of characters. The lower case letters “a”--”z”, digits, and the characters plus (“+”), period (“.”), and hyphen (“-”) are allowed. For resiliency, programs interpreting URLs should treat upper case letters as equivalent to lower case in scheme names (e.g., allow “HTTP” as well as “http”).

Related protocols

http, www, FTP

Sponsor Source

URL is defined by IETF (http://www.ietf.org) in RFC 1738.

Reference

http://www.javvin.com/protocol/rfc1738.pdf
Uniform Resource Locators (URL)

Protocol Name

Whois (and RWhois): Remote Directory Access Protocol

Protocol Description

The whois protocol retrieves information about domain names from a central registry. The whois service is provided by the organizations that run the Internet. Whois is often used to retrieve registration information about an Internet domain or server. It can tell you who owns the domain, how their technical contact can be reached, along with other information.

The original Whois function was to be a central directory of resources and people on ARPANET. However, it could not adequately meet the needs of the expanded Internet. RWhois extends and enhances the Whois concept in a hierarchical and scaleable fashion. In accordance with this, RWhois focuses primarily on the distribution of “network objects”, or the data representing Internet resources or people, and uses the inherently hierarchical nature of these network objects (domain names, Internet Protocol (IP) networks, email addresses) to more accurately discover the requested information.

The RWhois defines both a directory access protocol and a directory architecture. As a directory service, RWhois is a distributed database, where data is split across multiple servers to keep database sizes manageable.

On the Internet, two such types of data are widely used: domain names and IP networks. Domain names are organized via a label-dot system, reading from a more specific label to a more general label left to right. IP networks are also lexically hierarchical labels using the Classless Inter-Domain Routing (CIDR) notation but their hierarchy is not easily determined with simple text manipulation. Instead, an IP network’s hierarchy is determined by converting the network to binary notation and applying successively shorter bit masks.

RWhois directs clients toward the appropriate authority area by generating referrals. Referrals are pointers to other servers that are presumed to be closer to the desired data. The client uses this referral to contact the next server and ask the same question. The next server may respond with data, an error, or another referral (or referrals). By following this chain of referrals, the client will eventually reach the server with the appropriate authority area.

Protocol Structure

The entire RWhois protocol can be defined as a series of directives, responses, queries, and results.

```
rwhois-protocol = client-sends / server-returns
client-sends = *(directives / rwhois-query)
server-returns = *(responses / rwhois-query-result)
```

Related protocols

TCP, SMTP, FTP, Finger, DNS

Sponsor Source

Whois is originally defined by IETF (http://www.ietf.org).

Reference

http://www.javvin.com/protocol/rfc954.pdf
Nickname/Whois
http://www.javvin.com/protocol/rfc2167.pdf
Referral Whois (RWhois) Protocol V1.5

Protocol Name

X Window/X Protocol: X Window System Protocol

Protocol Description

The X Window System Protocol, also known as X Window or X Protocol, is a graphics architecture used as the graphical system on UNIX systems (primarily) and Linux systems. The X Window System is also used, less commonly, on VMS, MVS, and MS-Windows systems. X Window System (X Protocol) provides an inherently client/server oriented base for displaying windowed graphics. X Window provides a public protocol by which client programs can query and update information on X servers. X Window (X Protocol) allows processes on various computers on a network to display contents on display devices elsewhere on the network.

X Window System (X Protocol) defines the Client and Server roles as follows:

- An X server is a program that runs on a user's desktop to manage a video system including "interactive" I/O devices such as mice, keyboards, and some more unusual devices. The key functions are: 1) displays drawing requests on the screen. 2) replies to information requests. 3) reports an error in a request. 4) manages the keyboard, mouse and display device. 5) multiplexes keyboard and mouse input onto the network (or via local IPC) to the respective X clients. (X events) 6) creates, maps and destroys windows and 7) writes and draws in windows.
- X client is an application program that often runs on another host which connect to an X Server in order to display things. The client is often on a powerful Unix/Linux box that would commonly be known as a "server." The key functions are: 1) sends requests to the server. 2) receives events from server. 3) receives errors from the server.

X systems separate out the various components as separate subsystems. The key components in the X Window System (X Protocol) architecture are:

- Window manager - controls what happens when the mouse pointer is pointing outside of screen areas controlled by specific applications.
- Program/File manager - which is commonly a program that displays icons representing applications, and allows the user to run those applications.
- Inter-application interfaces - The standard scheme for X Window clients to communicate is commonly termed ICCCM. CORBA is also used to provide more sophisticated ways for X Window clients to communicate. The communications are based on TCP/IP network.

X Window System (X Protocol) has two primary versions: X10 and X11.

Protocol Structure

The X Protocol has the following key communication messages between the Client and Server:

Requests

- X clients make requests to the X server for a certain action to take place. i.e.: Create Window
- To enhance performance, the X client normally does not expect nor wait for a response. The request is typically left to the reliable network layer to deliver.
- X requests are any multiple of 4 bytes.

Replies

- The X server will respond to certain X client requests that require a reply. As noted, not all requests require a reply.
- X replies are any multiple of 4 bytes with a minimum of 32 bytes.

Events

- The X server will forward to the X client an event that the application is expecting. This could include keyboard or mouse input. To minimize network traffic, only expected events are sent to X clients.
- X events are 32 bytes

Errors

- The X server will report errors in requests to the X client. Errors are like an event but are handled differently.
- X errors are the same size as events to simplify their handling. They are sent to the error handling routine of the X client. (32 bytes)

Related protocols

IP, TCP, CORBA

Sponsor Source

X Window/ X Protocol is currently developed by X.ORG. (http://www.ietf.org).

Reference

http://www.x.org/X11_protocol.html
The X Protocol

Presentation Layer Protocols

Protocol Name

LPP: Lightweight Presentation Protocol

Protocol Description

Lightweight Presentation Protocol (LPP) describes an approach for providing “stream-lined” support of OSI application services on top of TCP/IP-based network for some constrained environments. LPP was initially derived from a requirement to run the ISO Common Management Information Protocol (CMIP) in TCP/IP-based networks.

LPP is designed for a particular class of OSI applications, namely those entities whose application context contains only an Association Control Service Element (ACSE) and a Remote Operations Service Element (ROSE). In addition, a Directory Services Element (DSE) is assumed for use by the application-entity, but only in a very limited sense. LPP is not applicable to entities whose application context is more extensive (e.g., contains a Reliable Transfer Service Element).

If one wants to implement ISO applications in a TCP/IP based network without constrains, the ITOT mechanisms (specified in RFC 2126) should be used.

Protocol Structure

The service provider is in one of the following states:
IDLE, WAIT1, WAIT2, DATA, WAIT3 or WAIT4

The possible events are:

PS-user	P-CONNECT.REQUEST
	P-CONNECT.RESPONSE
	P-RELEASE.REQUEST
	P-RELEASE.RESPONSE
	P-DATA.REQUEST
	P-U-ABORT.REQUEST
network	TCP closed or errored(*)
	receive ConnectRequest PDU
	receive ConnectResponse PDU
	receive ReleaseRequest PDU
	receive ReleaseResponse PDU
	receive UserData(*) or CL-UserData(**) PDU
	receive user-initiated Abort PDU
	receive provider-initiated Abort PDU
	timer expires(**)

The possible actions are:

PS-user	P-CONNECT.INDICATION
	P-CONNECT.CONFIRMATION
	P-RELEASE.INDICATION
	P-RELEASE.CONFIRMATION
	P-DATA.INDICATION
	P-U-ABORT.INDICATION
	P-P-ABORT.INDICATION
network	open TCP(*)
	close TCP(*)
	send ConnectRequest PDU
	send ConnectResponse PDU
	send ReleaseRequest PDU
	send ReleaseResponse PDU
	send UserData(*) or CL-UserData(**) PDU
	send user-initiated Abort PDU
	send provider-initiated Abort PDU
	set timer(**)

(*) tcp-based service only
(**) udp-based service only

Related protocols

TCP, UDP, IP, CMIP, CMOT, CMIS, ACSE, ROSE, CMISE, ITOT

Sponsor Source

LPP is defined by ISO (http://www.ietf.org) and IETF (http://www.ietf.org).

Reference

http://www.javvin.com/protocol/rfc1085.pdf
ISO Presentation Services on top of TCP/IP-based internets
http://www.javvin.com/protocol/rfc2126.pdf
ISO Transport Service on top of TCP (ITOT)

Session Layer Protocols

Protocol Name

RPC: Remote Procedure Call protocol

Protocol Description

Remote Procedure Call (RPC) is a protocol for requesting a service from a program located in a remote computer through a network, without having to understand the under layer network technologies. RPC presumes the existence of a low-level transport protocol, such as TCP or UDP, for carrying the message data between communicating programs. RPC spans the Transport layer and the Application layer in the Open Systems Interconnection (OSI) model of network communication. RPC makes it easier to develop an application that includes multiple programs distributed in a network.

RPC uses the client/server model. The requesting program is a client and the service-providing program is the server. First, the caller process sends a call message that includes the procedure parameters to the server process. Then, the caller process waits for a reply message (blocks). Next, a process on the server side, which is dormant until the arrival of the call message, extracts the procedure parameters, computes the results, and sends a reply message. The server waits for the next call message. Finally, a process on the caller receives the reply message, extracts the results of the procedure, and the caller resumes execution.

There are several RPC models and implementations. Sun Microsystem originally introduced the RPC. IETF ONC charter modified the Sun version and made the ONC PRC protocol, an IETF standard protocol. A popular model and implementation is the Open Software Foundation's Distributed Computing Environment (DCE).

Protocol Structure

The Remote Procedure Call (RPC) message protocol consists of two distinct structures: the call message and the reply message. The message flows are displayed as follows:

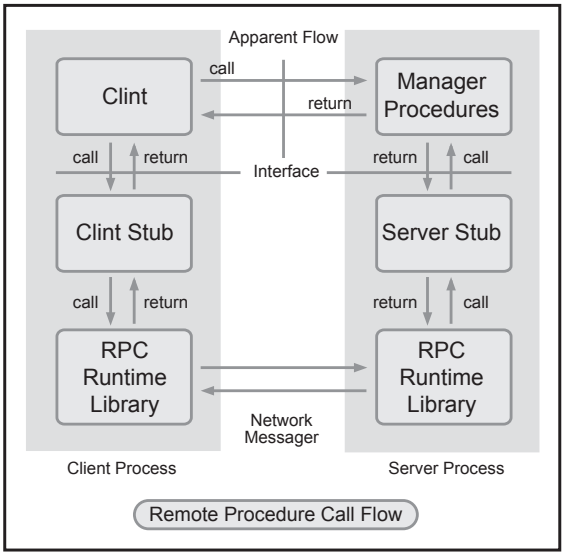


Figure 2-2: Remote Procedure Call Flow

RPC Call Message: Each remote procedure call message contains the following unsigned integer fields to uniquely identify the remote procedure:

- Program number
- Program version number
- Procedure number

The body of an RPC call message takes the following form:

```
struct call_body {
    unsigned int rpcvers;
    unsigned int prog;
    unsigned int vers;
    unsigned int proc;
    opaque_auth cred;
    opaque_auth verf;
    1 parameter
    2 parameter . . .
};
```

RPC Reply Message: The RPC protocol for a reply message varies depending on whether the call message is accepted or rejected by the network server. The reply message to a request contains information to distinguish the following conditions:

- RPC executed the call message successfully.
- The remote implementation of RPC is not protocol version 2. The lowest and highest supported RPC version

numbers are returned.
- The remote program is not available on the remote system.
- The remote program does not support the requested version number. The lowest and highest supported remote program version numbers are returned.
- The requested procedure number does not exist. This is usually a caller-side protocol or programming error.

The RPC reply message takes the following form:

```
enum reply_stat stat {
    MSG_ACCEPTED = 0,
    MSG_DENIED   = 1
};
```

Reference

http://www.javvin.com/protocol/rfc1831.pdf
RPC: Remote Procedure Call Protocol Specification Version 2 (ONC version)
http://www.javvin.com/protocol/rfc1057.pdf
RPC: Remote Procedure Call Protocol Specification Version 2 (Sun version)
The IEEE defines RPC in its ISO Remote Procedure Call Specification, ISO/IEC CD 11578 N6561, ISO/IEC, November 1991.

Transport Layer Protocols

Protocol Name

ITOT: ISO Transport Service on top of TCP

Protocol Description

ISO Transport Service on top of TCP (ITOT) is a mechanism that enables ISO applications to be ported to a TCP/IP network. There are two basic approaches which can be taken when "porting" ISO applications to TCP/IP (and IPv6) environments. One approach is to port each individual application separately, developing local protocols on top of TCP. A second approach is based on the notion of layering the ISO Transport Service over TCP/IP. This approach solves the problem for all applications which use the ISO Transport Service.

ITOT is a Transport Service which is identical to the Services and Interfaces offered by the ISO Transport Service Definition [ISO8072], but which will in fact implement the ISO Transport Protocol [ISO8073] on top of TCP/IP (IPv4 or IPv6), rather than the ISO Network Service [ISO8348]. The 'well known' TCP port 102 is reserved for hosts which implement the ITOT Protocol.

Two variants of the ITOT protocol are defined, "Class 0 over TCP" and "Class 2 over TCP", which are based closely on the ISO Transport Class 0 and 2 Protocol. Class 0 provides the functions needed for connection establishment with negotiation, data transfer with segmentation, and protocol error reporting. It provides Transport Connection with flow control based on that of the NS-provider (TCP). It provides Transport Disconnection based on the NS-provider Disconnection. Class 0 is suitable for data transfer with no Explicit Transport Disconnection.

Class 2 provides the functions needed for connection establishment with negotiation, data transfer with segmentation and protocol error reporting. It provides Transport Connection with flow control based on that of the NS-provider TCP. It provides Explicit Transport Disconnection. Class 2 is suitable when independence of Normal and Expedited Data channels is required or when Explicit Transport Disconnection is needed.

Protocol Structure

8	16	32bit	Variable
Version	Reserved	Packet Length	TPDU
Message Length			

Protocol Version: Value: 3
Reserved - Value: 0
Packet Length - Value: Length of the entire TPKT in octets, including Packet Header
TPDU - ISO Transport TPDU as defined in ISO 8073.

Mapping parameters between the TCP service and the ISO 8348 CONS service is done as follow:

ISO Network Service	*TCP*
CONNECTION ESTABLISHMENT	
Called address	Server's IPv4 or IPv6 address and TCP port number.
Calling address	Client's IPv4 or IPv6 address
All other parameters	Ignored
DATA TRANSFER	
NS User Data (NSDU)	DATA
CONNECTION RELEASE	
All parameters	Ignored

Related protocols

TCP, UDP, IP, CMIP, CMOT, CMIS, ACSE, ROSE, CMISE, ITOT

Sponsor Source

LPP is defined by ISO (http://www.ietf.org) and IETF (http://www.ietf.org).

Reference

http://www.javvin.com/protocol/rfc1085.pdf
ISO Presentation Services on top of TCP/IP-based internets
http://www.javvin.com/protocol/rfc2126.pdf
ISO Transport Service on top of TCP (ITOT)

Protocol Name

RDP : Reliable Data Protocol

Protocol Description

RDP is a connection-oriented transport protocol designed to efficiently support the bulk transfer of data for such host monitoring and control applications as loading/dumping and remote debugging. It attempts to provide only those services necessary, in order to be efficient in operation and small in size. The key functions of RDP are as follows:

RDP will provide a full-duplex communications channel between the two ports of each transport connection.

RDP will attempt to reliably deliver all user messages and will report a failure to the user if it cannot deliver a message. RDP extends the datagram service of IP to include reliable delivery.

RDP will attempt to detect and discard all damaged and duplicate segments. It will use a checksum and sequence number in each segment header to achieve this goal.

RDP will optionally provide sequenced delivery of segments. Sequenced delivery of segments must be specified when the connection is established.

RDP will acknowledge segments received out of sequence, as they arrive. This will free up resources on the sending side.

RDP supports a much simpler set of functions than TCP. The flow control, buffering, and connection management schemes of RDP are considerably simpler. The goal is a protocol that can be easily and efficiently implemented and that will serve a range of applications.

RDP functions can also be subset to further reduce the size of a particular implementation. For example, a target processor requiring down-loading from another host might implement an RDP module supporting only the passive Open function and a single connection. The module might also choose not to implement out-of-sequence acknowledgements.

Protocol Structure

1	2	3	4	5	6	8	16bit
SYN	ACK	EAK	RST	NUL	0	Ver No	Header Length
Source Port							
Destination Port							
Data Length							
Sequence Number							
Acknowledgement Number							
Checksum							
Variable header area …							

Control flags

The 8 control bits are divided as follows:.

SYN The SYN bit indicates a synchronization segment is present.

ACK The ACK bit indicates the acknowledgment number in the header is valid.

EACK The EACK bit indicates an extended acknowledge segment is present.

RST The RST bit indicates the packet is a reset segment.

NUL The NUL bit indicates the packet is a null segment.

0: The value of this field must be zero.

Ver no: version number; current version is 2.

Header length

The length of the RDP header.

Source Ports

Source address to identify the processes that originated the communication. The combination of the port identifiers with the source and destination addresses in the network access protocol header serves to fully qualify the connection and constitutes the connection identifier. This permits RDP to distinguish multiple connections between two hosts.

Destination Ports

Destination address to identify the processes targeted in the communication.

Data Length

The length in octets of the data in this segment. The data length does not include the RDP header.

Sequence number

The sequence number of this segment.

Acknowledgement number

If the ACK bit is set in the header, this is the sequence number of the segment that the sender of this segment last received correctly and in sequence. Once a connection is established this should always be sent.

Checksum

The checksum to ensure integrity

Variable Header Area

This area is used to transmit parameters for the SYN and EACK segments.

Related protocols

UDP, RUDP, IP, TCP, ICMP

Sponsor Source

RDP is defined by IETF (http://www.ietf.org) in RFC 908 and updated by RFC 1151.

Reference

http://www.javvin.com/protocol/rfc908.pdf
Reliable Data Protocol (RDP)
http://www.javvin.com/protocol/rfc1151.pdf
Version 2 of the Reliable Data Protocol (RDP)

Protocol Name

RUDP: Reliable User Datagram Protocol (Reliable UDP)

Protocol Description

Reliable UDP (RUDP) is a simple packet based transport protocol, based on RFCs 908 (version 1) and 1151 (version 2), which was intended as a reliable transport protocol to transport telephony signalling across IP networks. RUDP is designed to allow characteristics of each connection to be individually configured so that a number of protocols with different transport requirement can be implemented simultaneously not on the same platform. It is layered on the UDP/IP protocols and provides reliable in-order delivery (up to a maximum number of retransmissions) for virtual connections. RUDP has a very flexible design that makes it suitable for a variety of transport uses. One such use would be to transport telecommunication-signalling protocols.

Reliable UDP is a set of quality of service enhancements, such as congestion control tuning improvements, retransmit, and thinning server algorithms, that improves the ability to present a good quality RTP stream to RTP clients even in the presence of packet loss and network congestion. Reliable UDP's congestion control mechanisms allow streams to behave in a TCP-friendly fashion without disturbing the real-time nature of the protocol.

To work well with TCP traffic on the Internet, Reliable UDP uses retransmission and congestion control algorithms similar to the algorithms used by TCP. Additionally, these algorithms are time-tested to utilize available bandwidth optimally.

Reliable UDP features include:

- Client acknowledgment of packets sent by the server to the client
- Windowing and congestion control so the server does not exceed the currently available bandwidth
- Server retransmission to the client in the event of packet loss
- Faster than real-time streaming known as "overbuffering"

Protocol Structure

The basic TFTP header structure:

1	2	3	4	5	6	7	8	16bit
SYN	ACK	EAK	RST	NUL	CHK	TCS	0	Header Length
Sequence number								Ack number
Checksum								

Control bits
Indicate what is present in the packet. Details as follows:

SYN The SYN bit indicates a synchronization segment is present.
ACK The ACK bit indicates the acknowledgment number in the header is valid.
EACK The EACK bit indicates an extended acknowledge segment is present.
RST The RST bit indicates the packet is a reset segment.
NUL The NUL bit indicates the packet is a null segment.
CHK The CHK bit indicates whether the Checksum field contains the checksum of just the header or the header and the body (data).
TCS The TCS bit indicates the packet is a transfer connection state segment.
0 The value of this field must be zero.

Header length
Indicates where user data begins in the packet.

Sequence number
When a connection is first opened, each peer randomly picks an initial sequence number. This sequence number is used in the SYN segments to open the connection. Each transmitter increments the sequence number before sending a data, null, or reset segment.

Acknowledgement number
This field indicates to a transmitter the last in- sequence packet the receiver has received.

Checksum
The checksum is always calculated on the RUDP header to ensure integrity. The checksum here is the same algorithm used in UDP and TCP headers.

Related protocols

UDP, RDP, IP, TCP

Sponsor Source

RUDP is discussed in IETF (http://www.ietf.org) as documented in a memo.

Reference

http://www.javvin.com/protocol/reliable-UDP.pdf
Reliable UDP protocol
http://www.javvin.com/protocol/rfc908.pdf
Reliable Data Protocol (RDP)
http://www.javvin.com/protocol/rfc1151.pdf
Version 2 of the Reliable Data Protocol (RDP)

Protocol Name

TALI: Tekelec's Transport Adapter Layer Interface

Protocol Description

TALI is the interface of a Signalling Gateway, which provides interworking between the Switched Circuit Network (SCN) and an IP network. Since the Gateway is the central point of signalling information, not only does it provide transportation of signalling from one network to another, but can also provide additional functions such as protocol translation, security screening, routing information, and seamless access to Intelligent Network (IN) services on both networks.

The Transport Adapter Layer Interface (TALI) protocol provides TCAP, ISUP, and MTP messaging over TCP/IP and is used to support reliable communication between the SS7 Signalling Network and applications residing within the IP network.

This version of TALI provides 3 SS7 signalling transport methods and provides functionality for MTP over TCP/IP, SCCP/TCAP over TCP/IP and ISUP over TCP/IP.

Protocol Structure

The basic TFTP header structure:

16	32bit
SYNC	
OpCode	
Length	Service message data

SYNC
Four bytes must be (54 41 4C 49) TALI in ASCII.

OpCode
Operation code are specified as follows:

- Type of frame
- Test Service on this Socket test
- Allow Service messages on this socket allo
- Prohibit Service messages on this socket proh
- Prohibit Service messages Ack proa
- Monitor Socket message on this socket moni
- Monitor Socket message Ack mona
- SCCP Service message sccp
- ISUP Service message isot
- MTP3 Service message mtp3
- MTP Primitives mtpp
- SCCP Primitives scpp
- Routing Key Registration rkrg
- Routing Key De-Registration rkdr
- Special Service Message spcl

Length
The length of the frame. Non-zero if message contains a Service or Monitor Socket message.

Service message data
The service message data.

Related protocols

TCAP, ISUP, SCCP, TCP, IP, MTP, SS7

Sponsor Source

TALI is defined by IETF (http://www.ietf.org) in RFC 3094.

Reference

http://www.javvin.com/protocol/rfc3094.pdf
Tekelec's Transport Adapter Layer Interface

Protocol Name

TCP: Transmission Control Protocol

Protocol Description

Transmission Control Protocol (TCP) is the transport layer protocol in the TCP/IP suite, which provides a reliable stream delivery and virtual connection service to applications through the use of sequenced acknowledgment with retransmission of packets when necessary. Along with the Internet Protocol (IP), TCP represents the heart of the Internet protocols.

Since many network applications may be running on the same machine, computers need something to make sure the correct software application on the destination computer gets the data packets from the source machine, and some way to make sure replies get routed to the correct application on the source computer. This is accomplished through the use of the TCP “port numbers”. The combination of IP address of a network station and its port number is known as a “socket” or an “endpoint”. TCP establishes connections or virtual circuits between two “endpoints” for reliable communications.

Among the services TCP provides are stream data transfer, reliability, efficient flow control, full-duplex operation, and multiplexing.

With stream data transfer, TCP delivers an unstructured stream of bytes identified by sequence numbers. This service benefits applications because the application does not have to chop data into blocks before handing it off to TCP. TCP can group bytes into segments and pass them to IP for delivery.

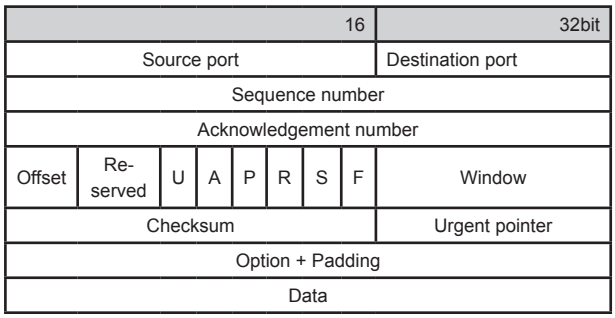
TCP offers reliability by providing connection-oriented, end-to-end reliable packet delivery. It does this by sequencing bytes with a forwarding acknowledgment number that indicates to the destination the next byte the source expects to receive. Bytes not acknowledged within a specified time period are retransmitted. The reliability mechanism of TCP allows devices to deal with lost, delayed, duplicate, or misread packets. A time-out mechanism allows devices to detect lost packets and request retransmission.

TCP offers efficient flow control - When sending acknowledgments back to the source, the receiving TCP process indicates the highest sequence number it can receive without overflowing its internal buffers.

Full-duplex operation: TCP processes can both send and receive packets at the same time.

Multiplexing in TCP: Numerous simultaneous upper-layer conversations can be multiplexed over a single connection.

Protocol Structure



- Source port -- Identifies points at which upper-layer source process receives TCP services.
- Destination port -- Identifies points at which upper-layer Destination process receives TCP services.
- Sequence number -- Usually specifies the number assigned to the first byte of data in the current message. In the connection-establishment phase, this field also can be used to identify an initial sequence number to be used in an upcoming transmission.
- Acknowledgment number – Contains the sequence number of the next byte of data the sender of the packet expects to receive. Once a connection is established, this value is always sent.
- Data offset -- 4 bits. The number of 32-bit words in the TCP header indicates where the data begins.
- Reserved -- 6 bits. Reserved for future use. Must be zero.
- Control bits (Flags) -- 6 bits. Carry a variety of control information. The control bits may be:
  - U (URG) Urgent pointer field significant.
  - A (ACK) Acknowledgment field significant.
  - P (PSH) Push function.
  - R (RST) Reset the connection.
  - S (SYN) Synchronize sequence numbers.
  - F (FIN) No more data from sender.
- Window -- 16 bits. Specifies the size of the sender’s receive window, that is, the buffer space available in octets for incoming data.
- Checksum -- 16 bits. Indicates whether the header was damaged in transit.
- Urgent Pointer -- 16 bits. Points to the first urgent data byte in the packet.
- Option + Paddling – Specifies various TCP options. There are two possible formats for an option: a single octet of option type; an octet of option type, an octet of option length and the actual option data octets.
- Data – contains upper-layer information.

Related protocols

IP, UDP, ICMP, SNMP, FTP, TELNET, SMTP, RPC, XDR, and NFS

Sponsor Source

TCP is defined by IETF (http://www.ietf.org) RFC793.

Reference

http://www.javvin.com/protocol/rfc793.pdf
TCP Specifications
http://www.javvin.com/protocol/rfc3168.pdf
The Addition of Explicit Congestion Notification (ECN) to IP
http://www.iana.org/assignments/port-numbers
TCP and UDP port numbers

Protocol Name

UDP: User Datagram Protocol

Protocol Description

UDP is a connectionless transport layer (layer 4) protocol in the OSI model which provides a simple and unreliable message service for transaction-oriented services. UDP is basically an interface between IP and upper-layer processes. UDP protocol ports distinguish multiple applications running on a single device from one another.

Since many network applications may be running on the same machine, computers need something to make sure the correct software application on the destination computer gets the data packets from the source machine and some way to make sure replies get routed to the correct application on the source computer. This is accomplished through the use of the UDP “port numbers”. For example, if a station wished to use a Domain Name System (DNS) on the station 128.1.123.1, it would address the packet to station 128.1.123.1 and insert destination port number 53 in the UDP header. The source port number identifies the application on the local station that requested domain name server, and all response packets generated by the destination station should be addressed to that port number on the source station. Details of UDP port numbers can be found in the reference.

Unlike TCP, UDP adds no reliability, flow-control, or error-recovery functions to IP. Because of UDP’s simplicity, UDP headers contain fewer bytes and consume less network overhead than TCP.

UDP is useful in situations where the reliability mechanisms of TCP are not necessary, such as in cases where a higher-layer protocol or application might provide error and flow control.

UDP is the transport protocol for several well-known application-layer protocols, including Network File System (NFS), Simple Network Management Protocol (SNMP), Domain Name System (DNS), and Trivial File Transfer Protocol (TFTP).

Protocol Structure

16	32bit
Source port	Destination port
Length	Checksum
Data	

- Source port – 16 bits. Source port is an optional field. When used, it indicates the port of the sending process and may be assumed to be the port to which a reply should be addressed in the absence of any other information. If not used, a value of zero is inserted.
- Destination port – 16 bits. Destination port has a meaning within the context of a particular Internet destination address.
- Length – 16 bits. The length in octets of this user datagram, including this header and the data. The minimum value of the length is eight.
- Checksum -- 16-bits The sum of a pseudo header of information from the IP header, the UDP header and the data, padded with zero octets at the end, if necessary, to make a multiple of two octets.
- Data – Contains upper-level data information.

Related protocols

IP, TCP, ICMP, SNMP, DNS, TFTP and NFS

Sponsor Source

UDP is defined by IETF (http://www.ietf.org) RFC768.

Reference

http://www.javvin.com/protocol/rfc768.pdf
User Datagram Protocol (UDP) Specifications
http://www.iana.org/assignments/port-numbers
UDP and TCP port numbers

Protocol Name

Van Jacobson: Compressed TCP protocol

Protocol Description

Van Jacobson is a compressed TCP protocol which improves the TCP/IP performance over low speed (300 to 19,200 bps) serial links and to solves problems in link-level framing, address assignment, routing, authentication and performance.

The compression proposed in the Van Jacobson protocol is similar in spirit to the Thinwire-II protocol. However, this protocol compresses more effectively (the average compressed header is 3 bytes compared to 13 in Thinwire-II) and is both efficient and simple to implement. Van Jacobson compression is specific to TCP/IP datagrams.

Protocol Structure

The format of the compressed TCP is as follows:

	C	I	P	S	A	W	U
Connection number (C)							
TCP checksum							
Urgent pointer (U)							
D Window (W)							
D Ack (A)							
D Sequence (S)							
D IP ID (I)							
data							

C, I, P, S, A, W, U - Change mask. Identifies which of the fields expected to change per-packet actually changed.

Connection number - Used to locate the saved copy of the last packet for this TCP connection.

TCP checksum - Included so that the end-to-end data integrity check will still be valid.

Urgent pointer - This is sent if URG is set.

D values for each field - Represent the amount the associated field changed from the original TCP (for each field specified in the change mask).

Related protocols

TCP

Sponsor Source

Van Jacobson is defined by IETF (http://www.ietf.org) in RFC 1144.

Reference

http://www.javvin.com/protocol/rfc1144.pdf
Compressing TCP/IP Headers for Low-Speed Serial Links

Network Layer Protocols
Routing Protocals

Protocol Name

BGP (BGP-4): Border Gateway Protocol

Protocol Description

The Border Gateway Protocol (BGP), runs over TCP and is an inter-Autonomous System routing protocol. BGP is the only protocol that is designed to deal with a network of the Internet's size, and the only protocol that can deal well with having multiple connections to unrelated routing domains. It is built on experience gained with EGP. The primary function of a BGP system is to exchange network reachability information with other BGP systems. This network reachability information includes information on the list of Autonomous Systems (ASs) that reachability information traverses. This information is sufficient to construct a graph of AS connectivity from which routing loops may be pruned and some policy decisions at the AS level may be enforced.

BGP-4 provides a new set of mechanisms for supporting classless interdomain routing (CIDR). These mechanisms include support for advertising an IP prefix and eliminate the concept of network "class" within BGP. BGP-4 also introduces mechanisms which allow aggregation of routes, including aggregation of AS paths. These changes provide support for the proposed supernetting scheme.

Protocol Structure

Marker (16 bytes)	Length (2 bytes)	Type (1 byte)

Marker — Message containing a value predictable by the receiver of the message.

Length — The length of the message including the header.

Type — The message type. Possible messages are: Open, Update, Notification, KeepAlive.

After a transport protocol connection is established, the first message sent by each side is an OPEN message. If the OPEN message is acceptable, a KEEPALIVE message confirming the OPEN is sent back. Once the OPEN is confirmed, UPDATE, KEEPALIVE, and NOTIFICATION messages may be exchanged. The format of each type of messages could be found in the reference documents.

Related protocols

IP, TCP, EGP

Sponsor Source

BGP is defined by IETF (http://www.ietf.org) RFC1771.

Reference

http://www.javvin.com/protocol/rfc1771.pdf
A Border Gateway Protocol 4 (BGP-4)
http://www.javvin.com/protocol/rfc1772.pdf
Application of the Border Gateway Protocol in the Internet
http://www.javvin.com/protocol/rfc1773.pdf
Experience with the BGP-4 protocol
http://www.javvin.com/protocol/rfc1774.pdf
BGP-4 Protocol Analysis

Protocol Name

EGP: Exterior Gateway Protocol

Protocol Description

Exterior Gateway Protocol (EGP) is for exchanging routing information between two neighbor gateway hosts in a network of autonomous systems. EGP is commonly used between hosts on the Internet to exchange routing table information. The protocol is based on periodic polling using Hello/I-Heard-You (I-H-U) message exchanges to monitor neighbor reachability and Poll commands to solicit Update responses. The routing table contains a list of known routers, the addresses they can reach, and a cost metric associated with the path to each router so that the best available route is chosen. Each router polls its neighbor at intervals between 120 to 480 seconds and the neighbor responds by sending its complete routing table. EGP-2 is the latest version of EGP.

A more recent exterior gateway protocol, the Border Gateway Protocol (BGP), provides additional capabilities.

Protocol Structure

Here are the EGP message types:

Name	*Function*
Request	request acquisition of neighbor and/or initialize polling variables
Confirm	confirm acquisition of neighbor and/or initialize polling variables
Refuse	refuse acquisition of neighbor
Cease	request de-acquisition of neighbor
Cease-ack	confirm de-acquisition of neighbor
Hello	request neighbor reachability
I-H-U	confirm neighbor reachability
Poll	request net-reachability update
Update	net-reachability update
Error	error

The common portion of the message format:

8	16	24	32bit
Version	Type	Code	Status
Checksum		Autonomous System number	
Sequence number		(Different for different messages)	

- Version -- The version number. This version is version 2.
- Type -- Identifies the message type.
- Code -- Identifies the message code.
- Status -- Contains message-dependent status information.
- Checksum -- The EGP checksum is the 16-bit one's complement of the one's complement sum of the EGP message starting with the EGP version number field. When computing the checksum the checksum field itself should be zero.
- Autonomous System Number -- Assigned number identifying the particular autonomous system.
- Sequence Number -- Send state variable (commands) or receive state variable (responses and indications).

Related protocols

IP, TCP, BGP, IGP

Sponsor Source

EGP is defined by IETF (http://www.ietf.org) RFC904.

Reference

http://www.javvin.com/protocol/rfc904.pdf
Exterior Gateway Protocol formal specification

Protocol Name

IP: Internet Protocol (IPv4)

Protocol Description

The Internet Protocol (IP) is a network-layer (Layer 3 in the OSI model) protocol that contains addressing information and some control information to enable packets to be routed in a network. IP is the primary network-layer protocol in the TCP/IP protocol suite. Along with the Transmission Control Protocol (TCP), IP represents the heart of the Internet protocols. IP is equally well suited for both LAN and WAN communications.

IP has two primary responsibilities: providing connectionless, best-effort delivery of datagrams through a network; and providing fragmentation and reassembly of datagrams to support data links with different maximum-transmission unit (MTU) sizes. The IP addressing scheme is integral to the process of routing IP datagrams through an internetwork. Each IP address has specific components and follows a basic format. These IP addresses can be subdivided and used to create addresses for subnetworks. Each computer (known as a host) on a TCP/IP network is assigned a unique 32-bit logical address that is divided into two main parts: the network number and the host number. The network number identifies a network and must be assigned by the Internet Network Information Center (InterNIC) if the network is to be part of the Internet. An Internet Service Provider (ISP) can obtain blocks of network addresses from the InterNIC and can itself assign address space as necessary. The host number identifies a host on a network and is assigned by the local network administrator.

When you send or receive data (for example, an e-mail note or a Web page), the message gets divided into little chunks called packets. Each of these packets contains both the sender's Internet address and the receiver's address. Because a message is divided into a number of packets, each packet can, if necessary, be sent by a different route across the Internet. Packets can arrive in a different order than the order they were sent in. The Internet Protocol just delivers them. It's up to another protocol, the Transmission Control Protocol (TCP) to put them back in the right order.

All other protocols within the TCP/IP suite, except ARP and RARP, use IP to route frames from host to host. There are two basic IP versions, IPv4 and IPv6. This document describes the IPv4 details. The IPv6 details are described in a separate document.

Protocol Structure

<table>
<tr><th>4</th><th>8</th><th>16</th><th colspan="2">32bit</th></tr>
<tr><td>Version</td><td>IHL</td><td>Type of service</td><td colspan="2">Total length</td></tr>
<tr><td colspan="3">Identification</td><td>Flags</td><td>Fragment offset</td></tr>
<tr><td colspan="3">Time to live</td><td>Protocol</td><td>Header checksum</td></tr>
<tr><td colspan="5">Source address</td></tr>
<tr><td colspan="5">Destination address</td></tr>
<tr><td colspan="5">Option + Padding</td></tr>
<tr><td colspan="5">Data</td></tr>
</table>

- Version— 4-bit field indicates the version of IP currently used.
- IP Header Length (IHL)— is the datagram header length in 32-bit words. Points to the beginning of the data. The minimum value for a correct header is 5.
- Type-of-Service— indicates the quality of service desired by specifying how an upper-layer protocol would like a current datagram to be handled, and assigns datagrams various levels of importance. These 8 bits fields are used for the assignment of Precedence, Delay, Throughput and Reliability.
- Total Length—specifies the length, in bytes, of the entire IP packet, including the data and header. The maximum length which can be specified by this field is 65,535 bytes. Typically, hosts are prepared to accept datagrams up to 576 bytes.
- Identification—contains an integer that identifies the current datagram. This field is assigned by sender to help receiver to assemble the datagram fragments.
- Flags—consists of a 3-bit field of which the two low-order (least-significant) bits control fragmentation. The low-order bit specifies whether the packet can be fragmented. The middle bit specifies whether the packet is the last fragment in a series of fragmented packets. The third or high-order bit is not used.
- Fragment Offset— This 13-bits field indicates the position of the fragment's data relative to the beginning of the data in the original datagram, which allows the destination IP process to properly reconstruct the original datagram.
- Time-to-Live— is a counter that gradually decrements down to zero, at which point the datagram is discarded. This keeps packets from looping endlessly.
- Protocol—indicates which upper-layer protocol receives incoming packets after IP processing is complete.
- Header Checksum—helps ensure IP header integrity. Since some header fields change, e.g., Time to Live, this is recomputed and verified at each point the Internet header is processed.
- Source Address—specifies the sending node.
- Destination Address—specifies the receiving node.
- Options—allows IP to support various options, such as security.
- Data—contains upper-layer information.

Related protocols

IPv6, TCP, UDP, ICMP, SNMP, FTP, TELNET, SMTP, ARP, RARP, RPC, XDR, and NFS

Sponsor Source

The Internet Protocol is defined by IETF (http://www.ietf.org) RFC 791.

Reference

http://www.javvin.com/protocol/rfc791.pdf
Internet Protocol Specifications
http://www.cisco.com/univercd/cc/td/doc/cisintwk/ito_doc/ip.htm
IP Overview

Protocol Name

IPv6: Internet Protocol version 6

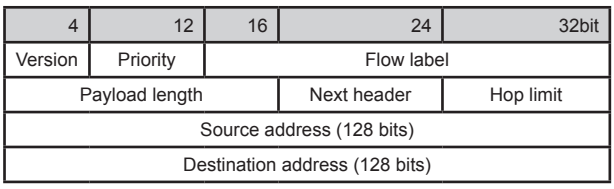
Protocol Description

IPv6 is the new version of Internet Protocol (IP) based on IPv4, a network-layer (Layer 3) protocol that contains addressing information and some control information enabling packets to be routed in the network. There are two basic IP versions: IPv4 and IPv6. IPv6 is also called next generation IP or IPng. IPv4 and IPv6 are de-multiplexed at the media layer. For example, IPv6 packets are carried over Ethernet with the content type 86DD (hexadecimal) instead of IPv4's 0800. This document describes the IPv6 details. The IPv4 is described in a separate document.

IPv6 increases the IP address size from 32 bits to 128 bits, to support more levels of addressing hierarchy, a much greater number of addressable nodes and simpler auto-configuration of addresses. IPv6 addresses are expressed in hexadecimal format (base 16) which allows not only numerals (0-9) but a few characters as well (a-f). A sample ipv6 address looks like: 3ffe:ffff:100:f101:210:a4ff:fee3:9566. Scalability of multicast addresses is introduced. A new type of address called an anycast address is also defined, to send a packet to any one of a group of nodes. Two major improvements in IPv6 vs. v4:

- Improved support for extensions and options - IPv6 options are placed in separate headers that are located between the IPv6 header and the transport layer header. Changes in the way IP header options are encoded allow more efficient forwarding, less stringent limits on the length of options, and greater flexibility for introducing new options in the future. The extension headers are: Hop-by-Hop Option, Routing (Type 0), Fragment, Destination Option, Authentication, and Encapsulation Payload.
- Flow labeling capability - A new capability has been added to enable the labeling of packets belonging to particular traffic flows for which the sender requests special handling, such as non-default Quality of Service or real-time service.

Protocol Structure



- Version – 4-bit Internet Protocol Version number (IPv6 is 6).
- Priority -- 8-bit traffic class field enables a source to identify the desired delivery priority of the packets. Priority values are divided into ranges: traffic where the source provides congestion control and non-congestion control traffic.
- Flow label -- 20-bit flow label is used by a source to label those products for which it requests special handling by the IPv6 router. The flow is uniquely identified by the combination of a source address and a non-zero flow label.
- Payload length -- 16-bit integer in octets is the length of payload including header.
- Next header – 8-bit selector identifies the type of header immediately following the IPv6 header.
- Hop limit -- 8-bit integer that is decremented by one by each node that forwards the packet. The packet is discarded if the Hop Limit is decremented to zero.
- Source address -- 128-bit address of the originator of the packet .
- Destination address -- 128-bit address of the intended recipient of the packet (possibly not the ultimate recipient, if a Routing header is present).

Related protocols

IP, TCP, UDP, ICMP, SNMP, FTP, TELNET, SMTP, ARP, RARP, RPC, XDR, and NFS

Sponsor Source

IPv6 is defined by IETF (http://www.ietf.org) RFC 1883 (original) and RFC 2460 (latest).

Reference

http://www.javvin.com/protocol/rfc1883.pdf
IPv6 Specifications (original)
http://www.javvin.com/protocol/rfc2460.pdf
IPv6 specifications (the latest)
http://www.ipv6forum.com
A good informational site

Protocol Name

ICMP & ICMPv6: Internet Message Control Protocol and ICMP version 6

Protocol Description

Internet Control Message Protocol (ICMP) is an integrated part of the IP suite. ICMP messages, delivered in IP packets, are used for out-of-band messages related to network operation or mis-operation. ICMP packet delivery is unreliable, so hosts can't count on receiving ICMP packets for any network problems. The key ICMP functions are:

- Announce network errors, such as a host or entire portion of the network being unreachable, due to some type of failure. A TCP or UDP packet directed at a port number with no receiver attached is also reported via ICMP.
- Announce network congestion. When a router begins buffering too many packets, due to an inability to transmit them as fast as they are being received, it will generate ICMP Source Quench messages. Directed at the sender, these messages should cause the rate of packet transmission to be slowed. Of course, generating too many Source Quench messages would cause even more network congestion, so they are used sparingly.
- Assist Troubleshooting. ICMP supports an Echo function, which just sends a packet on a round--trip between two hosts. Ping, a common network management tool, is based on this feature. Ping will transmit a series of packets, measuring average round--trip times and computing loss percentages.
- Announce Timeouts. If an IP packet's TTL field drops to zero, the router discarding the packet will often generate an ICMP packet announcing this fact. TraceRoute is a tool which maps network routes by sending packets with small TTL values and watching the ICMP timeout announcements.

The Internet Control Message Protocol (ICMP) was revised during the definition of IPv6. In addition, the multicast control functions of the IPv4 Group Membership Protocol (IGMP) are now incorporated in the ICMPv6.

Protocol Structure

8	16	32bit
Type	Code	Checksum
Identifier		Sequence number
Address mask		

- Type -- Messages can be error or informational messages. Error messages can be Destination unreachable, Packet too big, Time exceed, Parameter problem. The possible informational messages are, Echo Request, Echo Reply, Group Membership Query, Group Membership Report, Group Membership Reduction.
- Code -- For each type of message several different codes are defined. An example of this is the Destination Unreachable message, where possible messages are: no route to destination, communication with destination administratively prohibited, not a neighbor, address unreachable, port unreachable. For further details, refer to the standard.
- Checksum -- The 16-bit one's complement of the one's complement sum of the ICMP message starting with the ICMP Type. For computing the checksum, the checksum field should be zero.
- Identifier -- An identifier to aid in matching requests/replies; may be zero.
- Sequence number -- Sequence number to aid in matching requests/replies; may be zero.
- Address mask -- A 32-bit mask.

Related protocols

IP, TCP, IGMP, SNMP, DNS, TFTP and NFS

Sponsor Source

ICMP is defined by IETF (http://www.ietf.org) RFC792 and 950; ICMPv6 is defined by RFC 2461, 2463.

Reference

http://www.javvin.com/protocol/rfc792.pdf
Internet Control Message Protocol
http://www.javvin.com/protocol/rfc950.pdf
Internet Standard Subnetting Procedure
http://www.javvin.com/protocol/rfc2461.pdf
Neighbor Discovery for IP Version 6 (IPv6).
http://www.javvin.com/protocol/rfc2463.pdf
ICMPv6 for the Internet Protocol Version 6 (IPv6) Specification

Protocol Name

IRDP: ICMP Router Discovery Protocol

Protocol Description

ICMP Router Discovery Protocol (IRDP) enables a host to determine the address of a router that it can use as a default gateway. Similar to ES-IS but used with IP.

Router discovery uses Internet Control Message Protocol (ICMP) router advertisements and router solicitation messages to allow a host to discover the addresses of operational routers on the subnet. Hosts must discover routers before they can send IP datagrams outside their subnet. Router discovery allows a host to discover the addresses of operational routers on the subnet.

Each router periodically multicasts a router advertisement from each of its multicast interfaces, announcing the IP address of that interface. Hosts listen for advertisements to discover the addresses of their neighboring routers. When a host starts, it can send a multicast router solicitation to ask for immediate advertisements.

The router discovery messages do not constitute a routing protocol. They enable hosts to discover the existence of neighboring routers but do not determine which router is best to reach a particular destination.

Protocol Structure

ICMP Router Advertisement Message

8	16	32bit
Type	Code	Checksum
Num addrs	Addr Entry Size	Life Time
Router address 1		
Preference Level 1		
...		

IP Fields:

- Source Address - An IP address belonging to the interface from which this message is sent.
- Destination Address - The configured Advertisement Address or the IP address of a neighboring host.
- Time-to-Live - 1 if the Destination Address is an IP multicast address; at least 1 otherwise.

ICMP Fields:

- Type - 9
- Code – 0
- Checksum - The 16-bit one's complement of the one's complement sum of the ICMP message, starting with the ICMP Type. For computing the checksum, the Checksum field is set to 0.
- Num Addrs - The number of router addresses advertised in this message.
- Addr Entry Size - The number of 32-bit words of information per each router address (2, in the version of the protocol described here).
- Lifetime - The maximum number of seconds that the router addresses may be considered valid.
- Router Address[i] - The sending router's IP address(es) on the i = 1..Num Addrs interface from which this message is sent.
- Preference Level[i] - The preferability of each Router Address[i] i = 1..Num Addrs as a default router address, relative to other router addresses on the same subnet.

ICMP Router Solicitation Message:

8	16	32bit
Type	Code	Checksum
Reserved		

P Fields:

- Source Address - An IP address belonging to the interface from which this message is sent, or 0.
- Destination Address - The configured SolicitationAddress.
- Time-to-Live - 1 if the Destination Address is an IP multicast address; at least 1 otherwise.

ICMP Fields:

- Type - 10
- Code - 0
- Checksum - The 16-bit one's complement of the one's complement sum of the ICMP message, starting with the ICMP Type. For computing the checksum, the Checksum field is set to 0.
- Reserved - Sent as 0; ignored on reception.

Related protocols

IP, TCP, IGMP, ICMP

Sponsor Source

IRDP is defined by IETF (http://www.ietf.org) RFC 1256.

Reference

http://www.javvin.com/protocol/rfc1256.pdf
ICMP Router Discovery Messages
http://www.javvin.com/protocol/rfc792.pdf
Internet Control Message Protocol
http://www.javvin.com/protocol/rfc2463.pdf
ICMPv6 for the Internet Protocol Version 6 (IPv6) Specification

Protocol Name

Mobile IP: IP Mobility Support Protocol for IPv4 & IPv6

Protocol Description

Mobile IP is the key protocol to enable mobile computing and networking, which brings together two of the world's most powerful technologies, the Internet and mobile communication. In Mobile IP, two IP addresses are provided for each computer: home IP address which is fixed and care-of IP address which is changing as the computer moves. When the mobile moves to a new location, it must send its new address to an agent at home so that the agent can tunnel all communications to its new address timely.

The main components defined in the Mobile IPv6 architecture are shown as follows:

• Mobile node – A mobile unit that can change links, and therefore addresses, and maintain reachability using its home address.

•Home link - The link from which the mobile node originates.

•Home address - An address assigned to the mobile node when it is attached to the home link and through which the mobile node is always reachable, regardless of its location on an IPv6 network.

•Home agent - A router on the home link that maintains registrations of mobile nodes that are away from home and their current addresses.

•Foreign link - A link that is not the mobile node's home link.

•Care-of address - An address used by a mobile node while it is attached to a foreign link. The association of a home address with a care-of address for a mobile node is known as a binding.

•Correspondent node A node that communicates with a mobile node. A correspondent node does not have to be Mobile IPv6-capable.

There are two versions of Mobile IP: Mobile IP for IPv4 and IPv6. The major differences are summarized as follows:

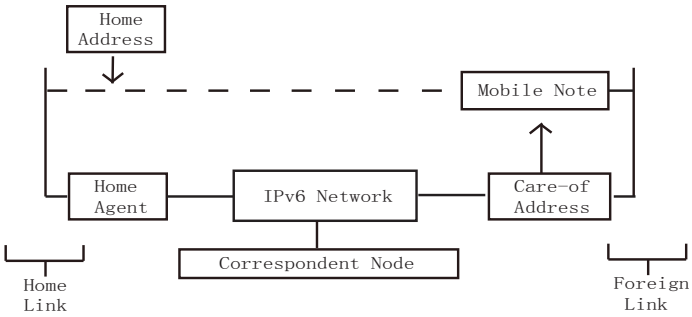


Figure 2-3: Mobile IP Functional Flow Chart

Key Features	Mobile IPv4	Mobile IPv6
Special router as foreign agent	Yes	No
Support for route optimization	Part of the protocol	In Extensions
Ensure symmetric reachability between mobile nodes and its router at current location	No	Yes
Routing bandwidth overhead	More	Less
Decouple from Link Layer	No	Yes
Need to manage "Tunnel soft state"	Yes	No
Dynamic home agent address discovery	No	Yes

Protocol Structure

Mobility IPv6 Protocol header structure:

8	16	24	32bit
Next Header	Length	Type	reserved
Checksum		Data :::	

Next Header - Identifies the protocol following this header.

Length - 8 bits unsigned. Size of the header in units of 8 bytes excluding the first 8 bytes.

Type - Mobility message types.

Type	Description
0	BRR, Binding Refresh Request.
1	HoTI, Home Test Init.
2	CoTI, Care-of Test Init.
3	HoT, Home Test.

4	CoT, Care-of Test.
5	BU, Binding Update.
6	Binding Acknowledgement.
7	BE, Binding Error.

Reserved - MUST be cleared to zero by the sender and MUST be ignored by the receiver.

Checksum - The 16 bit one's complement checksum of the Mobility Header.

Data - Variable length.

Related protocols

IP, UDP, IGMP, ICMP, Correspondent Node, Care-of Address, Destination Address, Home Agent, Mobility Internet Protocol, Mobile Node

Sponsor Source

Mobile IP is defined by IETF (http://www.ietf.org) RFC 3344 and 3775.

Reference

http://www.javvin.com/protocol/rfc3344.pdf
IP Mobility Support for IPv4

http://www.javvin.com/protocol/rfc3775.pdf
IP Mobility Support of IPv6

Protocol Name

NARP: NBMA Address Resolution Protocol

Protocol Description

The NBMA Address Resolution Protocol (NARP) allows a source terminal (a host or router), wishing to communicate over a Non-Broadcast, Multi-Access (NBMA) link layer network, to find out the NBMA addresses of a destination terminal if the destination terminal is connected to the same NBMA network as the source.

A conventional address resolution protocol, such as ARP for IP, may not be sufficient to resolve the NBMA address of the destination terminal, since it only applies to terminals belonging to the same IP subnetwork, whereas an NBMA network can consist of multiple logically independent IP subnets.

Once the NBMA address of the destination terminal is resolved, the source may either start sending IP packets to the destination (in a connectionless NBMA network such as SMDS) or may first establish a connection to the destination with the desired bandwidth and QOS characteristics (in a connection oriented NBMA network such as ATM).

An NBMA network can be non-broadcast either because it technically doesn't support broadcasting (e.g., an X.25 network) or because broadcasting is not feasible for one reason or another (e.g., an SMDS broadcast group or an extended Ethernet would be too large).

Protocol Structure

8	16	32bit
Version	Hop Count	Checksum
Type	Code	Unused
Destination IP address		
Source IP address		
NBMA Len.	NBMA address (variable length)	

- Version - NARP version number. Currently this value is 1.
- Hop Count - Indicates the maximum number of NASs that a request or reply is allowed to traverse before being discarded.
- Checksum - Standard IP checksum over the entire NARP packet (starting with the fixed header).
- Type - NARP packet type. The NARP Request has a type code 1; NARP Reply has a type code 2.
- Code - A response to an NARP request may contain cached information. If an authoritative answer is desired, then code 2.
- Source and Destination IP Address - Respectively, these are the IP addresses of the NARP requestor and the target terminal for which the NBMA address is destined.
- NBMA Length and NBMA Address - The NBMA length field is the length of the NBMA address of the source terminal in bits.

Related protocols

ARP

Sponsor Source

NARP is defined by IETF (http://www.ietf.org) in RFC 1735.

Reference

http://www.javvin.com/protocol/rfc1735.pdf
NBMA Address Resolution Protocol (NARP)

Protocol Name

NHRP: Next Hop Resolution Protocol

Protocol Description

NBMA Next Hop Resolution Protocol (NHRP) is used by a source station (host or router) connected to a Non-Broadcast, Multi-Access (NBMA) subnetwork to determine the internetworking layer address and NBMA subnetwork addresses of the "NBMA next hop" towards a destination station. If the destination is connected to the NBMA subnetwork, then the NBMA next hop is the destination station itself. Otherwise, the NBMA next hop is the egress router from the NBMA subnetwork that is "nearest" to the destination station. NHRP is intended for use in a multiprotocol internetworking layer environment over NBMA subnetworks.

NHRP Resolution Requests traverse one or more hops within an NBMA subnetwork before reaching the station that is expected to generate a response. Each station, including the source station, chooses a neighboring NHS to which it will forward the NHRP Resolution Request. The NHS selection procedure typically involves applying a destination protocol layer address to the protocol layer routing table which causes a routing decision to be returned. This routing decision is then used to forward the NHRP Resolution Request to the downstream NHS. The destination protocol layer address previously mentioned is carried within the NHRP Resolution Request packet. Note that even though a protocol layer address was used to acquire a routing decision, NHRP packets are not encapsulated within a protocol layer header but rather are carried at the NBMA layer using the encapsulation described in its own header.

Protocol Structure

8	16	24	32 bit
ar$afn		ar$pro.type	
ar$pro.snap			
ar$pro.snap	ar$hopcnt	ar$pkstz	
ar$chksum		ar$extoff	
ar$op.version	ar$op.type	ar$shtl	ar$sstl

- ar$afn - Defines the type of link layer address being carried.
- ar$pro.type - This field is an unsigned integer reserved for various uses.
- ar$pro.snap - Where a protocol has an assigned number in the ar$pro.type space (excluding 0x0080) the short form MUST be used when transmitting NHRP messages. If both Ethertype and NLPID codings exist then when transmitting NHRP messages, the Ethertype coding MUST be used. When ar$pro.type has a value of 0x0080, a snap encoded extension is being used to encode the protocol type.
- ar$hopcnt - The hop count. This indicates the maximum number of NHSs that an NHRP packet is allowed to traverse before being discarded.
- ar$pktsz - The total length of the NHRP packet in octets.
- ar$chksum - The standard IP checksum over the entire NHRP packet.
- ar$extoff - This field identifies the existence and location of NHRP extensions.
- ar$op.version - This field indicates what version of generic address mapping and management protocol is represented by this message.
- ar$op.type - If the ar$op.version is 1 then this field represents the NHRP packet type. Possible values for packet types are:
  - 1 NHRP Resolution Request.
  - 2 NHRP Resolution Reply.
  - 3 NHRP Registration Request.
  - 4 NHRP Registration Reply.
  - 5 NHRP Purge Request.
  - 6 NHRP Purge Reply.
  - 7 NHRP Error Indication.
- ar$shtl - The type and length of the source NBMA address interpreted in the context of the address family number.
- ar$sstl - The type and length of the source NBMA subaddress interpreted in the context of the "address family number".

Related protocols

ARP, NARP

Sponsor Source

NHRP is defined by IETF (http://www.ietf.org) in RFC 2332.

Reference

http://www.javvin.com/protocol/rfc2332.pdf
NBMA Next Hop Resolution Protocol (NHRP)

Protocol Name

OSPF: Open Shortest Path First protocol (version 2)

Protocol Description

Open Shortest Path First (OSPF) is an interior gateway protocol which is used for routing between routers belonging to a single Autonomous System. OSPF uses link-state technology in which routers send each other information about the direct connections and links which they have to other routers. Each OSPF router maintains an identical database describing the Autonomous System's topology. From this database, a routing table is calculated by constructing a shortest- path tree. OSPF recalculates routes quickly in the face of topological changes, utilizing a minimum of routing protocol traffic. OSPF provides support for equal-cost multi-path. An area routing capability is provided, enabling an additional level of routing protection and a reduction in routing protocol traffic. In addition, all OSPF routing protocol exchanges are authenticated.

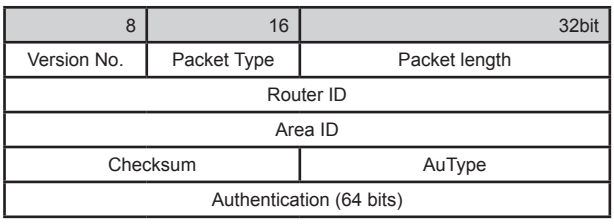
OSPF has been designed expressly for the TCP/IP internet environment, including explicit support for CIDR and the tagging of externally-derived routing information. OSPF also provides for the authentication of routing updates and utilizes IP multicast when sending/receiving the updates.

OSPF routes IP packets based solely on the destination IP address found in the IP packet header. IP packets are routed "as is"; they are not encapsulated in any further protocol headers as they transit the Autonomous System.

OSPF allows sets of networks to be grouped together. Such a grouping is called an area. The topology of an area is hidden from the rest of the Autonomous System. This information hiding enables a significant reduction in routing traffic. Also, routing within the area is determined only by the area's own topology, lending the area protection from bad routing data.

OSPF enables the flexible configuration of IP subnets. Each route distributed by OSPF has a destination and mask. Two different subnets of the same IP network number may have different sizes (i.e., different masks). This is commonly referred to as variable length subnetting. A packet is routed to the best (i.e., longest or most specific) match.

Protocol Structure



- Version number - Protocol version number (currently 2).
- Packet type - Valid types are as follows:
  1 Hello
  2 Database Description
  3 Link State Request
  4 Link State Update
  5 Link State Acknowledgment.
- Packet length - The length of the protocol packet in bytes. This length includes the standard OSPF header.
- Router ID - The router ID of the packet's source. In OSPF, the source and destination of a routing protocol packet are the two ends of a (potential) adjacency.
- Area ID - identifying the area that this packet belongs to. All OSPF packets are associated with a single area. Most travel a single hop only.
- Checksum - The standard IP checksum of the entire contents of the packet, starting with the OSPF packet header but excluding the 64-bit authentication field.
- AuType - Identifies the authentication scheme to be used for the packet.
- Authentication - A 64-bit field for use by the authentication scheme.

Related protocols

IP, TCP

Sponsor Source

OSPF is defined by IETF (http://www.ietf.org) in RFC 2328.

Reference

http://www.javvin.com/protocol/rfc2328.pdf
OSPF (Open Shortest Path First) version 2

Protocol Name

RIP: Routing Information Protocol (RIP2)

Protocol Description

Routing Information Protocol (RIP) is a standard for exchange of routing information among gateways and hosts. This protocol is most useful as an “interior gateway protocol”. In a nationwide network such as the current Internet, there are many routing protocols used for the whole network. The network will be organized as a collection of “autonomous systems”. Each autonomous system will have its own routing technology, which may well be different for different autonomous systems. The routing protocol used within an autonomous system is referred to as an interior gateway protocol, or “IGP”. A separate protocol is used to interface among the autonomous systems. The earliest such protocol, still used in the Internet, is “EGP” (exterior gateway protocol). Such protocols are now usually referred to as inter-AS routing protocols. RIP is designed to work with moderate-size networks using reasonably homogeneous technology. Thus it is suitable as an IGP for many campuses and for regional networks using serial lines whose speeds do not vary widely. It is not intended for use in more complex environments.

RIP2, derives from RIP, is an extension of the Routing Information Protocol (RIP) intended to expand the amount of useful information carried in the RIP2 messages and to add a measure of security. RIP2 is a UDP-based protocol. Each host that uses RIP2 has a routing process that sends and receives datagrams on UDP port number 520.

RIP and RIP2 are for the IPv4 network while the RIPng is designed for the IPv6 network. In this document, the details of RIP and RIP2 will be described.

Protocol Structure

8	16	32bit
Command	Version	Unused
Address family identifier		Route tag (only for RIP2; 0 for RIP)
IP address		
Subnet mask (only for RIP2; 0 for RIP)		
Next hop (only for RIP2; 0 for RIP)		
Metric		

- Command -- The command field is used to specify the purpose of the datagram. There are five commands: Request, Response, Traceon (obsolete), Traceoff (Obsolete) and Reserved.
- Version -- The RIP version number. The current version is 2.
- Address family identifier -- Indicates what type of address is specified in this particular entry. This is used because RIP2 may carry routing information for several different protocols. The address family identifier for IP is 2.
- Route tag -- Attribute assigned to a route which must be preserved and readvertised with a route. The route tag provides a method of separating internal RIP routes (routes for networks within the RIP routing domain) from external RIP routes, which may have been imported from an EGP or another IGP.
- IP address -- The destination IP address.
- Subnet mask -- Value applied to the IP address to yield the non-host portion of the address. If zero, then no subnet mask has been included for this entry.
- Next hop -- Immediate next hop IP address to which packets to the destination specified by this route entry should be forwarded.
- Metric -- Represents the total cost of getting a datagram from the host to that destination. This metric is the sum of the costs associated with the networks that would be traversed in getting to the destination.

Related protocols

IP, IPv6, IGP, EGP, RIPng, UDP, TCP

Sponsor Source

RIP is defined by IETF (http://www.ietf.org) RFC1058, RFC2453.

Reference

http://www.javvin.com/protocol/rfc1058.pdf
Routing Information Protocol Specification (Version 1)
http://www.javvin.com/protocol/rfc2453.pdf
RIP Version 2 Specification.

Protocol Name

RIPng: Routing Information Protocol next generation for IPv6

Protocol Description

RIPng, an information routing protocol for the IPv6, is based on protocols and algorithms used extensively in the IPv4 Internet. In an international network, such as the Internet, there are many routing protocols used for the entire network. The network will be organized as a collection of Autonomous Systems (AS). Each AS will have its own routing technology, which may differ among AS's. The routing protocol used within an AS is referred to as an Interior Gateway Protocol (IGP). A separate protocol, called an Exterior Gateway Protocol (EGP), is used to transfer routing information among the AS's. RIPng was designed to work as an IGP in moderate-size AS's. It is not intended for use in more complex environments.

RIPng is one of a class of algorithms known as Distance Vector algorithms. The basic algorithms used by this protocol were used in computer routing as early as 1969 in the ARPANET. However, the specific ancestry of this protocol is within the Xerox network protocols. The PUP protocols used the Gateway Information Protocol to exchange routing information. A somewhat updated version of this protocol was adopted for the Xerox Network Systems (XNS) architecture, with the name Routing Information Protocol (RIP). Berkeley's routed is largely the same as the Routing Information Protocol, with XNS addresses replaced by a more general address format capable of handling IPv4 and other types of address, and with routing updates limited to one every 30 seconds. Because of this similarity, the term Routing Information Protocol (or just RIP) is used to refer to both the XNS protocol and the protocol used by routed.

For the IPv4 network, the routing information protocols are RIP and RIP2 - click for details. In the document, only the details of RIPng will be described.

Protocol Structure

Command (1 byte)	Version (1 byte)	0 (2 bytes)
Route table entry 1 (20 bytes)		
. .		
Route table entry N (20 bytes)		

- Command -- Two commands are:
  - Request — A request for the responding system to send all or part of its routing table
  - Response — A message containing all or part of the sender's routing table.
- Version -- The version of the protocol. The current version is version 1.
- Route table entry -- Each route table entry contains a destination prefix, the number of significant bits in the prefix and the cost of reaching that destination.

Related protocols

RIP, RIP2, IP, UDP, TCP, EGP, IGP

Sponsor Source

RIPng is defined by IETF (http://www.ietf.org) RFC2080.

Reference

http://www.javvin.com/protocol/rfc2080.pdf
RIPng for IPv6

Protocol Name

RSVP: Resource ReSerVation Protocol

Protocol Description

Resource ReSerVation Protocol (RSVP) is a resource reservation setup protocol designed for quality integrated services over the Internet. RSVP is used by a host to request specific qualities of service from the network for particular application data streams or flows. RSVP is also used by routers to deliver quality-of-service (QoS) requests to all nodes along the path(s) of the flows and to establish and maintain state to provide the requested service. RSVP requests will generally result in resources being reserved in each node along the data path.

RSVP requests resources in only one direction. Therefore, RSVP treats a sender as logically distinct from a receiver, although the same application process may act as both a sender and a receiver at the same time. RSVP operates on top of IPv4 or IPv6, occupying the place of a transport protocol in the protocol stack. However, RSVP does not transport application data but is rather an Internet control protocol, like ICMP, IGMP, or routing protocols. Like the implementations of routing and management protocols, an implementation of RSVP will typically execute in the background, not in the data forwarding path.

RSVP is not a routing protocol by itself; RSVP is designed to operate with current and future unicast and multicast routing protocols. An RSVP process consults the local routing database(s) to obtain routes. In the multicast case, for example, a host sends IGMP messages to join a multicast group and then sends RSVP messages to reserve resources along the delivery path(s) of that group. Routing protocols determine where packets get forwarded; RSVP is only concerned with the QoS of those packets that are forwarded in accordance with routing. In order to efficiently accommodate large groups, dynamic group membership, and heterogeneous receiver requirements, RSVP makes receivers responsible for requesting a specific QoS [RSVP93]. A QoS request from a receiver host application is passed to the local RSVP process. The RSVP protocol then carries the request to all the nodes (routers and hosts) along the reverse data path(s) to the data source(s), but only as far as the router where the receiver's data path joins the multicast distribution tree. As a result, RSVP's reservation overhead is in general logarithmic rather than linear in the number of receivers.

Protocol Structure

4	8	16	32 bit
Version	Flags	Message type	RSVP checksum
Send TTL		(Reserved)	RSVP length

- Version -- The protocol version number, the current version is1.
- Flags -- No flag bits are defined yet.
- Message type -- Possible values are: 1 Path, 2 Resv, 3 PathErr, 4 ResvErr,, 5 PathTear, 6 ResvTear, 7 ResvConf.
- RSVP checksum -- The checksum for message errors.
- Send TTL -- The IP TTL value with which the message was sent.
- RSVP length -- The total length of the RSVP message in bytes, including the common header and the variable length objects that follow.

Related protocols

IP, TCP, UDP, RSVP-TE

Sponsor Source

RSVP is defined by IETF (http://www.ietf.org) RFC2205 with an update RFC2750.

Reference

http://www.javvin.com/protocol/rfc2205.pdf
RSVP Functional Specification
http://www.javvin.com/protocol/rfc2750.pdf
RSVP Extensions for Policy Control

Protocol Name

VRRP: Virtual Router Redundancy Protocol

Protocol Description

Virtual Router Dedundancy Protocol (VRRP) specifies an election protocol that dynamically assigns responsibility for a virtual router to one of the VRRP routers on a LAN. The VRRP router controlling the IP address(es) associated with a virtual router is called the Master, and forwards packets sent to these IP addresses. The election process provides dynamic fail over in the forwarding responsibility should the Master become unavailable. This allows any of the virtual router IP addresses on the LAN to be used as the default first hop router by end-hosts. The advantage of using VRRP is a higher availability default path without requiring configuration of dynamic routing or router discovery protocols on every end-host. VRRP packets are sent encapsulated in IP packets.

Using VRRP, a virtual IP address can be specified manually or with Dynamic Host Configuration Protocol (DHCP) as a default. A virtual IP address is shared among the routers, with one designated as the master router and the others as backups. In case, the master fails, the virtual IP address is mapped to a backup router's IP address. (This backup becomes the master router.) VRRP can also be used for load balancing. VRRP is part of both IPv4 and IPv6.

Protocol Structure

<table>
<tr><th>4</th><th>8</th><th>16</th><th>24</th><th>32bit</th></tr>
<tr><td>Version</td><td>Type</td><td>Virtual Rtr ID</td><td>Priority</td><td>Count IP Addrs</td></tr>
<tr><td colspan="2">Auth Type</td><td>Advet Int</td><td colspan="2">Checksum</td></tr>
<tr><td colspan="5">IP Address 1</td></tr>
<tr><td colspan="5">.....</td></tr>
<tr><td colspan="5">IP Address n</td></tr>
<tr><td colspan="5">Authentication Data 1</td></tr>
<tr><td colspan="5">Authentication Data 2</td></tr>
</table>

- Version -- The version field specifies the VRRP protocol version of this packet. This version is version 2.
- Type -- The type field specifies the type of this VRRP packet. The only packet type defined in this version of the protocol is: 1 ADVERTISEMENT.
- Virtual Rtr ID -- The Virtual Router Identifier (VRID) field identifies the virtual router this packet is reporting status for.
- Priority -- Specifies the sending VRRP router's priority for the virtual router. VRRP routers backing up a virtual router MUST use priority values between 1 to 254 (decimal).
- Count IP Addresses --The number of IP addresses contained in this VRRP advertisement.
- Auth Type -- Identifies the authentication method being utilized.
- Advertisement Interval -- Indicates the time interval (in seconds) between advertisements.
- Checksum – 16 bit field used to detect data corruption in the VRRP message.
- IP Address(es) -- One or more IP addresses that are associated with the virtual router. The number of addresses included is specified in the "Count IP Addrs" field. These fields are used for troubleshooting misconfigured routers.
- Authentication Data -- The authentication string is currently only utilized for simple text authentication, similar to the simple text authentication found in the Open Shortest Path First routing protocol (OSPF). It is up to 8 characters of plain text.

Related protocols

IP, IPv6, DHCP, TCP

Sponsor Source

VRRP is defined by IETF (http://www.ietf.org) RFC2338.

Reference

http://www.javvin.com/protocol/rfc2338.pdf
VRRP Specification

Multicasting Protocals

Protocol Name

BGMP: Border Gateway Multicast Protocol

Protocol Description

Border Gateway Multicast Protocol (BGMP) is a protocol for inter-domain multicast routing. BGMP natively supports “source-specific multicast” (SSM). To also support “any-source multicast” (ASM), BGMP builds shared trees for active multicast groups, and allows domains to build source-specific, inter-domain, distribution branches where needed. Building upon concepts from PIM-SM and CBT, BGMP requires that each global multicast group be associated with a single root. However, in BGMP, the root is an entire exchange or domain, rather than a single router.

For non-source-specific groups, BGMP assumes that ranges of the multicast address space have been associated with selected domains. Each such domain then becomes the root of the shared domain-trees for all groups in its range. An address allocator will generally achieve better distribution trees if it takes its multicast addresses from its own domain’s part of the space, thereby causing the root domain to be local.

BGMP uses TCP as its transport protocol. This eliminates the need to implement message fragmentation, retransmission, acknowledgement, and sequencing. BGMP uses TCP port 264 for establishing its connections. This port is distinct from BGP’s port to provide protocol independence, and to facilitate distinguishing between protocol packets.

Two BGMP peers form a TCP connection between one another, and exchange messages to open and confirm the connection parameters. They then send incremental Join/Prune Updates as group memberships change. BGMP does not require periodic refresh of individual entries. KeepAlive messages are sent periodically to ensure the liveness of the connection. Notification messages are sent in response to errors or special conditions. If a connection encounters an error condition, a notification message is sent and the connection is closed if the error is a fatal one.

Protocol Structure

16	24	32bit
Length	Type	Reserved

- Length - The total length of the message including the header in octets. It allows one to locate in the transport-level stream the start of the next message.
- Type - The type code of the message. The following type codes are available:
  1 OPEN; 2 UPDATE; 3 NOTIFICATION; 4 KEEPALIVE

  After a transport protocol connection is established, the first message sent by each side is an OPEN message. If the OPEN message is acceptable, a KEEPALIVE message confirming the OPEN is sent back. Once the OPEN is confirmed, UPDATE, KEEPALIVE, and NOTIFICATION messages may be exchanged.

  The format of each message type is different.

Related protocols

IP, TCP, BGP, PIM-SM

Sponsor Source

BGMP is drafted by IETF (http://www.ietf.org) currently.

Reference

http://www.javvin.com/protocol/ietf-bgmp-spec05.pdf
Border Gateway Multicast Protocol (BGMP): Protocol Specification.

Protocol Name

DVMRP: Distance Vector Multicast Routing Protocol

Protocol Description

Distance Vector Multicast Routing Protocol (DVMRP) is an Internet routing protocol that provides an efficient mechanism for connectionless message multicast to a group of hosts across an internetwork. DVMRP is an “interior gateway protocol” (IGP); suitable for use within an autonomous system, but not between different autonomous systems. DVMRP is not currently developed for use in routing non-multicast datagrams, so a router that routes both multicast and unicast datagrams must run two separate routing processes.

DVMRP is developed based upon RIP. DVMRP combines many of the features of RIP with the Truncated Reverse Path Broadcasting (TRPB) algorithm. In addition, to allow experiments to traverse networks that do not support multicasting, a mechanism called tunneling was developed. The key differences of DVMRP from RIP are: RIP routes and forwards datagrams to a particular destination. The purpose of DVMRP is to keep track of the return paths to the source of multicast datagrams.

DVMRP packets are encapsulated in IP datagrams, with an IP protocol number of 2 (IGMP).

Protocol Structure

DVMRP uses the IGMP to exchange routing datagrams. DVMRP datagrams are composed of two portions: a small, fixed length IGMP header, and a stream of tagged data.

4	8	16	24	32 bit
Version	Type	Sub-Type	Checksum	
DVMRP Data stream				

- Version – It is 1.
- Type – DVMRP type is 3.
- Sub-type - The subtype is one of:
  - 1 = Response; the message provides routes to some destination(s).
  - 2 = Request; the message requests routes to some destination(s).
  - 3 = Non-membership report; the message provides non-membership report(s).
  - 4 = Non-membership cancellation; the message cancels previous non-membership report(s).
- Checksum -- One’s complement of the one’s complement sum of the DVMRP message. The checksum must be calculated upon transmission and must be validated on reception of a packet. The checksum of the DVMRP message should be calculated with the checksum field set to zero.

Related protocols

IP, IGMP, RIP

Sponsor Source

DVMRP is defined by IETF (http://www.ietf.org) in RFC 1075.

Reference

http://www.javvin.com/protocol/rfc1075.pdf
Distance Vector Multicast Routing Protocol

Protocol Name

IGMP: Internet Group Management Protocol

Protocol Description

Internet Group Management Protocol (IGMP), a multicasting protocol in the internet protocols family, is used by IP hosts to report their host group memberships to any immediately neighboring multicast routers. IGMP messages are encapsulated in IP datagrams, with an IP protocol number of 2. IGMP has versions IGMP v1, v2 and v3.

- IGMPv1: Hosts can join multicast groups. There are no leave messages. Routers use a time-out based mechanism to discover the groups that are of no interest to the members.
- IGMPv2: Leave messages were added to the protocol, allowing group membership termination to be quickly reported to the routing protocol, which is important for high-bandwidth multicast groups and/or subnets with highly volatile group membership.
- IGMPv3: A major revision of the protocol allows hosts to specify the list of hosts from which they want to receive traffic. Traffic from other hosts is blocked inside the network. It also allows hosts to block inside the network packets that come from sources that send unwanted traffic.

The variant protocols of IGMP are:

- DVMRP: Distance Vector Multicast Routing Protocol.
- IGAP: IGMP for user Authentication Protocol.
- RGMP: Router-port Group Management Protocol.

Protocol Structure

There are basically 5 types of messages that must be implemented for IGMP v3 to function properly and be compatible with previous versions:

0x11: membership query
0x22: version 3 membership report
0x12: version 1 membership report
0x16: version 2 membership report
0x17 version 2 leave group

As an example, the message format for 0x11 (membership query) is displayed:

<table>
<tr><th colspan="2">8</th><th>16</th><th colspan="2">32 bit</th></tr>
<tr><td colspan="2">Type</td><td>Max response time</td><td colspan="2">Checksum</td></tr>
<tr><td colspan="5">Group address</td></tr>
<tr><td>RSV</td><td>S</td><td>QRV</td><td>QQIC</td><td>Number of Source</td></tr>
<tr><td colspan="5">Source Address (1)</td></tr>
<tr><td colspan="5">…</td></tr>
<tr><td colspan="5">Source Address (N)</td></tr>
</table>

- Type -- The message type: 0x11 (Membership query).
- Max Response Time -- Used only in Membership query messages. Specifies the maximum time allowed, in units of 1/10 second, before sending a responding report. In all other messages, it is set to 0 by the sender and ignored by the receiver.
- Checksum -- The checksum for message errors
- Group Address -- The Group address is set to 0 when sending a general query. It is set to the group address being queried, when sending a group specific query or group-and-source-specific query. In a membership report of a leave group message, it holds the IP multicast group address of the group being reported or left.
- RSV – Reserved; Set to zero on transmission, and ignored on reception.
- QQIC – Querier's Query Interval Code
- Number of Source (N) -- The number of source addresses in this message.
- Source Address – The vector of the IP unicast address

The details of other message types can be found in the reference RFC 1112, 2236 and 3376.

Related protocols

IP, TCP, DVMRP, IGAP, RGMP

Sponsor Source

IGMP is defined by IETF (http://www.ietf.org) RFC1112, RFC2236 and RFC3376.

Reference

http://www.javvin.com/protocol/rfc1112.pdf
IGMP version 1 specification
http://www.javvin.com/protocol/rfc2236.pdf
IGMP version 2 specification
http://www.javvin.com/protocol/rfc3376.pdf
IGMP version 3 specification

Protocol Name

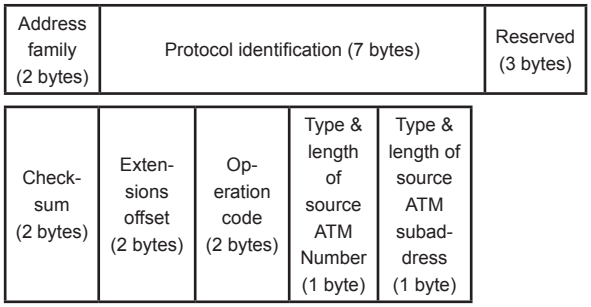
MARS: Multicast Address Resolution Server

Protocol Description

Multicasting is the process in which a source host or protocol entity sends a packet to multiple destinations simultaneously using a single, local 'transmit' operation. ATM is being utilized as a link layer technology to support a variety of protocols, including IP. ATM-based IP hosts and routers use a Multicast Address Resolution Server (MARS) to support IP multicast over the ATM Forum's UNI 3.0/3.1 point to multipoint connection service. Clusters of endpoints share a MARS and use it to track and disseminate information identifying the nodes listed as receivers for given multicast groups. This allows endpoints to establish and manage point to multipoint VCs when transmitting to the group.

The MARS protocol has two broad goals: to define a group address registration and membership distribution mechanism that allows UNI 3.0/3.1-based networks to support the multicast service of protocols and to define specific endpoint behaviors for managing point to multipoint VCs to achieve multicasting of layer 3 packets. MARS is an extended analog of the ATM ARP Server. It acts as a registry, associating layer 3 multicast group identifiers with the ATM interfaces representing the group's members. MARS messages support the distribution of multicast group membership information between MARS and endpoints (hosts or routers). Endpoint address resolution entities query the MARS when a layer 3 address needs to be resolved to the set of ATM endpoints making up the group at any one time. Endpoints keep the MARS informed when they need to join or leave particular layer 3 groups. To provide for asynchronous notification of group membership changes, the MARS manages a point to multipoint VC out to all endpoints desiring multicast support. Each MARS manages a cluster of ATM-attached endpoints.

Protocol Structure



- Address family -- Defines the type of link layer addresses being carried.
- Protocol ID -- Contains 2 subfields: 16 bits, protocol type; 40 bits, optional SNAP extension to protocol type.
- Reserved -- This reserved field may be subdivided and assigned specific meanings for other control protocols indicated by the version number.
- Checksum -- This field carries a standard IP checksum calculated across the entire message.
- Extension offset -- This field identifies the existence and location of an optional supplementary parameters list.
- Operation code -- This field is divided into 2 sub fields: version and type. Version indicates the operation being performed, within the context of the control protocol version indicated by mar$op.version.
- Type and length of ATM source number -- Information regarding the source hardware address.
- Type and length of ATM source subaddress -- Information regarding the source hardware subaddress.

Related protocols

ATM, UNI, IP

Sponsor Source

MARS is defined by IETF (http://www.ietf.org) RFC2022.

Reference

http://www.javvin.com/protocol/rfc2022.pdf
Support for Multicast over UNI 3.0/3.1 based ATM Networks

Protocol Name

MBGP: Multiprotocol BGP

Protocol Description

The multiprotocol BGP (MBGP) feature adds capabilities to BGP to enable multicast routing policy throughout the Internet and to connect multicast topologies within and between BGP autonomous systems. In other words, multiprotocol BGP (MBGP) is an enhanced BGP that carries IP multicast routes. BGP carries two sets of routes, one set for unicast routing and one set for multicast routing. The routes associated with multicast routing are used by the Protocol Independent Multicast (PIM) to build data distribution trees.

Multiprotocol BGP is useful when a link is required to be dedicated to multicast traffic, perhaps to limit which resources are used for which traffic, or if all multicast traffic exchange at one network access point (NAP) is required. Multiprotocol BGP allows a unicast routing topology different from a multicast routing topology.

The only three pieces of information carried by BGP-4 that are IPv4 specific are (a) the NEXT_HOP attribute (expressed as an IPv4 address), (b) AGGREGATOR (contains an IPv4 address), and (c) NLRI(expressed as IPv4 address prefixes). Any BGP speaker, including an MBGP speaker, has to have an IPv4 address, which will be used, among other things, in the AGGREGATOR attribute. To enable BGP-4 to support routing for multiple Network Layer protocols the only two things that have to be added to BGP-4 are (a) the ability to associate a particular Network Layer protocol with the next hop information, and (b) the ability to associate a particular Network Layer protocol with NLRI.

There are two attributes defined in the MBGP regarding NLRI: 1) MP_REACH_NLRI for the purpose of advertising a feasible route to a peer, permitting a route to advertise the network layer address of the router to be used as the next hop and allowing a given router to report some or all of the subnetwork points of attachment (SNPAs) and 2) MP_UNREACH_NLRI for the purpose of withdrawing multiple unfeasible routes from service.

To provide backward compatibility, as well as to simplify introduction of the multiprotocol capabilities into BGP-4, two new attributes, Multiprotocol Reachable NLRI (MP_REACH_NLRI), and Multiprotocol Unreachable NLRI (MP_UNREACH_NLRI) are used in the MBGP. MP_REACH_NLRI is used to carry the set of reachable destinations together with the next hop information to be used for forwarding to these destinations. MP_UNREACH_NLRI is used to carry the set of unreachable destinations. Both of these attributes are optional and non- transitive. This way a BGP speaker that doesn't support the multiprotocol capabilities will just ignore the information carried in these attributes, and will not pass it to other BGP speakers.

Protocol Structure

Multiprotocol Reachable NLRI - MP_REACH_NLRI (Type Code 14): The attribute is encoded as follows:

<table>
<tr><td colspan="2">2 Bytes</td><td>1Byte</td><td>1Byte</td></tr>
<tr><td colspan="2">Address Family Identifier</td><td>Subsequent Address Family Identifier</td><td>Length of Next Hop Network Address</td></tr>
<tr><td colspan="4">Network Address of Next Hop (variable)</td></tr>
<tr><td>Number of SNPAs</td><td>Length of first SNPA</td><td>First SNPA (variable)</td><td>Length of second SNPA (1 Byte)</td></tr>
<tr><td>Second SNPA (variable)</td><td>Length of Last SNPA (1 Byte)</td><td>Last SNPA (variable)</td><td>Network Layer Reachability Information (variable)</td></tr>
</table>

- Address Family Identifier - carries the identity of the Network Layer protocol associated with the Network Address that follows.
- Subsequent Address Family Identifier - provides additional information about the type of the Network Layer Reachability Information carried in the attribute.
- Length of Next Hop Network Address - expresses the length of the "Network Address of Next Hop" field as measured in octets.
- Network Address of Next Hop - a variable length field that contains the Network Address of the next router on the path to the destination system.
- Number of SNPAs - contains the number of distinct SNPAs to be listed in the following fields. The value 0 may be used to indicate that no SNPAs are listed in this attribute.
- Length of Nth SNPA – expresses the length of the "Nth SNPA of Next Hop" field as measured in semi-octets
- Nth SNPA of Next Hop - contains an SNPA of the router whose Network Address is contained in the "Network Address of Next Hop" field.
- Network Layer Reachability Information - lists NLRI for the feasible routes that are being advertised in this attribute.

Multiprotocol Unreachable NLRI - MP_UNREACH_NLRI: The attribute is encoded as follows:

<table>
<tr><td>Address Family Identifier (2 Bytes)</td><td>Subsequent Address Family Identifier (1 Byte)</td><td>Withdrawn Routes (variable)</td></tr>
</table>

- Address Family Identifier - carries the identity of the Network Layer protocol associated with the NLRI that follows.

- Subsequent Address Family Identifier - provides additional information about the type of the Network Layer Reachability Information carried in the attribute.
- Withdrawn Routes - lists NLRI for the routes that are being withdrawn from service.

Related protocols

IP, TCP, BGP

Sponsor Source

MBGP is defined by IETF (http://www.ietf.org) RFC2858.

Reference

http://www.javvin.com/protocol/rfc2858.pdf
Multiprotocol Extensions for BGP-4

Protocol Name

MOSPF: Multicast Extensions to OSPF

Protocol Description

Multicast Extensions to OSPF (MOSPF) provides enhancements to OSPF Version 2 to support IP multicast routing. The enhancements have been added in a backward-compatible fashion; routers running the multicast additions will interoperate with non-multicast OSPF routers when forwarding regular (unicast) IP data traffic.

MOSPF works by including multicast information in OSPF link state advertisements. An MOSPF router learns which multicast groups are active on which LANs. MOSPF builds a distribution tree for each source/group pair and computes a tree for active sources sending to the group. The tree state is cached, and trees must be recomputed when a link state change occurs or when the cache times out.

MOSPF provides the ability to forward multicast datagrams from one IP network to another through internet routers. MOSPF forwards a multicast datagram on the basis of both the datagram's source and destination. The OSPF link state database provides a complete description of the Autonomous System's topology. By adding a new type of link state advertisement, the group-membership-LSA, the location of all multicast group members is pinpointed in the database. The path of a multicast datagram can then be calculated by building a shortest-path tree rooted at the datagram's source. All branches not containing multicast members are pruned from the tree. These pruned shortest-path trees are initially built when the first datagram is received. The results of the shortest path calculation are then cached for use by subsequent datagrams having the same source and destination.

MOSPF is used internal to a single Autonomous System. When supporting IP multicast over the entire Internet, MOSPF would have to be used in concert with an inter-AS multicast routing protocol such as DVMRP.

Routers running MOSPF works only in internetworks that are using MOSPF but can be intermixed with non-multicast OSPF routers. Both types of routers can interoperate when forwarding regular (unicast) IP data traffic. In MOSPF, just as in the base OSPF protocol, datagrams (multicast or unicast) are routed "as is"; they are not further encapsulated or decapsulated as they transit the Autonomous System.

Protocol Structure

The MOSPF packet formats are the same as for OSPF Version 2. One additional option has been added to the Options field that appears in OSPF Hello packets, Database Description packets and all link state advertisements. This new option indicates a router's/network's multicast capability. The presence of this new option is ignored by all non-multicast routers.

1	2	3	4	5	6	7	8bit
*	*	*	*	*	MC	E	T

- T-bit – describes the router's TOS capability.
- E-bit – AS external link advertisements are not flooded into/through OSPF sub areas. The E-bit ensures that all members of a stub area agree on that area's configuration.
- MC-bit – describes the multicast capability of the various pieces of the OSPF routing domain.

To support MOSPF, one of OSPF's link state advertisements has been modified, and a new link state advertisement has been added. The format of the router-LSA has been modified to include a new flag indicating whether the router is a wild-card multicast receiver.

The rtype field in the router LSA:

1	2	3	4	5	6	7	8bit
*	*	*	*	W	V	E	B

- bit B - B is for border . When set, the router is an area border router. These routers forward unicast data traffic between OSPF areas.
- bit E - E is for external. When set, the router is an AS boundary router (). These routers forward unicast data traffic between Autonomous Systems.
- bit V - V is for virtual. When set, the router is an endpoint of an active virtual link which uses the described area as its Transit area.
- bit W - When set, the router is a wild-card multicast receiver. These routers receive all multicast datagrams, regardless of destination. Inter-area multicast forwarders and inter-AS multicast forwarders are sometimes wild-card multicast receivers.

A new link state advertisement, called the group-membership-LSA, has been added to pinpoint multicast group members in the link state database. This new advertisement is neither flooded nor processed by non-multicast routers.

Related protocols

IP, TCP, OSPF, IGMP

Sponsor Source

MOSPF is defined by IETF (http://www.ietf.org) in RFC 1584.

Reference

http://www.javvin.com/protocol/rfc1584.pdf
Multicast Extensions to OSPF
http://www.javvin.com/protocol/rfc1585.pdf
MOSPF: Analysis and Experience

Protocol Name

MSDP: Multicast Source Discovery Protocol

Protocol Description

The Multicast Source Discovery Protocol (MSDP) describes a mechanism to connect multiple PIM Sparse-Mode (PIM-SM) domains together. Each PIM-SM domain uses its own independent RP(s) and does not have to depend on RPs in other domains. Advantages of this approach include:

No Third-party resource dependencies on a domain's RP PIM-SM domains can rely on their own RPs only.

Receiver only Domains: Domains with only receivers get data without globally advertising group membership.

MSDP may be used with protocols other than PIM-SM.

MSDP-speaking routers in a PIM-SM domain have an MSDP peering relationship with MSDP peers in another domain. The peering relationship is made up of a TCP connection in which control information is exchanged. Each domain has one or more connections to this virtual topology.

The purpose of this topology is to allow domains to discover multicast sources from other domains. If the multicast sources are of interest to a domain which has receivers, the normal source-tree building mechanism in PIM-SM will be used to deliver multicast data over an inter-domain distribution tree.

Protocol Structure

MSDP TLV format

8	24bit	Variable
Type	Length	Value

- Type - Describes the format of the Value field. The following TLV Types are defined:

Code	*Type*
1	IPv4 Source-Active
2	IPv4 Source-Active Request
3	IPv4 Source-Active Response
4	KeepAlive
5	Reserved (Previously: Notification)
6	MSDP traceroute in progress
7	MSDP traceroute reply

- Length - Length of Type, Length, and Value fields in octets. Minimum length required is 4 octets, except for Keepalive messages. The maximum TLV length is 9192.
- Value (variable length) - Format is based on the Type value. The length of the value field is Length field minus 3. All reserved fields in the Value field MUST be transmitted as zeros and ignored on receipt.

Related protocols

IP, TCP, BGP, PIM-SM, PIM-DM

Sponsor Source

MSDP is circulated by IETF (http://www.ietf.org) as an experimental protocol.

Reference

http://www.javvin.com/protocol/rfc3618.pdf
Multicast Source Discovery Protocol

Protocol Name

MZAP: Multicast-Scope Zone Announcement Protocol

Protocol Description

Multicast-Scope Zone Announcement Protocol (MZAP) is for the discovery of the multicast administrative scope zones that are relevant at a particular location. MZAP also provides mechanisms to discover common misconfigurations of administrative scope zones.

The use of administratively-scoped IP multicast allows packets to be addressed to a specific range of multicast addresses such that the packets will not cross configured administrative boundaries, and also allows such addresses to be locally assigned and hence are not required to across administrative boundaries.

The range of administratively-scoped addresses can be subdivided by administrators so that multiple levels of administrative boundaries can be simultaneously supported. As a result, a “multicast scope” is defined as a particular range of addresses which has been given some topological meaning.

Multicast Scope Zone Announcement Protocol (MZAP) allows an entity to learn what scope zones it is within. Typically servers will cache the information learned from MZAP and can then provide this information to applications in a timely fashion upon request using other means, e.g., via MADCAP. MZAP also provides diagnostic information to the boundary routers themselves that enables misconfigured scope zones to be detected.

All MZAP messages are sent over UDP, with a destination port of [MZAP-PORT] and an IPv4 TTL or IPv6 Hop Limit of 255.

Protocol Structure

8	9	16	24	32bit
Version	B	PTYPE	Address Family	NameCount
Message Origin				
Zone ID Address				
Zone Start Address				
Zone End Address				
Encoded Zone Name-1 (variable length)				
...				
	Encoded Zone Name-N (variable length)			
			Padding (if needed)	

- Version - The version number; currently defined as 0.
- B - Big Scope bit. 0 Indicates that the addresses in the scoped range are not subdividable, and that address allocators may utilize the entire range. If 1, address allocators should not use the entire range, but should learn an appropriate sub- range via another mechanism.

  Packet Type - The packet types defined are:
  0: Zone Announcement Message (ZAM)
  1: Zone Limit Exceeded (ZLE)
  2: Zone Convexity Message (ZCM)
  3: Not-Inside Message (NIM)
- Address Family - Identifies the address family for all addresses in the packet. The families defined for IP are: 1: IPv4; 2: IPv6.
- Name Count - The number of encoded zone name blocks in this packet. The count may be zero.
- Message Origin - The IP address of the interface that originated the message.
- Zone Start Address - The start address for the scope zone boundary. For example, if the zone is a boundary for 239.1.0.0 to 239.1.0.255, then the Zone Start Address is 239.1.0.0.
- Zone End Address - The ending address for the scope zone boundary. For example, if the zone is a boundary for 239.1.0.0 to 239.1.0.255, then the Zone End Address is 239.1.0.255.
- Zone ID Address - The lowest IP address of a boundary router that has been observed in the zone originating the message. Together with the Zone Start Address and Zone End Address, it forms a unique ID for the zone. Note that this ID is usually different from the ID of the Local Scope zone in which the origin resides.
- Encoded Zone Name - Combined from the next fields: D, LangLen, Language Tag, NameLen, Zone Name.

Related protocols

IP, IPv6, UDP

Sponsor Source

MZAP is defined by IETF (http://www.ietf.org) in RFC 2776.

Reference

http://www.javvin.com/protocol/rfc2776.pdf
Multicast-Scope Zone Announcement Protocol (MZAP)

Protocol Name

PGM: Pragmatic General Multicast Protocol

Protocol Description

Pragmatic General Multicast (PGM) is a reliable transport protocol for applications that require ordered or unordered, duplicate-free, multicast data delivery from multiple sources to multiple receivers.

PGM is specifically intended as a workable solution for multicast applications with basic reliability requirements rather than as a comprehensive solution for multicast applications with sophisticated ordering, agreement, and robustness requirements. Its central design goal is simplicity of operation with due regard for scalability and network efficiency.

PGM has no notion of group membership. It simply provides reliable multicast data delivery within a transmit window advanced by a source according to a purely local strategy. Reliable delivery is provided within a source's transmit window from the time a receiver joins the group until it departs. PGM guarantees that a receiver in the group either receives all data packets from transmissions and repairs, or is able to detect unrecoverable data packet loss. PGM supports any number of sources within a multicast group, each fully identified by a globally unique Transport Session Identifier (TSI), but since these sources/sessions operate entirely independently of each other, this specification is phrased in terms of a single source and extends without modification to multiple sources.

More specifically, PGM is not intended for use with applications that depend either upon acknowledged delivery to a known group of recipients, or upon total ordering amongst multiple sources. Rather, PGM is best suited to those applications in which members may join and leave at any time, and that are either insensitive to unrecoverable data packet loss or are prepared to resort to application recovery in the event. Through its optional extensions, PGM provides specific mechanisms to support applications as disparate as stock and news updates, data conferencing, low-delay real-time video transfer, and bulk data transfer.

Protocol Structure

PGM header:

<table>
<tr><th colspan="2">16</th><th>32bit</th></tr>
<tr><td colspan="2">Source Port</td><td>Destination Port</td></tr>
<tr><td>Flags</td><td>Options</td><td>Checksum</td></tr>
<tr><td colspan="3">Global Source ID</td></tr>
<tr><td>Global Source ID</td><td colspan="2">TSDU Length</td></tr>
<tr><td colspan="3">Data :::</td></tr>
</table>

- Source Port - Data-Destination Port
- Destination Port. Data-Source Port
- Flags – Here are the bits definitions:

1	2	3	4	5	6	7	8
Version		0	0	Type			

- Version - PGM version number.
- Type – Type of message
- Options – Here are the bits definitions:

1	2	3	4	5	6	7	8
E	N					T	P

E Option Extensions. 1 bit.
N Options are network-significant. 1 bit.
T Packet is a parity packet for a transmission group of variable sized packets. 1 bit.
P Packet is a parity packet. 1 bit.

- Checksum – Error checking.
- Global Source ID - A globally unique source identifier.
- TSDU Length - The length in bytes of the transport data unit exclusive of the transport header.
- Data - Variable length.

Related protocols

IP, TCP

Sponsor Source

PGM is circulated by IETF (http://www.ietf.org) as an experimental protocol.

Reference

http://www.javvin.com/protocol/rfc3208.pdf
PGM Reliable Transport Protocol Specification

Protocol Name

PIM-DM: Protocol Independent Multicast – Dense Mode

Protocol Description

Protocol Independent Multicast (PIM) has two modes: Sparse Mode and Dense Mode. We focus on the Dense Mode in this document.

PIM-DM is mainly designed for multicast LAN applications, while the PIM-SM is for wide area, inter-domain networks. PIM-DM implements the same flood-and-prune mechanism that Distance Vector Multicast Routing Protocol (DVMRP) and other dense mode routing protocols employ. The main difference between DVMRP and PIM-DM is that PIM-DM introduces the concept of protocol independence. PIM-DM can use the routing table populated by any underlying unicast routing protocol to perform reverse path forwarding (RPF) checks.

ISPs typically appreciate the ability to use any underlying unicast routing protocol with PIM-DM because they need not introduce and manage a separate routing protocol just for RPF checks. Unicast routing protocols extended as Multiprotocol Extensions to BGP (MBGP) and Multitopology Routing for IS-IS (M-ISIS) were later employed to build special tables to perform RPF checks, but PIM-DM does not require them.

PIM-DM can use the unicast routing table populated by OSPF, IS-IS, BGP, and so on, or PIM-DM can be configured to use a special multicast RPF table populated by MBGP or M-ISIS when performing RPF checks.

Protocol Structure

The protocol format of PIM-DM is the same as that of PIM-SM:

PIM version	Type	Reserved (Address length)	Checksum

- PIM version – The current PIM version is 2.
- Type -- Types for specific PIM messages.
- Address length -- Address length in bytes. The length of the address field throughout, in the specific message.
- Reserved - The value of this field is set to 0, ignore on receipt
- Checksum - The 16-bit field is the one's complement sum of the entire PIM message.

Related protocols

PIM-SM, ICMP, RIP, OSPF, DVMRP, IS-IS, BGP, IGRP, EIGRP

Sponsor Source

PIM-DM has been discussed but yet not finalized by IETF (http://www.ietf.org) yet.

Reference

http://www.javvin.com/protocol/rfcPIM-DM.pdf
PIM-DM: Protocol Specification Draft
http://www.javvin.com/protocol/rfcPIMDM-refresh.pdf
PIM-DM Refresh Draft

Protocol Name

PIM-SM: Protocol Independent Multicast-Sparse Mode

Protocol Description

Protocol Independent Multicast (PIM) has two modes: Sparse Mode and Dense Mode. We focus on the Sparse Mode in this document.

PIM-SM is a protocol for efficiently routing to multicast groups that may span wide-area (WAN and inter-domain) internets, while PIM-DM is mainly for LAN. The protocol is not dependent on any particular unicast routing protocol, and is designed to support sparse groups. It uses the traditional IP multicast model of receiver-initiated membership, supports both shared and shortest-path trees, and uses soft-state mechanisms to adapt to changing network conditions. It can use the route information that any routing protocol enters into the multicast Routing Information Base (RIB). Examples of these routing protocols include unicast protocols such as the Routing Information Protocol (RIP) and Open Shortest Path First (OSPF), but multicast protocols that populate the routing tables—such as the Distance Vector Multicast Routing Protocol (DVMRP)—can also be used.

PIM-SM was designed to support the following goals:

- Maintain the traditional IP multicast service model of receiver-initiated multicast group membership. In this model, sources simply put packets on the first-hop Ethernet, without any signaling. Receivers signal to routers in order to join the multicast group that will receive the data.
- Leave the host model unchanged. PIM-SM is a router-to-router protocol, which means that hosts don't have to be upgraded, but that PIM-SM-enabled routers must be deployed in the network.
- Support both shared and source distribution trees. For shared trees, PIM-SM uses a central router, called the Rendezvous Point (RP), as the root of the shared tree. All source hosts send their multicast traffic to the RP, which in turn forwards the packets through a common tree to all the members of the group. Source trees directly connect sources to receivers. There is a separate tree for every source. Source trees are considered shortest-path trees from the perspective of the unicast routing tables. PIM-SM can use either type of tree or both simultaneously.
- Maintain independence from any specific unicast routing protocol (see above).
- Use soft-state mechanisms to adapt to changing network conditions and multicast group dynamics. Soft-state means that, unless it is refreshed, the router's state configuration is short-term and expires after a certain amount of time.

Currently, there are two versions of PIM-SM. We focus on version 2, which is widely deployed.

Protocol Structure

The protocol format of PIM-DM is the same as that of PIM-SM:

PIM version	Type	Reserved (Address length)	Checksum

- PIM version – The current PIM version is 2.
- Type -- Types for specific PIM messages.
- Address length -- Address length in bytes. The length of the address field throughout, in the specific message.
- Reserved - The value of this field is set to 0, ignore on receipt
- Checksum - The 16-bit field is the one's complement sum of the entire PIM message.

Related protocols

PIM-DM, ICMP, RIP, OSPF, DVMRP, IS-IS, BGP, IGRP, EIGRP

Sponsor Source

PIM-SM is defined by IETF (http://www.ietf.org) RFC2362.

Reference

http://www.javvin.com/protocol/rfc2362.pdf
PIM-SM: Protocol Specification

MPLS Protocals

Protocol Name

MPLS: Multiprotocol Label Switching

Protocol Description

Multiprotocol Label Switching (MPLS), an architecture for fast packet switching and routing, provides the designation, routing, forwarding and switching of traffic flows through the network. More specifically, it has mechanisms to manage traffic flows of various granularities. It is independent of the layer-2 and layer-3 protocols such as ATM and IP. It provides a means to map IP addresses to simple, fixed-length labels used by different packet-forwarding and packet-switching technologies. It interfaces to existing routing and switching protocols, such as IP, ATM, Frame Relay, Resource ReSerVation Protocol (RSVP) and Open Shortest PathFirst (OSPF), etc.

In MPLS, data transmission occurs on Label-Switched Paths (LSPs). LSPs are a sequence of labels at each and every node along the path from the source to the destination. There are several label distribution protocols used today, such as Label Distribution Protocol (LDP) or RSVP or piggybacking on routing protocols like border gateway protocol (BGP) and OSPF. High-speed switching of data is possible because the fixed-length labels are inserted at the very beginning of the packet or cell and can be used by hardware to switch packets quickly between links.

MPLS is designed to address network problems such as networks-speed, scalability, quality-of-service (QoS) management, and traffic engineering. MPLS has also become a solution to the bandwidth-management and service requirements for next-generation IP-based backbone networks.

In this section, we focus on the general MPLS framework. LDP, CR-LDP and RSVP-TE will be discussed in separate documents.

Protocol Structure

MPLS label structure:

20	23	24	32bit
Label	Exp	S	TTL

- Label - Label Value carries the actual value of the Label. When a labeled packet is received, the label value at the top of the stack is looked up and the system learns:
  a) the next hop to which the packet is to be forwarded;
  b) the operation to be performed on the label stack before forwarding; this operation may be to replace the top label stack entry with another, or to pop an entry off the label stack, or to replace the top label stack entry and then to push one or more additional entries on the label stack.
- Exp - Experimental Use: Reserved for experimental use.
- S - Bottom of Stack: This bit is set to one for the last entry in the label stack, and zero for all other label stack entries
- TTL - Time to Live field is used to encode a time-to-live value.

The MPLS architecture protocol family includes:

- MPLS related Signalling Protocols such as OSPF, BGP, ATM PNNI, etc.
- LDP: Label Distribution Protocol.
- CR-LDP: Constraint-Based LDP
- RSVP-TE: Resource Reservation Protocol – Traffic Engineering

The following figure depicts the MPLS protocol stack architecture:

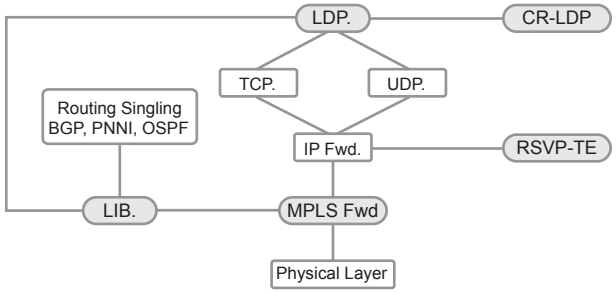


Figure 2-4: MPLS protocol stack architecture

The structure of each protocol will be discussed in separate documents.

Related protocols

LDP, CR-LDP, RSVP-TE, IP, ATM, RSVP, OSPF

Sponsor Source

MPLS is defined by IETF (http://www.ietf.org) RFC3031 and RFC 3032.

Reference

http://www.javvin.com/protocol/rfc3031.pdf
Multiprotocol Label Switching Architecture
http://www.javvin.com/protocol/rfc3032.pdf
MPLS Label Stack Encoding
http://www.javvin.com/protocol/rfc3443.pdf
Time To Live (TTL) Processing in Multi-Protocol Label Switching (MPLS) Networks
http://www.javvin.com/protocol/rfc3036.pdf

LDP Specification
http://www.javvin.com/protocol/rfc3209.pdf
RSVP-TE: Extensions to RSVP for LSP Tunnels
http://www.javvin.com/protocol/rfc3212.pdf
Constraint-Based LSP Setup using LDP
http://www.javvin.com/protocol/rfc3213.pdf
Applicability Statement for CR-LDP

Protocol Name

CR-LDP : Constraint-based LDP

Protocol Description

CR-LDP, constraint-based LDP, is one of the protocols in the MPLS architecture. It contains extensions for LDP to extend its capabilities such as setup paths beyond what is available for the routing protocol. For instance, an LSP (Label Switched Path) can be setup based on explicit route constraints, QoS constraints, and other constraints. Constraint-based routing (CR) is a mechanism used to meet Traffic Engineering requirements. These requirements are met by extending LDP for support of constraint-based routed label switched paths (CR-LSPs). Other uses for CR-LSPs include MPLS-based VPNs.

Protocol Structure

CR-LDP has the same structure as LDP except for the following additional TLV parameters.

Value	*Parameter*
821	LSPID
822	ResCls
503	Optical Session Parameters
800	Explicit Route
801-804	ER-Hop TLVS
810	Traffic Parameters
820	Preemption
823	Route Pinning
910	Optical Interface Type
920	Optical Trail Desc
930	Optical Label
940	Lambada Set

Related protocols

MPLS, LDP, RSVP-TE, IP, ATM, RSVP, OSPF

Sponsor Source

CR-LDP is specified by IETF (http://www.ietf.org) RFC3212.

Reference

http://www.javvin.com/protocol/rfc3031.pdf
Multiprotocol Label Switching Architecture
http://www.javvin.com/protocol/rfc3036.pdf
LDP Specification
http://www.javvin.com/protocol/rfc3212.pdf
Constraint-Based LSP Setup using LDP
http://www.javvin.com/protocol/rfc3213.pdf
Applicability Statement for CR-LDP

Protocol Name

LDP: Label Distribution Protocol

Protocol Description

LDP (Label Distribution Protocol) is a key protocol in the MPLS (Multi Protocol Label Switching) architecture. In the MPLS network, 2 label switching routers (LSR) must agree on the meaning of the labels used to forward traffic between and through them. LDP defines a set of procedures and messages by which one LSR (Label Switched Router) informs another of the label bindings it has made. The LSR uses this protocol to establish label switched paths through a network by mapping network layer routing information directly to data-link layer switched paths.

Two LSRs (Label Switched Routers) which use LDP to exchange label mapping information are known as LDP peers and they have an LDP session between them. In a single session, each peer is able to learn about the others label mappings, in other words, the protocol is bi-directional.

Protocol Structure

2 bytes	2 bytes
Version	PDU Length
LDP Identifier (6 bytes)	
LDP Messages	

- Version -- The protocol version number. The present number is 1.
- PDU Length -- The total length of the PDU excluding the version and the PDU length field.
- LDP identifier -- This field uniquely identifies the label space of the sending LSR for which this PDU applies. The first 4 octets encode the IP address assigned to the LSR. The lst 2 indicate a label space within the LSR.

LDP messages -- All LDP messages have the following format:

U	Message type	Message length
Message ID		
Parameters		

- U -- The U bit is an unknown message bit.
- Message type -- The type of message. The following message types exist: Notification, Hello, Initialization, Keep Alive, Address, Address Withdraw, Label Request, Label Withdraw, Label Release, and Unknown Message name.
- Message length -- The length in octets of the message ID, mandatory parameters and optional parameters
- Message ID -- 32-bit value used to identify the message.
- Parameters -- The parameters contain the TLVs. There are both mandatory and optional parameters. Some messages have no mandatory parameters, and some have no optional parameters.

TLV format:

U	F	Type	Length
Value			
TLV format			

- U -- The U bit is an unknown TLV bit.
- F -- Forward unknown TLV bit.
- Type -- Encodes how the Value field is to be interpreted.
- Length -- Specifies the length of the Value field in octets.
- Value -- Octet string of Length octets that encodes information to be interpreted as specified by the Type field.

Related protocols

MPLS, CR-LDP, RSVP-TE, IP, ATM, RSVP, OSPF

Sponsor Source

LDP is specified by IETF (http://www.ietf.org) RFC3036.

Reference

http://www.javvin.com/protocol/rfc3031.pdf
Multiprotocol Label Switching Architecture
http://www.javvin.com/protocol/rfc3036.pdf
LDP Specification

Protocol Name

RSVP-TE: Resource Reservation Protocol - Traffic Extension

Protocol Description

The RSVP-TE protocol is an addition to the RSVP protocol for establishing label switched paths (LSPs) in MPLS networks. The extended RSVP protocol supports the instantiation of explicitly routed LSPs, with or without resource reservations. It also supports smooth rerouting of LSPs, preemption, and loop detection.

The RSVP protocol defines a session as a data flow with a particular destination and transport-layer protocol. However, when RSVP and MPLS are combined, a flow or session can be defined with greater flexibility and generality. The ingress node of an LSP (Label Switched Path) uses a number of methods to determine which packets are assigned a particular label. Once a label is assigned to a set of packets, the label effectively defines the flow through the LSP. We refer to such an LSP as an LSP tunnel because the traffic through it is opaque to intermediate nodes along the label switched path. New RSVP Session, Sender and Filter Spec objects, called LSP Tunnel IPv4 and LSP Tunnel IPv6 have been defined to support the LSP tunnel feature. The semantics of these objects, from the perspective of a node along the label switched path, is that traffic belonging to the LSP tunnel is identified solely on the basis of packets arriving from the “previous hop” (PHOP) with the particular label value(s) assigned by this node to upstream senders to the session. In fact, the IPv4(v6) that appears in the object name only denotes that the destination address is an IPv4(v6) address. When referring to these objects generically, the qualifier LSP Tunnel is used.

In some applications it is useful to associate sets of LSP tunnels, such as during reroute operations or in spreading a traffic trunk over multiple paths, such sets are called TE tunnels. To enable the identification and association of the LSP tunnels, two identifiers are carried. A tunnel ID is part of the Session object. The Session object uniquely defines a traffic engineered tunnel. The Sender and Filter Spec objects carry an LSP ID. The Sender (or Filter Spec) object, together with the Session object, uniquely identifies an LSP tunnel.

Protocol Structure

Apart from the existing message types listed in RSVP an additional message type is available:

Value	*Message type*
14	Hello

In addition, the following additional Protocol Object Types exist:

Value	*Object type*
16	Label
19	Optical
20	Explicit Route
21	Record Route
22	Hello
207	Attribute Session

Related protocols

MPLS, LDP, CR-LDP, IP, ATM, RSVP, OSPF

Sponsor Source

RSVP-TE is defined by IETF (http://www.ietf.org) RFC3209.

Reference

http://www.javvin.com/protocol/rfc3031.pdf
Multiprotocol Label Switching Architecture
http://www.javvin.com/protocol/rfc3209.pdf
RSVP-TE: Extensions to RSVP for LSP Tunnels

Data Link Layer Protocols

Protocol Name

ARP and InARP: Address Resolution Protocol and Inverse ARP

Protocol Description

Address Resolution Protocol (ARP) performs mapping of an IP address to a physical machine address (MAC address for Ethernet) that is recognized in the local network. For example, in IP Version 4, an address is 32 bits long. In an Ethernet local area network, however, addresses for attached devices are 48 bits long. A table, usually called the ARP cache, is used to maintain a correlation between each MAC address and its corresponding IP address. ARP provides the rules for making this correlation and providing address conversion in both directions.

Since protocol details differ for each type of local area network, there are separate ARP specifications for Ethernet, Frame Relay, ATM, Fiber Distributed-Data Interface, HIPPI, and other protocols. InARP is an addition to ARP to address ARP in Frame Relay environment.

There is a Reverse ARP (RARP) for host machines that don't know their IP address. RARP enables them to request their IP address from the gateway's ARP cache. Details of RARP are presented in a separate document.

Protocol Structure

ARP and InARP have the same structure:

16		32 bit
Hardware Type		Protocol Type
HLen	Plen	Operation
Sender Hardware Address		
Sender Protocol Address		
Target Hardware Address		
Target Protocol Address		

- Hardware type - Specifies a hardware interface type for which the sender requires a response.
- Protocol type - Specifies the type of high-level protocol address the sender has supplied.
- Hlen - Hardware address length.
- Plen - Protocol address length.
- Operation - The values are as follows:
  1 ARP request.
  2 ARP response.
  3 RARP request.
  4 RARP response.
  5 Dynamic RARP request.
  6 Dynamic RARP reply.
  7 Dynamic RARP error.
  8 InARP request.
  9 InARP reply.
- Sender hardware address -HLen bytes in length.
- Sender protocol address - PLen bytes in length.
- Target hardware address - HLen bytes in length.
- Target protocol address - PLen bytes in length.

Related protocols

ARP, RARP, InARP

Sponsor Source

ARP/IARP are defined by IETF (http://www.ietf.org) in RFC 826, 2390, 2625.

Reference

http://www.javvin.com/protocol/rfc826.pdf
An Ethernet Address Resolution Protocol
http://www.javvin.com/protocol/rfc2390.pdf
Inverse Address Resolution Protocol (Frame Relay)
http://www.javvin.com/protocol/rfc2625.pdf
IP and ARP over Fibre Channel

Protocol Name

IPCP and IPv6CP: IP Control Protocol and IPv6 Control Protocol

Protocol Description

IP Control Protocol (IPCP) and IPv6 Control Protocol (IPv6CP) define the Network Control Protocol for establishing and configuring the Internet Protocol or IPv6 over PPP, and a method to negotiate and use Van Jacobson TCP/IP header compression with PPP.

IPCP is responsible for configuring, enabling, and disabling the IP protocol modules on both ends of the point-to-point link. IPCP uses the same packet exchange mechanism as the Link Control Protocol (LCP). IPCP packets may not be exchanged until PPP has reached the Network-Layer Protocol phase. IPCP packets received before this phase is reached should be silently discarded.

Before any IP packets may be communicated, PPP must reach the Network-Layer Protocol phase, and the IP Control Protocol must reach the Opened state.

Van Jacobson TCP/IP header compression reduces the size of the TCP/IP headers to as few as three bytes. This can be a significant improvement on slow serial lines, particularly for interactive traffic.

The IP Compression Protocol Configuration Option is used to indicate the ability to receive compressed packets. Each end of the link must separately request this option if bidirectional compression is desired.

IPv6CP is responsible for configuring, enabling, and disabling the IPv6 protocol modules on both ends of the point-to-point link. IPv6CP uses the same packet exchange mechanism as the Link Control Protocol (LCP). IPv6CP packets may not be exchanged until PPP has reached the Network-Layer Protocol phase. IPv6CP packets received before this phase is reached should be silently discarded.

Protocol Structure

IPCP and IPv6CP configuration option packet header:

8	16	32bit
Type	Length	Configuration Option

- Type – 1 for IP-Address, 2 for IP-Compression Protocol, and 3 for IP-Address
- Length >= 4
- Configuration Option - The field is two octets and indicates one of the following options:

For IPCP:
Type 1: IP-Addresses
Type 2: IP-Compression Protocol
Type 3: IP-Address.

For IPv6CP:
Type 1: Interface – Identifier
Type 2: IPv6-Compression Protocol

IPCP and IPv6CP header structure:

8	16	32bit
Code	Identifier	Length
Data (variable)		

- Code - Specifies the function to be performed.
- Identifier - Used to match requests and replies.
- Length - Size of the packet including the header.
- Data -Zero or more bytes of data as indicated by the Length. This field may contain one or more Options.

Related protocols

IP, IPv6, PPP, TCP, Van Jacobson

Sponsor Source

IPCP is defined by IETF (http://www.ietf.org) in RFC 1332 and IPv6CP is defined in RFC 2472.

Reference

http://www.javvin.com/protocol/rfc1332.pdf
The PPP Internet Protocol Control Protocol (IPCP).
http://www.javvin.com/protocol/rfc2472.pdf
IP Version 6 over PPP
http://www.javvin.com/protocol/rfc3241.pdf
Robust Header Compression (ROHC) over PPP.
http://www.javvin.com/protocol/rfc3544.pdf
IP Header Compression over PPP.

Protocol Name

RARP: Reverse Address Resolution Protocol

Protocol Description

Reverse Address Resolution Protocol (RARP) allows a physical machine in a local area network to request its IP address from a gateway server's Address Resolution Protocol (ARP) table or cache. A network administrator creates a table in a local area network's gateway router that maps the physical machines' (or Media Access Control - MAC) addresses to corresponding Internet Protocol addresses. When a new machine is set up, its RARP client program requests its IP address from the RARP server on the router. Assuming that an entry has been set up in the router table, the RARP server will return the IP address to the machine, which can store it for future use.

RARP is available for Ethernet, Fiber Distributed-Data Interface, and Token Ring LANs.

Protocol Structure

The protocol header for RARP is the same as for ARP:

16		32bit
Hardware Type		Protocol Type
Hlen	Plen	Operation
Sender Hardware Address		
Sender Protocol Address		
Target Hardware Address		
Target Protocol Address		

- Hardware type - Specifies a hardware interface type for which the sender requires a response.
- Protocol type - Specifies the type of high-level protocol address the sender has supplied.
- Hlen - Hardware address length.
- Plen - Protocol address length.
- Operation - The values are as follows:
    1 ARP request.
    2 ARP response.
    3 RARP request.
    4 RARP response.
    5 Dynamic RARP request.
    6 Dynamic RARP reply.
    7 Dynamic RARP error.
    8 InARP request.
    9 InARP reply.
- Sender hardware address -HLen bytes in length.
- Sender protocol address - PLen bytes in length.
- Target hardware address - HLen bytes in length.
- Target protocol address - PLen bytes in length.

Related protocols

ARP, RARP, InARP

Sponsor Source

RARP is defined by IETF (http://www.ietf.org) in RFC 903.

Reference

http://www.javvin.com/protocol/rfc903.pdf
Reverse Address Resolution Protocol

Protocol Name

SLIP: Serial Line IP

Protocol Description

Serial Line IP (SLIP) is used for point-to-point serial connections running TCP/IP. SLIP is commonly used on dedicated serial links and sometimes for dialup purposes, and is usually used with line speeds between 1200bps and 19.2Kbps. SLIP is useful for allowing mixes of hosts and routers to communicate with one another (host-host, host-router and router-router are all common SLIP network configurations).

SLIP is merely a packet framing protocol: SLIP defines a sequence of characters that frame IP packets on a serial line. It does not provide addressing, packet type identification, error detection/correction or compression mechanisms.

The SLIP protocol defines two special characters: END and ESC. END is octal 300 (decimal 192) and ESC is octal 333 (decimal 219). To send a packet, a SLIP host simply starts sending the data in the packet. If a data byte is the same code as the END character, a two byte sequence of ESC and octal 334 (decimal 220) is sent instead. If it the same as an ESC character, a two byte sequence of ESC and octal 335 (decimal 221) is sent instead. When the last byte in the packet has been sent, an END character is then transmitted.

Because there is no ‘standard’ SLIP specification, there is no real defined maximum packet size for SLIP. It is probably best to accept the maximum packet size used by the Berkeley UNIX SLIP drivers: 1006 bytes including the IP and transport protocol headers (not including the framing characters).

Compressed Serial Line IP (CSLIP) performs the Van Jacobson header compression on outgoing IP packets. This compression improves throughput for interactive sessions noticeably.

Today, SLIP is largely replaced by the Point–to-Point Protocol (PPP), which is more feature rich and flexible.

Related protocols

IP, TCP, PPP, Van Jacobson

Sponsor Source

SLIP is defined by IETF (http://www.ietf.org).

Reference

http://www.javvin.com/protocol/rfc1055.pdf
A Nonstandard for Transmission of IP Datagramsover serial Lines: SLIP

Network Security Technologies and Protocols

Description

Network security covers such issues as network communication privacy, information confidentiality and integrity over network, controlled access to restricted network domains and sensitive information, and using the public network such as Internet for private communications. To address these issues, various network and information security technologies have been developed by various organizations and technology vendors. Here is a summary of the technologies:

AAA: Authorization, Authentication and Accounting is a technology for intelligently controlling access to network resources, enforcing policies, auditing usage, and providing the information necessary to bill for services. Authentication provides a way of identifying a user, typically by having the user enter a valid user name and valid password before access is granted. The authorization process determines whether the user has the authority to access certain information or some network sub-domains. Accounting measures the resources a user consumes while using the network, which includes the amount of system time or the amount of data a user has sent and/or received during a session, which could be used for authorization control, billing, trend analysis, resource utilization, and capacity planning activities. A dedicated AAA server or a program that performs these functions often provides authentication, authorization, and accounting services.

VPN: Virtual Private Network is a technology allowing private communications by business and individuals, such as remote access to a corporate network or using a public telecommunication infrastructure, such as the Internet. A virtual private network can also be a specially configured network over the public network infrastructure that is only used by one organization. Various network-tunneling technologies such as L2TP have been developed to reach this goal. Using encryption technologies such as IPsec could further enhance information privacy over network and virtual private networks.

Firewall: Firewall is a software program or hardware device that filters the information coming through the Internet connection into a private network or computer system. Firewalls use one or more of three methods to control traffic flowing in and out the network:

- Packet filtering - Packets are analyzed against a set of filters. Packets that make it through the filters are sent to the requesting system and all others are discarded.
- Proxy service - Information from the Internet is retrieved by the firewall and then sent to the requesting system and vice versa.
- Stateful inspection - compares certain key parts of packets passing through with a database of trusted information. Outgoing information from inside the firewall is monitored for specific defining characteristics, and incoming information is then compared with these characteristics. If the comparison yields a reasonable match, the information is allowed through. Otherwise it is discarded.

Protocols

The key protocols for AAA and VPN:

Authentication	Kerberos: Network Authentication Protocol
Authorization	RADIUS: Remote Authentication Dial In User Service
Accounting	SSH: Secure Shell Protocol
Tunneling	L2F: Level 2 Forwarding protocol
	L2TP: Layer 2 Tunneling Protocol
	PPTP: Point to Point Tunneling Protocol
Secured Routing	DiffServ: Differentiated Service
	GRE: Generic Routing Encapsulation
	IPsec: Security Architecture for IP network
	IPsec AH: IPsec Authentication Header
	IPsec ESP: IPsecEncapsulating Security Payload
	IPsec IKE: Internet Key Exchange Protocol
	IPsec ISAKMP: Internet Security Association and Key Management Protocol
	TLS: Transport Layer Security Protocol
Others	Socks: Protocol for sessions traversal across firewall securely

Reference

http://www.cisco.com/univercd/cc/td/doc/cisintwk/ito_doc/security.htm
Securities Technologies

AAA Protocols

Protocol Name

Kerberos: Network Authentication Protocol

Protocol Description

Kerberos is a network authentication protocol. Kerberos is designed to provide strong authentication for client/server applications by using secret-key cryptography. This is accomplished without relying on authentication by the host operating system, without basing trust on host addresses, without requiring physical security of all the hosts on the network, and under the assumption that packets traveling along the network can be read, modified, and inserted at will. Kerberos performs authentication under these conditions as a trusted third-party authentication service by using conventional cryptography, i.e., shared secret key.

The authentication process proceeds as follows: A client sends a request to the authentication server (AS) requesting "credentials" for a given server. The AS responds with these credentials, encrypted in the client's key. The credentials consist of 1) a "ticket" for the server and 2) a temporary encryption key (often called a "session key"). The client transmits the ticket (which contains the client's identity and a copy of the session key, both encrypted in the server's key) to the server. The session key (now shared by the client and server) is used to authenticate the client, and may optionally be used to authenticate the server. It may also be used to encrypt further communication between the two parties or to exchange a separate sub-session key to be used to encrypt further communication.

The authentication exchanges mentioned above require read-only access to the Kerberos database. Sometimes, however, the entries in the database must be modified, such as when adding new principals or changing a principal's key. This is done using a protocol between a client and a third Kerberos server, the Kerberos Administration Server (KADM). The administration protocol is not described in this document. There is also a protocol for maintaining multiple copies of the Kerberos database, but this can be considered an implementation detail and may vary to support different database technologies.

Protocol Structure

Kerberos messages:

The Client/Server Authentication Exchange

Message direction	Message type
1. Client to Kerberos	KRB_AS_REQ
2. Kerberos to client	KRB_AS_REP or KRB_ERROR

The Client/Server Authentication Exchange

Message direction	Message type
Client to Application server	KRB_AP_REQ
[optional] Application server to client	KRB_AP_REP or KRB_ERROR

The Ticket-Granting Service (TGS) Exchange

Message direction	Message type
1. Client to Kerberos	KRB_TGS_REQ
2. Kerberos to client	KRB_TGS_REP or KRB_ERROR

The KRB_SAFE Exchange
The KRB_PRIV Exchange
The KRB_CRED Exchange

Related protocols

RADIUS, TACACS+

Sponsor Source

Kerberos is defined by MIT.

Reference

http://www.javvin.com/protocol/rfc1510.pdf
The Kerberos Network Authentication Service (V5)
http://www.javvin.com/protocol/rfc1964.pdf
The Kerberos Version 5 GSS-API Mechanism
http://web.mit.edu/kerberos/www/
Kerberos: The Network Authentication Protocol

Protocol Name

RADIUS: Remote Authentication Dial In User Service

Protocol Description

RADIUS is a protocol for carrying authentication, authorization, and configuration information between a Network Access Server which desires to authenticate its links and a shared Authentication Server. RADIUS uses UDP as the transport protocol. RADIUS also carries accounting information between a Network Access Server and a shared Accounting Server.

Key features of RADIUS are:

Client/Server Model: A Network Access Server (NAS) operates as a client of RADIUS. The client is responsible for passing user information to designated RADIUS servers, and then acting on the response which is returned. RADIUS servers are responsible for receiving user connection requests, authenticating the user, and then returning all configuration information necessary for the client to deliver service to the user. A RADIUS server can act as a proxy client to other RADIUS servers or other kinds of authentication servers.

Network Security: Transactions between the client and RADIUS server are authenticated through the use of a shared secret, which is never sent over the network. In addition, any user passwords are sent encrypted between the client and RADIUS server, to eliminate the possibility that someone snooping on an insecure network could determine a user's password.

Flexible Authentication Mechanisms: The RADIUS server can support a variety of methods to authenticate a user. When it is provided with the user name and original password given by the user, it can support PPP PAP or CHAP, UNIX login, and other authentication mechanisms.

Extensible Protocol: All transactions are comprised of variable length Attribute-Length-Value 3-tuples. New attribute values can be added without disturbing existing implementations of the protocol.

Protocol Structure

8	16	32 bit
Code	Identifier	Length
Authenticator (16 bytes)		

- Code - The message types are described as follows:
  - 1 Access-Request
  - 2 Access-Accept
  - 3 Access-Reject
  - 4 Accounting-Request
  - 5 Accounting-Response
  - 11 Access-Challenge
  - 12 Status-Server (experimental)
  - 13 Status-Client (experimental)
  - 255 Reserved
- Identifier - The identifier matches requests and replies.
- Length - The message length including the header.
- Authenticator - A field used to authenticate the reply from the radius server and in the password hiding algorithm.

Related protocols

UDP, CHAP, RAP

Sponsor Source

RADIUS is defined by IETF (http://www.ietf.org) in RFC 2865 and 2866.

Reference

http://www.javvin.com/protocol/rfc2865.pdf
Remote Authentication Dial In User Service (RADIUS)
http://www.javvin.com/protocol/rfc2866.pdf
RADIUS Accounting

Protocol Name

SSH: Secure Shell Protocol

Protocol Description

SSH is a protocol for secure remote login and other secure network services over an insecure network. SSH consists of three major components:

The Transport Layer Protocol [SSH-TRANS] provides server authentication, confidentiality, and integrity. It may optionally also provide compression. The transport layer will typically be run over a TCP/IP connection, but might also be used on top of any other reliable data stream. SSH-Trans provides strong encryption, cryptographic host authentication, and integrity protection. Authentication in this protocol level is host-based; this protocol does not perform user authentication. A higher level protocol for user authentication can be designed on top of this protocol.

The User Authentication Protocol [SSH-USERAUTH] authenticates the client-side user to the server. It runs over the transport layer protocol SSH-TRANS. When SSH-USERAUTH starts, it receives the session identifier from the lower-level protocol (this is the exchange hash H from the first key exchange). The session identifier uniquely identifies this session and is suitable for signing in order to prove ownership of a private key. SSH-USERAUTH also needs to know whether the lower-level protocol provides confidentiality protection.

The Connection Protocol [SSH-CONNECT] multiplexes the encrypted tunnel into several logical channels. It runs over the user authentication protocol. It provides interactive login sessions, remote execution of commands, forwarded TCP/IP connections, and forwarded X11 connections.

The client sends a service request once a secure transport layer connection has been established. A second service request is sent after user authentication is complete. This allows new protocols to be defined and coexist with the protocols listed above. The connection protocol provides channels that can be used for a wide range of purposes. Standard methods are provided for setting up secure interactive shell sessions and for forwarding (“tunneling”) arbitrary TCP/IP ports and X11 connections.

Protocol Structure

Secure Shell (SSH) protocols have many messages and each message may have different formats. For details of the message formats, please refer to the Reference documents listed below.

Related protocols

TCP

Sponsor Source

SSH is now drafted by IETF (http://www.ietf.org).

Reference

http://www.javvin.com/protocol/sshdraft15.pdf
SSH Protocol Architecture
http://www.javvin.com/protocol/sshtransport17.pdf
SSH Transport Layer Protocol
http://www.javvin.com/protocol/sshauth18.pdf
SSH User Authentication Protocol
http://www.javvin.com/protocol/sshconnect18.pdf
SSH Connection Protocol

Tunneling Protocols

Protocol Name

L2F: Layer 2 Forwarding Protocol

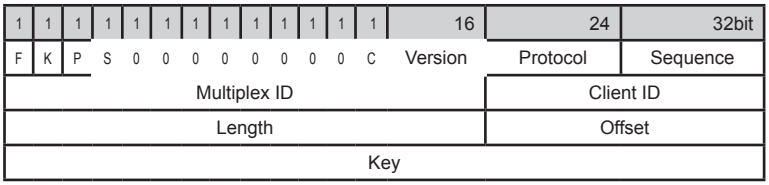
Protocol Description

The Layer 2 Forwarding protocol (L2F) is used to establish a secure tunnel across a public infrastructure (such as the Internet) that connects an ISP POP to an enterprise home gateway. This tunnel creates a virtual point-to-point connection between the user and the enterprise customer's network.

Layer Two Forwarding protocol (L2F) permits the tunneling of the link layer (i.e., HDLC, async HDLC, or SLIP frames) of higher level protocols. Using such tunnels, it is possible to divorce the location of the initial dial-up server from the location at which the dial-up protocol connection is terminated and access to the network provided.

L2F allows encapsulation of PPP/SLIP packets within L2F. The ISP NAS and the Home gateway require a common understanding of the encapsulation protocol so that SLIP/PPP packets can be successfully transmitted and received across the Internet.

Protocol Structure



- Version - The major version of the L2F software creating the packet.
- Protocol - The protocol field specifies the protocol carried within the L2F packet.
- Sequence - The sequence number is present if the S bit in the L2F header is set to 1.
- Multiplex ID - The packet multiplex ID identifies a particular connection within a tunnel.
- Client ID - The client ID (CLID) assists endpoints in demultiplexing tunnels.
- Length - The length is the size in octets of the entire packet, including the header, all the fields and the payload.
- Offset - This field specifies the number of bytes past the L2F header at which the payload data is expected to start. This field is present if the F bit in the L2F header is set to 1.
- Key - The key field is present if the K bit is set in the L2F header. This is part of the authentication process.
- Checksum - The checksum of the packet. The checksum field is present if the C bit in the L2F header is set to 1.

Related protocols

GRE, PPP, L2TP, PPTP, SLIP

Sponsor Source

L2F is defined by Cisco.

Reference

http://www.javvin.com/protocol/rfc2341.pdf
Cisco Layer Two Forwarding (Protocol) "L2F"

Protocol Name

L2TP: Layer 2 Tunneling Protocol

Protocol Description

The L2TP Protocol is used for integrating multi-protocol dial-up services into existing Internet Service Providers Point of Presence. PPP defines an encapsulation mechanism for transporting multiprotocol packets across layer 2 (L2) point-to-point links. Typically, a user obtains a L2 connection to a Network Access Server (NAS) using one of a number of techniques (e.g., dialup POTS, ISDN, ADSL, etc.) and then runs PPP over that connection. In such a configuration, the L2 termination point and PPP session endpoint reside on the same physical device (i.e., the NAS).

L2TP extends the PPP model by allowing the L2 and PPP endpoints to reside on different devices interconnected by a packet-switched network. With L2TP, a user has an L2 connection to an access concentrator (e.g., modem bank, ADSL DSLAM, etc.), and the concentrator then tunnels individual PPP frames to the NAS. This allows the actual processing of PPP packets to be divorced from the termination of the L2 circuit.

One obvious benefit of such a separation is that instead of requiring the L2 connection to terminate at the NAS, the connection may terminate at a (local) circuit concentrator, which then extends the logical PPP session over a shared infrastructure such as a frame relay circuit or the Internet. From the user's perspective, there is no functional difference between having the L2 circuit terminate in an NAS directly and using L2TP. This protocol may also be used to solve the "multilink hunt-group splitting" problem. Multilink PPP, often used to aggregate ISDN B channels, requires that all channels composing a multilink bundle be grouped at a single Network Access Server (NAS). Because L2TP makes a PPP session appear at a location other than the physical point at which the session was physically received, it can be used to make all channels appear at a single NAS, allowing for a multilink operation even when the physical calls are spread across distinct physical NASs.

L2TP utilizes two types of messages, control messages and data messages. Control messages are used in the establishment, maintenance and clearing of tunnels and calls. Data messages are used to encapsulate PPP frames being carried through the tunnel. Control messages utilize a reliable Control Channel within L2TP to guarantee delivery (see section 5.1 for details). Data messages are not retransmitted when packet loss occurs.

Protocol Structure

L2TP Common header:

<table>
<tr><td colspan="12"></td><td>16</td><td>32 bit</td></tr>
<tr><td>T</td><td>L</td><td>X</td><td>X</td><td>S</td><td>X</td><td>0</td><td>P</td><td>X</td><td>X</td><td>X</td><td>X</td><td>VER</td><td>Length</td></tr>
<tr><td colspan="13">Tunnel ID</td><td>Session ID</td></tr>
<tr><td colspan="13">Ns (opt)</td><td>Nr (opt)</td></tr>
<tr><td colspan="13">Offset size (opt)</td><td>Offset pad (opt)</td></tr>
</table>

- T - The T bit indicates the type of message. It is set to 0 for data messages and 1 for control messages.
- L - When set, this indicates that the Length field is present, indicating the total length of the received packet. Must be set for control messages.
- X - The X bits are reserved for future extensions. All reserved bits are set to 0 on outgoing messages and are ignored on incoming messages.
- S - If the S bit is set, both the Nr and Ns fields are present. S must be set for control messages.
- O - When set, this field indicates that the Offset Size field is present in payload messages. This bit is set to 0 for control messages.
- P - If the Priority (P) bit is 1, this data message receives preferential treatment in its local queuing and transmission.
- Ver - The value of the ver bit is always 002. This indicates a version 1 L2TP message.
- Length - Overall length of the message, including header, message type AVP, plus any additional AVP's associated with a given control message type.
- Tunnel ID - Identifies the tunnel to which a control message applies. If an Assigned Tunnel ID has not yet been received from the peer, Tunnel ID must be set to 0. Once an Assigned Tunnel ID is received, all further packets must be sent with Tunnel ID set to the indicated value.
- Call ID - Identifies the user session within a tunnel to which a control message applies. If a control message does not apply to a single user session within the tunnel (for instance, a Stop-Control-Connection-Notification message), Call ID must be set to 0.
- Nr - The sequence number expected in the next control message to be received.
- Ns - The sequence number for this data or control message.
- Offset size & pad - This field specifies the number of bytes past the L2TP header at which the payload data is expected to start. Actual data within the offset padding is undefined. If the offset field is present, the L2TP header ends after the last octet of the offset padding.

Related protocols

PPP, PPTP, L2F, ATM, Frame Relay, UDP

Sponsor Source

L2TP is defined by IETF (http://www.ietf.org) in RFC 2661.

Reference

http://www.javvin.com/protocol/rfc2661.pdf
Layer Two Tunneling Protocol “L2TP”

Protocol Name

PPTP: Point-to-Point Tunneling Protocol

Protocol Description

Point-to-Point-Tunneling Protocol (PPTP) is a networking technology that supports multiprotocol virtual private networks (VPN), enabling remote users to access corporate networks securely across the Microsoft Windows NT® Workstation, Windows® 95, and Windows 98 operating systems and other point-to-point protocol (PPP)-enabled systems to dial into a local Internet service provider to connect securely to their corporate network through the Internet.

PPTP can also be used to tunnel a PPP session over an IP network. In this configuration the PPTP tunnel and the PPP session run between the same two machines with the caller acting as a PNS. PPTP uses a client-server architecture to decouple functions which exist in current Network Access Servers and support Virtual Private Networks. PPTP specifies a call-control and management protocol which allows the server to control access for dial-in circuit switched calls originating from a PSTN or ISDN, or to initiate outbound circuit switched connections.

PPTP is implemented only by the PAC and PNS. No other systems need to be aware of PPTP. Dial networks may be connected to a PAC without being aware of PPTP. Standard PPP client software should continue to operate on tunneled PPP links.

PPTP uses an extended version of GRE to carry user PPP packets. These enhancements allow for low-level congestion and flow control to be provided on the tunnels used to carry user data between PAC and PNS. This mechanism allows for efficient use of the bandwidth available for the tunnels and avoids unnecessary retransmissions and buffer overruns. PPTP does not dictate the particular algorithms to be used for this low level control but it does define the parameters that must be communicated in order to allow such algorithms to work.

Protocol Structure

16	32 bit
Length	PPTP message type
Magic cookie	
Control message type	Reserved 0
Protocol Version	Reserved 1
Framing capability	
Bearing capability	
Maximum channels	Firmware revision
Host name (64 Octets)	
Vendor string (64 Octets)	

- Length - Total length in octets of this PPTP message including the entire PPTP header.
- PPTP message type - The message type. Possible values are: 1 Control message; 2 Management message.
- Magic cookie - The magic cookie is always sent as the constant 0x1A2B3C4D. Its basic purpose is to allow the receiver to ensure that it is properly synchronized with the TCP data stream.
- Control Message Type - Values may be: 1 Start-Control-Connection-Request; 2 Start-Control-Connection-Reply; 3 Stop-Control-Connection-Request; 4 Stop-Control-Connection-Reply; 5 Echo-Request; 6 Echo-Reply.
- Call Management – Values are: 7 Outgoing-Call-Request; 8 Outgoing-Call-Reply; 9 Incoming-Call-Request; 10 Incoming-Call-Reply; 11 Incoming-Call-Connected; 12 Call-Clear-Request; 13 Call-Disconnect-Notify; 14 WAN-Error-Notify.; PPP Session Control - 15 Set-Link-Info.
- Reserved 0 & 1 - Must be set to 0.
- Protocol version – PPTP version number
- Framing Capabilities - Indicating the type of framing that the sender of this message can provide: 1 - Asynchronous Framing supported; 2 - Synchronous Framing supported
- Bearer Capabilities - Indicating the bearer capabilities that the sender of this message can provide: 1 - Analog access supported; 2 - Digital access supported
- Maximum Channels - The total number of individual PPP sessions this PAC can support.
- Firmware Revision - Contains the firmware revision number of the issuing PAC, when issued by the PAC, or the version of the PNS PPTP driver if issued by the PNS.
- Host Name - Containing the DNS name of the issuing PAC or PNS.
- Vendor Name - Containing a vendor specific string describing the type of PAC being used, or the type of PNS software being used if this request is issued by the PNS.

Related protocols

GRE, PPP, L2TP, L2F

Sponsor Source

PPTP is defined by PPTP forum led by Microsoft and circulated among IETF community.

Reference

http://www.javvin.com/protocol/rfc2637.pdf
Point to Point Tunneling Protocol (PPTP)

Secured Routing Protocols

Protocol Name

DiffServ: Differentiated Service Architecture

Protocol Description

Diifrentiated Service (DiffServ) defines an architecture for implementing scalable service differentiation in the Internet. A “Service” defines some significant characteristics of packet transmission in one direction across a set of one or more paths within a network. These characteristics may be specified in quantitative or statistical terms of throughput, delay, jitter, and/or loss, or may otherwise be specified in terms of some relative priority of access to network resources. Service differentiation is desired to accommodate heterogeneous application requirements and user expectations, and to permit differentiated pricing of Internet service.

DiffServ architecture is composed of a number of functional elements implemented in network nodes, including a small set of per-hop forwarding behaviors, packet classification functions, and traffic conditioning functions including metering, marking, shaping, and policing. This architecture achieves scalability by implementing complex classification and conditioning functions only at network boundary nodes, and by applying per-hop behaviors to aggregates of traffic which have been appropriately marked using the DS field in the IPv4 or IPv6 headers [DSFIELD]. Per-hop behaviors are defined to permit a reasonably granular means of allocating buffer and bandwidth resources at each node among competing traffic streams. Per-application flow or per-customer forwarding state need not be maintained within the core of the network.

The differentiated services architecture is based on a simple model where traffic entering a network is classified and possibly conditioned at the boundaries of the network, and assigned to different behavior aggregates. Each behavior aggregate is identified by a single DS codepoint. Within the core of the network, packets are forwarded according to the per-hop behavior associated with the DS codepoint. In this section, we discuss the key components within a differentiated services region, traffic classification and conditioning functions, and how differentiated services are achieved through the combination of traffic conditioning and PHB-based forwarding.

Protocol Structure

In DiffServ, a replacement header field, called the DS field, is defined, which is intended to supersede the existing definitions of the IPv4 TOS octet and the IPv6 Traffic Class octet. The format of the header as follows:

- DSCP - differentiated services codepoint to select the PHB a packet experiences at each node
- CU - currently unused

Related protocols

IP, IPv6

Sponsor Source

DiffServ is defined by IETF (http://www.ietf.org) in RFC 2474 and 2475.

Reference

http://www.javvin.com/protocol/rfc2475.pdf
An Architecture for Differentiated Services
http://www.javvin.com/protocol/rfc2475.pdf
Differentiated Services Field

Protocol Name

GRE: Generic Routing Encapsulation

Protocol Description

Generic Routing Encapsulation (GRE) is a protocol for encapsulation of an arbitrary network layer protocol over another arbitrary network layer protocol.

In the most general case, a system has a packet, which is called payload, which needs to be encapsulated and delivered to some destination. The payload is first encapsulated in a GRE packet. The resulting GRE packet can then be encapsulated in some other protocol and then forwarded. This outer protocol is called the delivery protocol.

When IPv4 is being carried as the GRE payload, the Protocol Type field MUST be set to 0x800. When a tunnel endpoint decapsulates a GRE packet which has an IPv4 packet as the payload, the destination address in the IPv4 payload packet header MUST be used to forward the packet and the TTL of the payload packet MUST be decremented. Care should be taken when forwarding such a packet, since if the destination address of the payload packet is the encapsulator of the packet (i.e., the other end of the tunnel), looping can occur. In this case, the packet MUST be discarded. The IPv4 protocol 47 is used when GRE packets are encapsulated in IPv4.

Security in a network using GRE should be relatively similar to security in a normal IPv4 network, as routing using GRE follows the same routing that IPv4 uses natively. Route filtering will remain unchanged. However packet filtering requires either that a firewall look inside the GRE packet or that the filtering is done at the GRE tunnel endpoints. In those environments in which this is considered to be a security issue it may be desirable to terminate the tunnel at the firewall.

Protocol Structure

In DiffServ, a replacement header field, called the DS field, is defined, which is intended to supersede the existing definitions of the IPv4 TOS octet and the IPv6 Traffic Class octet. The format of the header as follows:

1	13	16	32bit
C	Reserved 0&1	Ver	Protocol type
Checksum (optional)			Reserved

- C – Checksum Present.
- Reserved 0 & 1 – reserved for future use.
- Ver – version number; must be zero.
- Protocol Type - contains the protocol type of the payload packet.
- Checksum - contains the IP checksum sum of the all the 16 bit words in the GRE header and the payload packet.

Related protocols

IPv4

Sponsor Source

GRE is defined by IETF (http://www.ietf.org) in RFC 2784.

Reference

http://www.javvin.com/protocol/rfc2784.pdf
Generic Routing Encapsulation (GRE)

Protocol Name

IPsec: Security Architecture for IP

Protocol Description

Internet Security architecture (IPsec) defines the security services at the IP layer by enabling a system to select required security protocols, determine the algorithm(s) to use for the service(s), and put in place any cryptographic keys required to provide the requested services. IPsec can be used to protect one or more “paths” between a pair of hosts, between a pair of security gateways, or between a security gateway and a host.

The set of security services that IPsec can provide includes access control, connectionless integrity, data origin authentication, rejection of replayed packets (a form of partial sequence integrity), confidentiality (encryption), and limited traffic flow confidentiality. Because these services are provided at the IP layer, they can be used by any higher layer protocol, e.g., TCP, UDP, ICMP, BGP, etc.

These objectives are met through the use of two traffic security protocols, the Authentication Header (AH) and the Encapsulating Security Payload (ESP), and through the use of cryptographic key management procedures and protocols. The set of IPsec protocols employed in any context, and the ways in which they are employed, will be determined by the security and system requirements of users, applications, and/or sites/organizations.

When these mechanisms are correctly implemented and deployed, they ought not to adversely affect users, hosts, and other Internet components that do not employ these security mechanisms for protection of their traffic. These mechanisms also are designed to be algorithm-independent. This modularity permits selection of different sets of algorithms without affecting the other parts of the implementation. For example, different user communities may select different sets of algorithms (creating cliques) if required.

A standard set of default algorithms is specified to facilitate interoperability in the global Internet. The use of these algorithms, in conjunction with IPsec traffic protection and key management protocols, is intended to permit system and application developers to deploy high quality, Internet layer, cryptographic security technology.

Protocol Structure

IPsec Architecture includes many protocols and algorithms. The relationship of these protocols are displayed as follows:

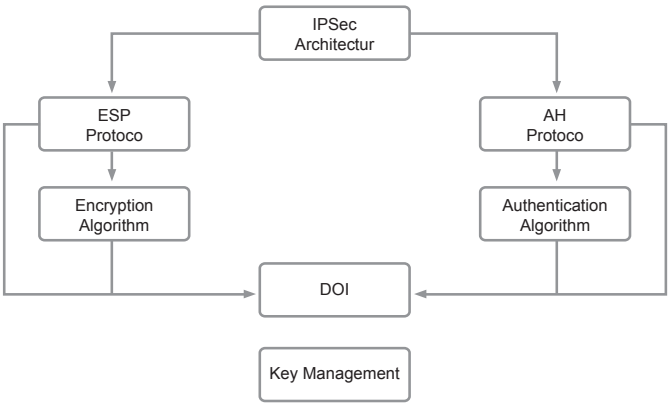


Figure 2-5: IPsec Protocol Stack Structure

The details of each protocol will be presented in separate documents.

Related protocols

ESP, AH, DES, AES, IKE, DOI, HMAC, HMAC-MD5, HMAC-SHA, PKI, IP, IPv6, ICMP

Sponsor Source

IPsec is defined by IETF (http://www.ietf.org).

Reference

http://www.javvin.com/protocol/rfc2401.pdf
Security Architecture for the Internet Protocol
http://www.javvin.com/protocol/rfc2411.pdf
IP Security Document Roadmap

Protocol Name

IPsec AH: IPsec Authentication Header

Protocol Description

IP Authentication Header (AH), a key protocol in the IPsec (Internet Security) architecture, is used to provide connectionless integrity and data origin authentication for IP datagrams, and to provide protection against replays. This latter (optional) service may be selected, by the receiver, when a Security Association is established. AH provides authentication for as much of the IP header as possible, as well as for upper level protocol data. However, some IP header fields may change in transit and the value of these fields, when the packet arrives at the receiver, may not be predictable by the sender. The values of such fields cannot be protected by AH. Thus the protection provided to the IP header by AH is somewhat piecemeal.

AH may be applied alone, in combination with the IP Encapsulating Security Payload (ESP), or in a nested fashion through the use of tunnel mode. Security services can be provided between a pair of communicating hosts, between a pair of communicating security gateways, or between a security gateway and a host. ESP may be used to provide the same security services, and it also provides a confidentiality (encryption) service. The primary difference between the authentication provided by ESP and by AH is the extent of the coverage. Specifically, ESP does not protect any IP header fields unless those fields are encapsulated by ESP. For more details on how to use AH and ESP in various network environments, see the reference documents.

When used with IPv6, the Authentication Header normally appears after the IPv6 Hop-by-Hop Header and before the IPv6 Destination Options. When used with IPv4, the Authentication Header normally follows the main IPv4 header.

Protocol Structure

8	16	32bit
Next Header	Payload Length	Reserved
Security parameters index (SPI)		
Sequence Number Field		
Authentication data (variable)		

- Next header - identifies the type of the next payload after the Authentication Header.
- Payload Length - specifies the length of AH in 32-bit words (4-byte units), minus “2”.
- SPI - an arbitrary 32-bit value that, in combination with the destination IP address and security protocol (AH), uniquely identifies the Security Association for this datagram.
- Sequence Number – contains a monotonically increasing counter value and is mandatory and is always present even if the receiver does not elect to enable the anti-replay service for a specific SA.
- Authentication Data - a variable-length field containing an Integrity Check Value (ICV) computed over the ESP packet minus the Authentication Data.

Related protocols

IPsec, ESP, DES, AES, IKE, DOI, HMAC, HMAC-MD5, HMAC-SHA, PKI, IP, IPv6, ICMP

Sponsor Source

IP AH is defined by IETF (http://www.ietf.org)in RFC 2402.

Reference

http://www.javvin.com/protocol/rfc2402.pdf
IP Authentication Header

Protocol Name

IPsec ESP: IPsec Encapsulating Security Payload

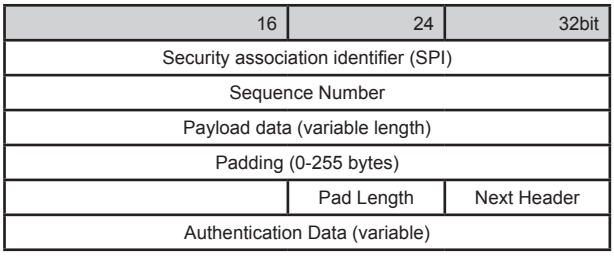
Protocol Description

Encapsulating Security Payload (ESP), a key protocol in the IPsec (Internet Security) architecture, is designed to provide a mix of security services in IPv4 and IPv6. The IP Encapsulating Security Payload (ESP) seeks to provide confidentiality and integrity by encrypting data to be protected and placing the encrypted data in the data portion of the IP ESP. Depending on the user's security requirements, this mechanism may be used to encrypt either a transport-layer segment (e.g., TCP, UDP, ICMP, IGMP) or an entire IP datagram. Encapsulating the protected data is necessary to provide confidentiality for the entire original datagram.

The ESP header is inserted after the IP header and before the upper layer protocol header (transport mode) or before an encapsulated IP header (tunnel mode). The Internet Assigned Numbers Authority has assigned Protocol Number 50 to ESP. The header immediately preceding an ESP header will always contain the value 50 in its Next Header (IPv6) or Protocol (IPv4) field. ESP consists of an unencrypted header followed by encrypted data. The encrypted data includes both the protected ESP header fields and the protected user data, which is either an entire IP datagram or an upper-layer protocol frame (e.g., TCP or UDP).

ESP is used to provide confidentiality, data origin authentication, connectionless integrity, an anti-replay service, and limited traffic flow confidentiality. The set of services provided depends on options selected at the time of Security Association establishment and on the placement of the implementation. Confidentiality may be selected independent of all other services. However, use of confidentiality without integrity/authentication (either in ESP or separately in AH) may subject traffic to certain forms of active attacks that could undermine the confidentiality service. Data origin authentication and connectionless integrity are joint services and are offered as an option in conjunction with (optional) confidentiality. The anti-replay service may be selected only if data origin authentication is selected, and its election is solely at the discretion of the receiver.

Protocol Structure



- Security association identifier - a pseudo-random value identifying the security association for this datagram.
- Sequence Number – contains a monotonically increasing counter value and is mandatory and is always present even if the receiver does not elect to enable the anti-replay service for a specific SA.
- Payload Data - a variable-length field containing data described by the Next Header field.
- Padding – padding for encryption.
- Pad length - indicates the number of pad bytes immediately preceding it.
- Next header - identifies the type of data contained in the Payload Data field, e.g., an extension header in IPv6 or an upper layer protocol identifier.
- Authentication Data - a variable-length field containing an Integrity Check Value (ICV) computed over the ESP packet minus the Authentication Data.

Related protocols

IPsec, AH, DES, AES, IKE, DOI, HMAC, HMAC-MD5, HMAC-SHA, PKI, IP, IPv6, ICMP

Sponsor Source

ESP is defined by IETF (http://www.ietf.org) in RFC 2406.

Reference

http://www.javvin.com/protocol/rfc2406.pdf
IP Encapsulating Security Payload (ESP)

Protocol Name

IPsec IKE: Internet Key Exchange Protocol

Protocol Description

Internet Key Exchange (IKE) Protocol, a key protocol in the IPsec architecture, is a hybrid protocol using part of Oakley and part of SKEME in conjunction with ISAKMP to obtain authenticated keying material for use with ISAKMP, and for other security associations such as AH and ESP for the IPsec DOI.

ISAKMP provides a framework for authentication and key exchange but does not define them. ISAKMP is designed to be key exchange independent and supports many different key exchanges. The Internet Key Exchange (IKE) is one of a series of key exchanges—called “modes”.

IKE processes can be used for negotiating virtual private networks (VPNs) and also for providing a remote user from a remote site (whose IP address need not be known beforehand) access to a secure host or network. Client negotiation is supported. Client mode is where the negotiating parties are not the endpoints for which security association negotiation is taking place. When used in client mode, the identities of the end parties remain hidden.

IKE implementations support the following attribute values:

- DES in CBC mode with a weak, and semi-weak, key check
- MD5 and SHA.
- Authentication via pre-shared keys.
- MODP over default group number one.

In addition, IKE implementations support: 3DES for encryption; Tiger for hash; the Digital Signature Standard, RSA signatures and authentication with RSA public key encryption; and MODP group number 2. IKE implementations MAY support any additional encryption algorithms and MAY support ECP and EC2N groups.

The IKE modes must be implemented whenever the IPsec DOI is implemented. Other DOIs MAY use the modes described here.

Protocol Structure

IKE messages are a combination of ISAKMP header and SKEME and Oakley fields. The specific message format depends on the message phases and modes. For more details, see the reference documents.

Related protocols

IPsec, ESP, AH, DES, AES, DOI, HMAC, HMAC-MD5, HMAC-SHA, PKI, IP, IPv6, ICMP

Sponsor Source

IP IKE is defined by IETF (http://www.ietf.org) in RFC 2409.

Reference

http://www.javvin.com/protocol/rfc2409.pdf
The Internet Key Exchange (IKE)

Protocol Name

IPsec ISAKMP: Internet Security Association and Key Management Protocol

Protocol Description

ISAKMP, a key protocol in the IPsec (Internet Security) architecture, combines the security concepts of authentication, key management, and security associations to establish the required security for government, commercial, and private communications on the Internet.

The Internet Security Association and Key Management Protocol (ISAKMP) defines procedures and packet formats to establish, negotiate, modify and delete Security Associations (SAs). SAs contain all the information required for execution of various network security services, such as the IP layer services (such as header authentication and payload encapsulation), transport or application layer services, or self-protection of negotiation traffic. ISAKMP defines payloads for exchanging key generation and authentication data. These formats provide a consistent framework for transferring key and authentication data independent of the key generation technique, encryption algorithm and authentication mechanism.

ISAKMP is distinct from key exchange protocols in order to cleanly separate the details of security association management (and key management) from the details of key exchange. There may be many different key exchange protocols, each with different security properties. However, a common framework is required for agreeing to the format of SA attributes, and for negotiating, modifying, and deleting SAs. ISAKMP serves as this common framework.

Separating the functionality into three parts adds complexity to the security analysis of a complete ISAKMP implementation. However, the separation is critical for interoperability between systems with differing security requirements, and should also simplify the analysis of further evolution of an ISAKMP server.

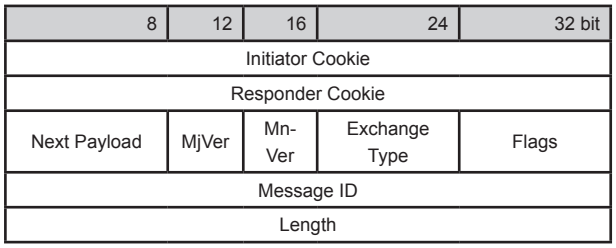
ISAKMP is intended to support the negotiation of SAs for security protocols at all layers of the network stack (e.g., IPSEC, TLS, TLSP, OSPF, etc.). By centralizing the management of the security associations, ISAKMP reduces the amount of duplicated functionality within each security protocol. ISAKMP can also reduce connection setup time, by negotiating a whole stack of services at once.

Within ISAKMP, a Domain of Interpretation (DOI) is used to group related protocols using ISAKMP to negotiate security associations. Security protocols sharing a DOI choose security protocol and cryptographic transforms from a common namespace and share key exchange protocol identifiers. They also share a common interpretation of DOI-specific payload data content, including the Security Association and Identification payloads.

Overall, ISAKMP places requirements on a DOI definition to define the following:

- Naming scheme for DOI-specific protocol identifiers
- Interpretation for the Situation field
- Set of applicable security policies
- Syntax for DOI-specific SA Attributes (Phase II)
- Syntax for DOI-specific payload contents
- Additional Key Exchange types, if needed
- Additional Notification Message types, if needed

Protocol Structure



- Initiator Cookie - The Initiator Cookie: Cookie of the entity that initiated SA establishment, SA notification, or SA deletion
- Responder Cookie - The Responder Cookie: Cookie of the entity that is responding to an SA establishment request, SA notification, or SA deletion.
- Next Payload - The type of the next payload in the message.
- Major Version - The major version of the ISAKMP protocol in use.
- Minor Version - The minor version of the ISAKMP protocol in use.
- Exchange Type - The type of exchange being used
- Flags - Various options that are set for the ISAKMP exchange.
- Message ID - A Unique Message Identifier used to identify protocol state during Phase 2 negotiations.
- Length - Length of total message (header + payloads) in octets.

Related protocols

IPsec, ESP, AH, DES, AES, IKE, DOI, HMAC, HMAC-MD5, HMAC-SHA, PKI, IP, IPv6, ICMP

Sponsor Source

ISAKMP is defined by IETF (http://www.ietf.org) in RFC 2408.

Reference

http://www.javvin.com/protocol/rfc2408.pdf
Internet Security Association and Key Management Protocol (ISAKMP)

Protocol Name

TLS: Transport Layer Security Protocol

Protocol Description

Transport Layer Security (TLS) Protocol is to provide privacy and data integrity between two communicating applications. The protocol is composed of two layers: the TLS Record Protocol and the TLS Handshake Protocol. At the lowest level, layered on top of some reliable transport protocol (TCP) is the TLS Record Protocol. The TLS Record Protocol provides connection security that has two basic properties:

- Private - Symmetric cryptography is used for data encryption (DES, RC4, etc.) The keys for this symmetric encryption are generated uniquely for each connection and are based on a secret negotiated by another protocol (such as the TLS Handshake Protocol). The Record Protocol can also be used without encryption.
- Reliable - Message transport includes a message integrity check using a keyed MAC. Secure hash functions (SHA, MD5, etc.) are used for MAC computations. The Record Protocol can operate without a MAC, but is generally only used in this mode while another protocol is using the Record Protocol as a transport for negotiating security parameters.

The TLS Record Protocol is used for encapsulation of various higher level protocols. One such encapsulated protocol, the TLS Handshake Protocol, allows the server and client to authenticate each other and to negotiate an encryption algorithm and cryptographic keys before the application protocol transmits or receives its first byte of data. The TLS Handshake Protocol provides connection security that has three basic properties:

1. The peer's identity can be authenticated using asymmetric, or public key, cryptography (RSA, DSS, etc.). This authentication can be made optional, but is generally required for at least one of the peers.
2. The negotiation of a shared secret is secure: The negotiated secret is unavailable to eavesdroppers, and for any authenticated connection the secret cannot be obtained, even by an attacker who can place himself in the middle of the connection.
3. The negotiation is reliable: no attacker can modify the negotiation communication without being detected by the parties to the communication.

TLS is based on the Secure Socket Layer (SSL), a protocol originally created by Netscape. One advantage of TLS is that it is application protocol independent. The TLS protocol runs above TCP/IP and below application protocols such as HTTP or IMAP. The HTTP running on top of TLS or SSL is often called HTTPS.

The TLS standard does not specify how protocols add security with TLS; the decisions on how to initiate TLS handshaking and how to interpret the authentication certificates exchanged are left up to the judgment of the designers and implementers of protocols which run on top of TLS.

Protocol Structure

TLS protocol includes two protocol groups: TLS Record Protocol and TLS Handshake Protocols, which have many messages with different formats. We only summarize the protocols here without details, which can be found in the reference documents.

TLS Record Protocol is a layered protocol. At each layer, messages may include fields for length, description, and content. The Record Protocol takes messages to be transmitted, fragments the data into manageable blocks, optionally compresses the data, applies a MAC, encrypts, and transmits the result. Received data is decrypted, verified, decompressed, and reassembled, then delivered to higher level clients.

TLS connection state is the operating environment of the TLS Record Protocol. It specifies a compression algorithm, encryption algorithm,and MAC algorithm.

TLS Record Layer receives uninterrupted data from higher layers in non-empty blocks of arbitrary size. Key calculation: The Record Protocol requires an algorithm to generate keys, IVs, and MAC secrets from the security parameters provided by the handshake protocol.

TLS Handshake Protocol: consists of a suite of three sub-protocols which are used to allow peers to agree upon security parameters for the record layer, authenticate themselves, instantiate negotiated security parameters, and report error conditions to each other.

Change cipher spec protocol
Alert protocol
Handshake protocol

Related protocols

GRE, PPP, L2TP, PPTP, RSA

Sponsor Source

TLS is defined by IETF (http://www.ietf.org) in RFC 2246 and updated in RFC 3546.

Reference

http://www.javvin.com/protocol/rfc2246.pdf
The TLS Protocol Version 1.0.

Other Security Protocols

Protocol Name

SOCKS v5: Protocol for sessions traversal across firewall securely

Protocol Description

The SOCKS protocol provides a framework for client-server applications in both the TCP and UDP domains to conveniently and securely use the services of a network firewall. The protocol is conceptually a “shim-layer” between the application layer and the transport layer, and as such does not provide network layer gateway services, such as forwarding of ICMP messages.

The use of network firewalls, systems that effectively isolate an organizations internal network structure from an exterior network, such as the Internet is becoming increasingly popular. These firewall systems typically act as application-layer gateways between networks, usually offering controlled TELNET, FTP, and SMTP access. SOCKS provides a general framework for these protocols to transparently and securely traverse a firewall.

SOCKS version 5, also, provides strong authentication of such traversal, while SOCKS Version 4 provides only unsecured firewall traversal for TCP-based client-server applications, including TELNET, FTP, and protocols such as HTTP, WAIS and GOPHER. SOCKS version 5 extends the SOCKS Version 4 model to include UDP, and extends the framework to include provisions for generalized strong authentication schemes. It also adapts the addressing scheme to encompass domain-name and IPv6 addresses.

The implementation of the SOCKS protocol typically involves the recompilation or relinking of TCP-based client applications to use the appropriate encapsulation routines in the SOCKS library.

Protocol Structure

SOCKS v5 has a few messages with different formats.

Version identifier/method selection message:

1 byte	1 byte	1-225 bytes
Version	NMethods	Methods

The SOCKS request message:

1 byte	1 byte	Value of 0	1 byte	Variable	2 bytes
Version	CMD	Rsv	ATYP	DST addr	DST Port

The method selection message:

1 byte	1 byte
Version	Method

The reply message:

1 byte	1 byte	Value of 0	1 byte	Variable	2 bytes
Version	REP	RSV	ATYP	BND addr	BND Port

UDP request header:

2 bytes	1 byte	1 byte	Variable	2 bytes	Variable
RSV	FRAG	ATYP	DST Addr	DST Port	Data

Related protocols

TCP, UDP, ICMP, HTTP, Gopher, TELNET, FTP

Sponsor Source

SOCKS is defined by IETF (http://www.ietf.org) in RFC 1928.

Reference

http://www.javvin.com/protocol/rfc1928.pdf
SOCKS Protocol Version 5

Voice over IP and VOIP Protocols

Description

Voice over IP (VOIP) uses the Internet Protocol (IP) to transmit voice as packets over an IP network. Using VOIP protocols, voice communications can be achieved on any IP network regardless whether it is Internet, Intranet or Local Area Networks (LAN). In a VOIP enabled network, the voice signal is digitized, compressed and converted to IP packets and then transmitted over the IP network. VOIP signaling protocols are used to set up and tear down calls, carry information required to locate users and negotiate capabilities. The key benefits of Internet telephony (Voice over IP) are the very low cost, the integration of data, voice and video on one network, the new services created on the converged network and simplified management of end user and terminals.

There are a few VOIP protocol stacks which are derived by various standard bodies and vendors, namely H.323, SIP, MEGACO and MGCP.

H.323 is the ITU-T's standard, which was originally developed for multimedia conferencing on LANs, but was later extended to cover Voice over IP. The standard encompasses both point to point communications and multipoint conferences. H.323 defines four logical components: Terminals, Gateways, Gatekeepers and Multipoint Control Units (MCUs). Terminals, gateways and MCUs are known as endpoints.

Session Initiation Protocol (SIP) is the IETF's standard for establishing VOIP connections. SIP is an application layer control protocol for creating, modifying and terminating sessions with one or more participants. The architecture of SIP is similar to that of HTTP (client-server protocol). Requests are generated by the client and sent to the server. The server processes the requests and then sends a response to the client. A request and the responses for that request make a transaction.

Media Gateway Control Protocol (MGCP) is a Cisco and Telcordia proposed VOIP protocol that defines communication between call control elements (Call Agents or Media Gateway) and telephony gateways. MGCP is a control protocol, allowing a central coordinator to monitor events in IP phones and gateways and instructs them to send media to specific addresses. In the MGCP architecture, The call control intelligence is located outside the gateways and is handled by the call control elements (the Call Agent). Also the call control elements (Call Agents) will synchronize with each other to send coherent commands to the gateways under their control.

The Media Gateway Control Protocol (Megaco) is a result of joint efforts of the IETF and the ITU-T (ITU-T Recommendation H.248). Megaco/H.248 is a protocol for the control of elements in a physically decomposed multimedia gateway, which enables separation of call control from media conversion. Megaco/H.248 addresses the relationship between the Media Gateway (MG), which converts circuit-switched voice to packet-based traffic, and the Media Gateway Controller, which dictates the service logic of that traffic. Megaco/H.248 instructs an MG to connect streams coming from outside a

packet or cell data network onto a packet or cell stream such as the Real-Time Transport Protocol (RTP). Megaco/H.248 is essentially quite similar to MGCP from an architectural standpoint and the controller-to-gateway relationship, but Megaco/H.248 supports a broader range of networks, such as ATM.

In the past few years, the VOIP industry has been working on addressing the following key issues

Quality of voice - As IP was designed for carrying data, it does not provide real time guarantees but only provides best effort service. For voice communications over IP to become acceptable to users, the packet delay and getter needs to be less than a threshold value.

Interoperability - In a public network environment, products from different vendors need to operate with each other for Voice over IP to become common among users.

Security - Encryption (such as SSL) and tunneling (L2TP) technologies have been developed to protect VOIP signaling and bear traffic.

Integration with Public Switched Telephone Network(PSTN) - While Internet telephony is being introduced, it will need to work in conjunction with PSTN in the foreseeable future. Gateway technologies are being developed to bridge the two networks.

Scalability - VOIP systems need to be flexible enough to grow to the large user market for both private and public services. Many network management and user management technologies and products are being developed to address the issue.

Key VOIP Protocols

The key protocols for AAA and VPN:

Signaling	
ITU-T H.323	H.323: Packet-based multimedia communications (VoIP) architecture
	H.225: Call Signaling and RAS in H.323 VOIP Architecture
	H.235: Security for H.323 based systems and communications
	H.245: Control Protocol for Multimedia Communication
	T.120: Multipoint Data Conferencing Protocol Suite
IETF	Megaco / H.248: Media Gateway Control protocol
	MGCP: Media Gateway Control Protocol
	RTSP: Real Time Streaming Protocol
	SIP: Session Initiation Protocol
	SDP: Session Description Protocol
	SAP: Session Announcement Protocol
Cisco Skinny	SCCP: Skinny Client Control Protocol
Media/CODEC	G.7xx: Audio (Voice) Compression Protocols (G.711, G.721, G.722, G.723, G.726, G.727. G.728, G.729)
	H.261: Video Coding and Decoding (CODEC)
	H.263: Video Coding and Decoding (CODEC)
	RTP: Real Time Transport Protocol
	RTCP: RTP Control Protocol
Others	COPS: Common Open Policy Service
	SCTP: Stream Control Transmission Protocol
	TRIP: Telephony Routing Over IP

Sponsor Source

VOIP protocols are defined by IETF, ITU-T and some vendors.

Reference

http://www.cis.ohio-state.edu/~jain/refs/ref_voip.htm
Voice Over IP and IP Telephony References

Signalling

Protocol Name

H.323: VOIP Protocols

Protocol Description

H.323, a protocol suite defined by ITU-T, is for voice transmission over internet (Voice over IP or VOIP). In addition to voice applications, H.323 provides mechanisms for video communication and data collaboration, in combination with the ITU-T T.120 series standards. H.323 is one of the major VOIP standards, on a par with Megaco and SIP.

H.323 is an umbrella specification, because it includes various other ITU standards. The components under H.323 architecture are terminal, gateway, gatekeeper and multipoint control units (MCUs).

Terminal represents the end device of every connection. It provides real time two way communications with another H.323 terminal, GW or MSU. This communication consists of speech, speech and data, speech, and video, or a combination of speech, data and video.

Gateways establish the connection between the terminals in the H.323 network and the terminals belonging to networks with different protocol stacks such as the traditional PSTN network or SIP or Megaco end points.

Gatekeepers are responsible for translating between telephone number and IP addresses. They also manage the bandwidth and provide a mechanism for terminal registration and authentication. Gatekeepers also provide services such as call transfer, call forwarding etc.

MCUs take care of establishing multipoint conferences. An MCU consists of a mandatory Multipoint Control, which is for call signaling and conference control, and an optional Multipoint Processor, which is for switching/mixing of the media stream and sometimes real-time transcoding of the received audio/video streams.

There are five types of information exchange enabled in the H.323 architecture:

Audio (digitized) voice
Video (digitized)
Data (files or image)
Communication control (exchange of supported functions, controlling logic channels, etc.)
Controlling connections and sessions (setup and tear down)

The H.323 was first published in 1996 and the latest version (v5) was completed in 2003.

Protocol Structure

The protocols in the H.323 protocol suite are:

Call control and signaling
H.225.0: Call signaling protocols and media stream packetization (uses a subset of Q.931 signaling protocol)
H.225.0/RAS: Registration, Admission and Status
H.245: Control protocol for multimedia communication

Audio processing:
G.711: Pulse code modulation of voice frequencies
G.722: 7 kHz audio coding within 64 kb/s
G.723.1: Dual rate speech coders for multimedia communication transmitting at 5.3 and 6.3 kb/s
G.728: Coding of speech at 16 kb/s using low-delay code excited linear prediction
G.729: Coding of speech at 8kb/s using conjugate-structure algebraic-code-excited linear-prediction

Video processing:
H.261: Video codecs for audiovisual services at Px64kps.
H.263: Video coding for low bit rate communication.

Data conferencing:
T.120: This is a protocol suite for data transmission between end points. It can be used for various applications in the field of Collaboration Work, such as white-boarding, application sharing, and joint document management. T.120 utilizes layer architecture similar to the OSI model. The top layers (T.126, T.127) are based on the services of lower layers (T.121, T.125).

Media transportation:
RTP: Real time Transport Protocol
RTCP: RTP Control Protocol
Security:
H.235: Security and encryption for H.series multimedia terminals.

Supplementary services:
H.450.1: Generic functions for the control of supplementary services in H.323
H.450.2: Call transfer
H.450.3: Call diversion
H.450.4: Call hold
H.450.5: Call park and pick up
H.450.6: Call waiting
H.450.7: Message waiting indication
H.450.8: Names Identification services
H.450.9: Call completion services for H.323 networks

The following figure illustrates the structure of the key protocol in the H.323 architecture. Details of each protocols will be discussed in separate documents.

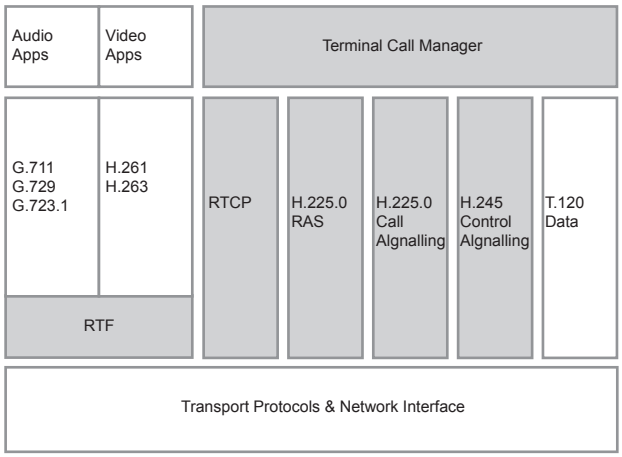


Figure 2-6: H.323 Protocol Stack Structure

Related protocols

RTP, RTSP, SIP, Megaco, H.248, Q.931, H.225

Sponsor Source

H.323 is a ITU-T (http://www.itu.int/ITU-T/) standard.

Reference

http://www.h323forum.org/papers/
H.323 papers and documents

Protocol Name

H.225.0: Call signalling protocols and media stream packetization for packet-based multimedia communication systems

Protocol Description

H.225.0, a key protocol in the H.323 VOIP architecture defined by ITU-T, is a standard to cover narrow-band visual telephone services defined in H.200/AV.120-Series Recommendations. It specifically deals with those situations where the transmission path includes one or more packet based networks, each of which is configured and managed to provide a non-guaranteed QoS, which is not equivalent to that of N-ISDN, such that additional protection or recovery mechanisms beyond those mandated by Rec. H.320 are necessary in the terminals. H.225.0 describes how audio, video, data and control information on a packet based network can be managed to provide conversational services in H.323 equipment. H.225 has two major parts: Call signaling and RAS (Registration, Admission and Status).

H.225 call control signaling is used to setup connections between H.323 endpoints. This is achieved by exchanging H.225 protocol messages on the call-signaling channel. The call-signaling channel is opened between two H.323 endpoints or between an endpoint and the gatekeeper. The ITU H.225 recommendation specifies the use and support of Q.931 signaling messages. A reliable (TCP) call control channel is created across an IP network on TCP port 1720. This port initiates the Q.931 call control messages for the purpose of connecting, maintaining, and disconnecting calls. When a gateway is present in the network zone, H.225 call setup messages are exchanged either via Direct Call Signaling or Gatekeeper-Routed Call Signaling (GKRCS). The gatekeeper decides the method chosen during the RAS admission message exchange. If no gatekeeper is present, H.225 messages are exchanged directly between the endpoints.

H.225/RAS (Registration, Admission and Status) is the protocol between endpoints (terminals and gateways) and gatekeepers. The RAS is used to perform registration, admission control, bandwidth changes, status, and disengage procedures between endpoints and gatekeepers. An RAS channel is used to exchange RAS messages. This signaling channel is opened between an endpoint and a gatekeeper prior to the establishment of any other channel.

Protocol Structure

1	2	3	4	8bit
Protocol Discriminator				
0	0	0	0	Length of call reference bits
Call reference value				
0	Message type			
Information Elements				

- Protocol discriminator - Distinguishes messages for user-network call control from other messages.
- Length of call ref - The length of the call reference value.
- Call reference value - Identifies the call or facility registration/cancellation request at the local user-network interface to which the particular message applies. May be up to 2 octets in length.
- Message type - Identifies the function of the message sent.
- Information elements - Two categories of information elements are defined: single octet information elements and variable length information elements, as shown in the following illustrations.

1	4	8bit
1	IEI	Contents of IE

1	8bit
1	IE Identifier

1	8bit
1	IEI
Length of contents of IE	
Contents of IE (variable)	

Key RAS messages:

Message	Function
RegistrationRequest (RRQ)	Request from a terminal or gateway to register with a gatekeeper. Gatekeeper either confirms or rejects (RCF or RRJ).
AdmissionRequest (ARQ)	Request for access to packet network from terminal to gatekeeper. Gatekeeper either confirms or rejects (ACF or ARJ).
BandwidthRequest (BRQ)	Request for changed bandwidth allocation, from terminal to gatekeeper. Gatekeeper either confirms or rejects (BCF or BRJ).
DisengageRequest (DRQ)	If sent from endpoint to gatekeeper, DRQ informs gatekeeper that endpoint is being dropped; if sent from gatekeeper to endpoint, DRQ forces call to be dropped. Gatekeeper either confirms or rejects (DCF or DRJ). If DRQ sent by gatekeeper, endpoint must reply with DCF.
InfoRequest (IRQ)	Request for status information from gatekeeper to terminal.

InfoRequestResponse (IRR)	Response to IRQ. May be sent unsolicited by terminal to gatekeeper at predetermined intervals.
RAS timers and Request in Progress (RIP)	Recommended default timeout values for response to RAS messages and subsequent retry counts if response is not received.

Related protocols

RTP, RTSP, SIP, Megaco, H.248, Q.931, H.323, H.245

Sponsor Source

H.225 is an ITU-T (http://www.itu.int/ITU-T/) standard.

Reference

http://www.javvin.com/protocol/H225v5.pdf
Call signalling protocols and media stream packetization for packet-based multimedia communication systems" Version 5.
http://www.h323forum.org/papers/
H.323 papers and documents

Protocol Name

H.235: Security and encryption for H-series (H.323 and other H.245-based) multimedia terminals

Protocol Description

H.235 is the security recommendation for the H.3xx series systems. In particular, H.235 provides security procedures for H.323-, H.225.0-, H.245- and H.460-based systems. H.235 is applicable to both simple point-to-point and multipoint conferences for any terminals which utilize H.245 as a control protocol.

The scope of H.235 is to provide authentication, privacy and integrity for H.xxx based systems. H.235 provides a means for a person, rather than a device, to be identified. The security profiles include: 1) a simple, password-based security profile; 2) a profile using digital certificates and dependent on a fully-deployed public-key infrastructure; and 3) combines features of both 1) and 2). Use of these security profiles is optional.

H.235 includes the ability to negotiate services and functionality in a generic manner, and to be selective concerning cryptographic techniques and capabilities utilized. The specific manner in which these are used relates to systems capabilities, application requirements and specific security policy constraints. H.235 supports varied cryptographic algorithms, with varied options appropriate for different purposes; e.g. key lengths. Certain cryptographic algorithms may be allocated to specific security services.

H.235 supports signalling of well-known algorithms in addition to signalling non standardized or proprietary cryptographic algorithms. There are no specifically mandated algorithms; however, it is strongly suggested in H.235 that endpoints support as many of the applicable algorithms as possible in order to achieve interoperability. This parallels the concept that the support of H.245 does not guarantee the interoperability between two entities' codecs.

Protocol Structure

H.235 recommends many messages, procedures, structures and algorithms for the security concerns of signaling, control and media communications under H.323 architecture. Here is a summary of the definitions:

1) The call signalling channel may be secured using TLS or IPSEC on a secure well-known port (H.225.0).
2) Users may be authenticated either during the initial call connection, in the process of securing the H.245 channel and/or by exchanging certificates on the H.245 channel.
3) The encryption capabilities of a media channel are determined by extensions to the existing capability negotiation mechanism.
4) Initial distribution of key material from the master is via H.245 OpenLogicalChannel or OpenLogicalChannelAck messages.
5) Re-keying may be accomplished by H.245 commands: EncryptionUpdateCommand, EncryptionUpdateRequest, EncryptionUpdate and EncryptionUpdateAck.
6) Key material distribution is protected either by operating the H.245 channel as a private channel or by specifically protecting the key material using the selected exchanged certificates.
7) The security protocols presented conform either to ISO published standards or to IETF proposed standards.

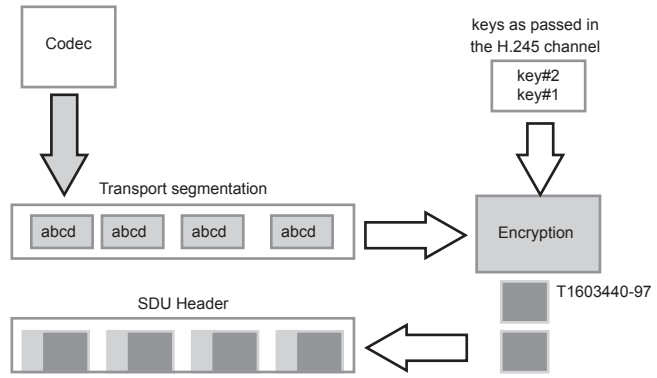


Figure 2-7: H.235 – Encryption of media

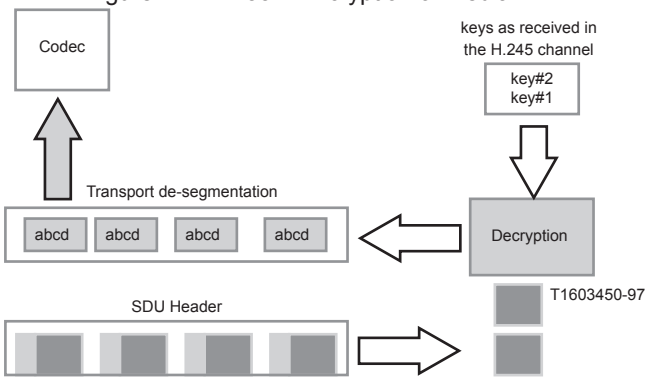


Figure 2-8: H.235 – Decryption of media

The following is a sample flow chart in the H.235 recommendations of encryption for media security.

Related protocols

RTP, RTSP, H.225, Q.931, H.323, H.245

Sponsor Source

H.235 is an ITU-T (http://www.itu.int/ITU-T/) standard.

Reference

http://www.javvin.com/protocol/H235v3.pdf
Security and encryption for H-series (H.323 and other H.245-based) multimedia terminals
http://www.h323forum.org/papers/
H.323 papers and documents

Protocol Name

H.245: Control Protocol for Multimedia Communication

Protocol Description

H.245, a control signaling protocol in the H.323 multimedia communication architecture, is for of the exchange of end-to-end H.245 messages between communicating H.323 endpoints/ terminals. The H.245 control messages are carried over H.245 control channels. The H.245 control channel is the logical channel 0 and is permanently open, unlike the media channels. The messages carried include messages to exchange capabilities of terminals and to open and close logical channels.

After a connection has been set up via the call signaling procedure, the H.245 call control protocol is used to resolve the call media type and establish the media flow, before the call can be established. It also manages the call after it has been established. The steps involved are:

- Master-slave determination process. This is used to determine the master of the call and is useful for avoiding conflicts during call control operations.
- Capability exchange procedure. Each endpoint notifies the other what kind of information it is capable of receiving and transmitting through the receive and transmit capabilities.
- Logical channel procedures. Used for opening and closing logical channels, which are multiplexed paths between the endpoints used for data transfer.
- Request mode command. Using this command, at any point during the conference, the receiving endpoint can request a change in mode of the transmitted information provided the mode is in the transmit capability of the transmitter.
- Control flow command. This can be used by the receiver to fix an upper limit for the transmitter bit rate on any logical channel.
- Communication mode messages. Used by the multipoint controller for selecting a common mode of operation in a multipoint conference.
- Conference request and response messages. Used for controlling a multipoint conference, e.g. password requests, conference chair control.
- Round trip delay commands. Used for measuring the round-trip delay between two endpoints on the control channel.
- Video fast update command. Used for requesting updates for video frames, in case of data loss.
- End session command. After this command the endpoints close all logical channels, drop the call and inform the gatekeeper about the end of the call.

Protocol Structure

H.245 messages are in ASN.1 syntax. MultimediaSystemControlMessage types can be defined as request, response, command and indication messages. Key H.245 messages are as follows:

Message	Function
Master-Slave Determination	Determines which terminal is the master and which is the slave. Possible replies: Acknowledge, Reject, Release (in case of a time out).
Terminal Capability Set	Contains information about a terminal's capability to transmit and receive multimedia streams. Possible replies: Acknowledge, Reject, Release.
Open Logical Channel	Opens a logical channel for transport of audiovisual and data information. Possible replies: Acknowledge, Reject, Confirm.
Close Logical Channel	Closes a logical channel between two endpoints. Possible replies: Acknowledge
Request Mode	Used by a receive terminal to request particular modes of transmission from a transmit terminal. General mode types include VideoMode, AudioMode, DataMode and Encryption Mode. Possible replies: Acknowledge, Reject, Release.
Send Terminal Capability Set	Commands the far-end terminal to indicate its transmit and receive capabilities by sending one or more Terminal Capability Sets.
End Session Command	Indicates the end of the H.245 session. After transmission, the terminal will not send any more H.245 messages.

Related protocols

RTP, RTSP, SIP, Megaco, H.248, Q.931, H.323, H.225, H.235

Sponsor Source

H.245 is an ITU-T (http://www.itu.int/ITU-T/) standard.

Reference

http://www.javvin.com/protocol/H245v9.pdf
Control Protocol for Multimedia Communication (version 9)
http://www.h323forum.org/papers/
H.323 papers and documents

Protocol Name

Megaco/H.248: Media Gateway Control Protocol

Protocol Description

Megaco/H.248, the Media Gateway Control Protocol, is for the control of elements in a physically decomposed multimedia gateway, enabling the separation of call control from media conversion. The Media Gateway Control Protocol (Megaco) is a result of joint efforts of the IETF and the ITU-T Study Group 16. Therefore, the IETF defined Megaco is the same as ITU-T Recommendation H.248.

Megaco/H.248 addresses the relationship between the Media Gateway (MG), which converts circuit-switched voice to packet-based traffic, and the Media Gateway Controller (MGC, sometimes called a call agent or softswitch, which dictates the service logic of that traffic). Megaco/H.248 instructs an MG to connect streams coming from outside a packet or cell data network onto a packet or cell stream such as the Real-Time Transport Protocol (RTP). Megaco/H.248 is essentially quite similar to MGCP from an architectural standpoint and the controller-to-gateway relationship, but Megaco/H.248 supports a broader range of networks, such as ATM.

There are two basic components in Megaco/H.248: terminations and contexts. Terminations represent streams entering or leaving the MG (for example, analog telephone lines, RTP streams, or MP3 streams). Terminations have properties, such as the maximum size of a jitter buffer, which can be inspected and modified by the MGC.

Terminations may be placed into contexts, which are defined as occuring when two or more termination streams are mixed and connected together. The normal, “active” context might have a physical termination (say, one DS0 in a DS3) and one ephemeral one (the RTP stream connecting the gateway to the network). Contexts are created and released by the MG under command of the MGC. A context is created by adding the first termination, and is released by removing (subtracting) the last termination.

A termination may have more than one stream, and therefore a context may be a multistream context. Audio, video, and data streams may exist in a context among several terminations.

Protocol Structure

All Megaco/H.248 messages are in the format of ASN.1 text messages. Megaco/H.248 uses a series of commands to manipulate terminations, contexts, events, and signals. The following is a list of the commands:

1. Add. - The Add command adds a termination to a context. The Add command on the first Termination in a Context is used to create a Context.
2. Modify - The Modify command modifies the properties, events and signals of a termination.
3. Subtract - The Subtract command disconnects a Termination from its Context and returns statistics on the Termination’s participation in the Context. The Subtract command on the last Termination in a Context deletes the Context.
4. Move - The Move command atomically moves a Termination to another context.
5. AuditValue - The AuditValue command returns the current state of properties, events, signals and statistics of Terminations.
6. AuditCapabilities - The AuditCapabilities command returns all the possible values for Termination properties, events and signals allowed by the Media Gateway.
7. Notify - The Notify command allows the Media Gateway to inform the Media Gateway Controller of the occurrence of events in the Media Gateway.
8. ServiceChange - The ServiceChange Command allows the Media Gateway to notify the Media Gateway Controller that a Termination or group of Terminations is about to be taken out of service or has just been returned to service. ServiceChange is also used by the MG to announce its availability to an MGC (registration), and to notify the MGC of impending or completed restart of the MG. The MGC may announce a handover to the MG by sending it ServiceChange command. The MGC may also use ServiceChange to instruct the MG to take a Termination or group of Terminations in or out of service.

All of these commands are sent from the MGC to the MG, although ServiceChange can also be sent by the MG. The Notify command, with which the MG informs the MGC that one of the events the MGC was interested in has occurred, is sent by the MG to the MGC.

Related protocols

RTP, RTSP, SIP, H.323, MGCP

Sponsor Source

Megaco/H.248 v1 is defined by IETF (www.ietf.org) and ITU-T. Megaco/H.248 version 2 is in drafting status.

Reference

http://www.javvin.com/protocol/rfc3525.pdf
Gateway Control Protocol Version 1
http://www.javvin.com/protocol/megaco-h248v2.pdf
The Megaco/H.248 Gateway Control Protocol, version 2

Protocol Name

MGCP: Media Gateway Control Protocol

Protocol Description

Media Gateway Control Protocol (MGCP) is a VOIP protocol used between elements of a decomposed multimedia gateway which consists of a Call Agent, containing the call control “intelligence”, and a media gateway containing the media functions, e.g., conversion from TDM voice to Voice over IP.

Media gateways contain endpoints on which the Call Agent can create, modify and delete connections in order to establish and control media sessions with other multimedia endpoints. A media gateway is typically a network element that provides conversion between the audio signals carried on telephone circuits and data packets carried over the Internet or over other packet networks. The Call Agent can instruct the endpoints to detect certain events and generate signals. The endpoints automatically communicate changes in service state to the Call Agent. Furthermore, the Call Agent can audit endpoints as well as the connections on endpoints.

MGCP assumes a call control architecture where the call control “intelligence” is outside the gateways and handled by Call Agents. It assumes that Call Agents will synchronize with each other to send coherent commands and responses to the gateways under their control. MGCP does not define a mechanism for synchronizing Call Agents. MGCP is, in essence, a master/slave protocol, where the gateways are expected to execute commands sent by the Call Agents.

MGCP assumes a connection model where the basic constructs are endpoints and connections. Endpoints are sources and/or sinks of data and can be physical or virtual. Creation of physical endpoints requires hardware installation, while creation of virtual endpoints can be done by software. Connections may be either point to point or multipoint. A point to point connection is an association between two endpoints with the purpose of transmitting data between these endpoints. Once this association is established for both endpoints, data transfer between these endpoints can take place. A multipoint connection is established by connecting the endpoint to a multipoint session. Connections can be established over several types of bearer networks.

In the MGCP model, the gateways focus on the audio signal translation function, while the Call Agent handles the call signaling and call processing functions. As a consequence, the Call Agent implements the “signaling” layers of the H.323 standard, and presents itself as an “H.323 Gatekeeper” or as one or more “H.323 Endpoints” to the H.323 systems.

Protocol Structure

The MGCP is a text based protocol. The transactions are composed of a command and a mandatory response. There are eight types of commands:

MGC --> MG	CreateConnection: Creates a connection between two endpoints; uses SDP to define the receive capabilities of the participating endpoints.
MGC --> MG	ModifyConnection: Modifies the properties of a connection; has nearly the same parameters as the CreateConnection command.
MGC <--> MG	DeleteConnection: Terminates a connection and collects statistics on the execution of the connection.
MGC --> MG	NotificationRequest: Requests the media gateway to send notifications on the occurrence of specified events in an endpoint.
MGC <-- MG	Notify: Informs the media gateway controller when observed events occur.
MGC --> MG	AuditEndpoint: Determines the status of an endpoint.
MGC --> MG	AuditConnection: Retrieves the parameters related to a connection.
MGC <-- MG	RestartInProgress: Signals that an endpoint or group of endpoints is taken in or out of service.

Related protocols

RTP, RTSP, SIP, H.323, Megaco, H.248

Sponsor Source

MGCP is a Cisco protocol.

Reference

http://www.javvin.com/protocol/rfc3435.pdf
Media Gateway Control Protocol (MGCP) Version 1.0.
http://www.javvin.com/protocol/rfc3661.pdf
Media Gateway Control Protocol (MGCP) Return Code Usage

Protocol Name

RTSP: Real-Time Streaming Protocol

Protocol Description

The Real-Time Streaming Protocol (RTSP) establishes and controls either a single or several time-synchronized streams of continuous media such as audio and video. RTSP does not typically deliver the continuous streams itself, although interleaving of the continuous media stream with the control stream is possible. In other words, RTSP acts as a "network remote control" for multimedia servers. RTSP provides an extensible framework to enable controlled, on-demand delivery of real-time data, such as audio and video. Sources of data can include both live data feeds and stored clips. RTSP is intended to control multiple data delivery sessions, provide a means for choosing delivery channels, such as UDP, multicast UDP and TCP, and provide a means for choosing delivery mechanisms based upon RTP.

There is no notion of an RTSP connection; instead, a server maintains a session labeled by an identifier. An RTSP session is in no way tied to a transport-level connection such as a TCP connection. During an RTSP session, an RTSP client may open and close many reliable transport connections to the server to issue RTSP requests. Alternatively, it may use a connectionless transport protocol such as UDP.

The streams controlled by RTSP may use RTP, but the operation of RTSP does not depend on the transport mechanism used to carry continuous media. RTSP is intentionally similar in syntax and operation to HTTP/1.1 so that extension mechanisms to HTTP can in most cases also be added to RTSP. However, RTSP differs in a number of important aspects from HTTP:

- RTSP introduces a number of new methods and has a different protocol identifier.
- An RTSP server needs to maintain state by default in almost all cases, as opposed to the stateless nature of HTTP.
- Both an RTSP server and client can issue requests.
- Data is carried out-of-band by a different protocol, in most cases.
- RTSP is defined to use ISO 10646 (UTF-8) rather than ISO 8859-1, consistent with current HTML internationalization efforts.
- The Request-URI always contains the absolute URI. Because of backward compatibility with a historical blunder, HTTP/1.1 carries only the absolute path in the request and puts the host name in a separate header field.

The protocol supports the following operations:

- Retrieval of media from a media server: The client can request a presentation description via HTTP or some other method.
- Invitation of a media server to a conference: A media server can be "invited" to join an existing conference, either to play back media into the presentation or to record all or a subset of the media in a presentation.
- Addition of media to an existing presentation: Particularly for live presentations, it is useful if the server can tell the client about additional media becoming available.

Protocol Structure

RTSP is a text-based protocol and uses the ISO 10646 character set in UTF-8 encoding. Lines are terminated by CRLF, but receivers should be prepared to also interpret CR and LF by themselves as line terminators. The header fields are summarized as follows:

Header	*type*	*support*	*methods*
Accept	R	opt.	entity
Accept-Encoding	R	opt.	entity
Accept-Language	R	opt.	all
Allow	r	opt.	all
Authorization	R	opt.	all
Bandwidth	R	opt.	all
Blocksize	R	opt.	All but OPTIONS, TEARDOWN
Cache-Control	g	opt.	SETUP
Conference	R	opt.	SETUP
Connection	g	req.	all
Content-Base	e	opt.	entity
Content-Encoding	e	req.	SET_PARAMETER
Content-Encoding	e	req.	DESCRIBE, ANNOUNCE
Content-Language	e	req.	DESCRIBE, ANNOUNCE
Content-Length	e	req.	SET_PARAMETER, ANNOUNCE
Content-Length	e	req.	entity
Content-Location	e	opt.	entity
Content-Type	e	req.	SET_PARAMETER, ANNOUNCE
Content-Type	r	req.	entity
CSeq	g	req.	all
Date	g	opt.	all
Expires	e	opt.	DESCRIBE, ANNOUNCE
From	R	opt.	all
If-Modified-Since	R	opt.	DESCRIBE, SETUP
Last-Modified	e	opt.	entity
Proxy-Authenticate			
Proxy-Require	R	req.	all
Public	r	opt.	all
Range	R	opt.	PLAY, PAUSE, RE-

			CORD
Range	r	opt.	PLAY, PAUSE, RECORD
Referer	R	opt.	all
Require	R	req.	all
Retry-After	r	opt.	all
RTP-Info	r	req.	PLAY
Scale	Rr	opt.	PLAY, RECORD
Session	Rr	req.	All but SETUP, OPTIONS
Server	r	opt.	all
Speed	Rr	opt.	PLAY
Transport	Rr	req.	SETUP
Unsupported	r	req.	all
User-Agent	R	opt.	all
Via	g	opt.	all
WWW-Authenticate	r	opt.	all

Type “g” designates general request headers to be found in both requests and responses, type “R” designates request headers, type “r” designates response headers, and type “e” designates entity header fields. Fields marked with “req.” in the column labeled “support” MUST be implemented by the recipient for a particular method, while fields marked “opt.” are optional. Note that not all fields marked “req.” will be sent in every request of this type. The “req.” means only that client (for response headers) and server (for request headers) MUST implement the fields. The last column lists the method for which this header field is meaningful; the designation “entity” refers to all methods that return a message body.

Related protocols

UDP, TCP, HTTP, S-HTTP, RTP

Sponsor Source

RTSP is defined by IETF (www.ietf.org) in RFC 2326.

Reference

http://www.javvin.com/protocol/rfc2326.pdf
Real Time Streaming Protocol

Protocol Name

SAP: Session Announcement Protocol

Protocol Description

Session Announcement Protocol (SAP) is an announcement protocol that is used to assist the advertisement of multicast multimedia conferences and other multicast sessions, and to communicate the relevant session setup information to prospective participants.

A SAP announcer periodically multicasts an announcement packet to a well-known multicast address and port. The announcement is multicast with the same scope as the session it is announcing, ensuring that the recipients of the announcement can also be potential recipients of the session the announcement describes (bandwidth and other such constraints permitting). This is also important for the scalability of the protocol, as it keeps local session announcements local.

A SAP listener learns of the multicast scopes it is within (for example, using the Multicast-Scope Zone Announcement Protocol) and listens on the well known SAP address and port for those scopes. In this manner, it will eventually learn of all the sessions being announced, allowing those sessions to be joined.

It is to be expected that sessions may be announced by a number of different mechanisms, not only SAP. For example, a session description may be placed on a web page, sent by email or conveyed in a session initiation protocol. To increase interoperability with these other mechanisms, application level security is employed, rather than using IPsec authentication headers.

Protocol Structure

3	4	5	6	7	8	16	32bit
V=1	A	R	T	E	C	Auth len	Msg ID hash
Originating source (32 or 128 bits)							
Optional Authentication Data							
Optional timeout							
Optional payload type							
						0	
Payload							

- V - Version Number field is three bits and MUST be set to 1.
- A: Address Type can have a value of 0 or 1:
  - 0 The originating source field contains a 32-bit IPv4 address.
  - 1 The originating source contains a 128-bit IPv6 address.
- R - Reserved. SAP announcers set this to 0. SAP listeners ignore the contents of this field.
- T - Message Type can have a value of 0 or 1:
  - 0 Session announcement packet
  - 1 Session deletion packet.
- E - Encryption Bit bit may be 0 or 1.
  - 1 The payload of the SAP packet is encrypted and the timeout field must be added to the packet header.
  - 0 The packet is not encrypted and the timeout must not be present.
- C - Compressed Bit. If the compressed bit is set to 1, the payload is compressed.
- Authentication Length - An 8bits unsigned quantity giving the number of 32 bit words, following the main SAP header, that contain authentication data. If it is zero, no authentication header is present.
- Message Identifier Hash - used in combination with the originating source, provides a globally unique identifier indicating the precise version of this announcement.
- Originating Source - This field contains the IP address of the original source of the message. This is an IPv4 address if the A field is set to zero; otherwise, it is an IPv6 address. The address is stored in network byte order.
- Timeout - When the session payload is encrypted, the detailed timing fields in the payload are not available to listeners not trusted with the decryption key. Under such circumstances, the header includes an additional 32-bit timestamp field stating when the session should be timed out. The value is an unsigned quantity giving the NTP time in seconds at which time the session is timed out. It is in network byte order.
- Payload Type - The payload type field is a MIME content type specifier, describing the format of the payload. This is a variable length ASCII text string, followed by a single zero byte (ASCII NUL).
- Payload - The Payload field includes various sub fields.

Related protocols

RTP, RTSP, SIP, SDP

Sponsor Source

SAP is defined by IETF (www.ietf.org) in RFC 2974.

Reference

http://www.javvin.com/protocol/rfc2974.pdf
SAP Session Announcement Protocol

Protocol Name

SDP: Session Description Protocol

Protocol Description

The Session Description Protocol (SDP) describes multimedia sessions for the purpose of session announcement, session invitation and other forms of multimedia session initiation.

Session directories assist the advertisement of conference sessions and communicate the relevant conference setup information to prospective participants. SDP is designed to convey such information to recipients. SDP is purely a format for session description - it does not incorporate a transport protocol, and is intended to use different transport protocols as appropriate including the Session Announcement Protocol (SAP), Session Initiation Protocol (SIP), Real-Time Streaming Protocol (RTSP), electronic mail using the MIME extensions, and the Hypertext Transport Protocol (HTTP).

SDP is intended to be general purpose so that it can be used for a wider range of network environments and applications than just multicast session directories. However, it is not intended to support negotiation of session content or media encodings.

On Internet Multicast backbone (Mbone) a session directory tool is used to advertise multimedia conferences and communicate the conference addresses and conference tool-specific information necessary for participation. The SDP does this. It communicates the existence of a session and conveys sufficient information to enable participation in the session. Many of the SDP messages are sent by periodically multicasting an announcement packet to a well-known multicast address and port using SAP (Session Announcement Protocol). These messages are UDP packets with a SAP header and a text payload. The text payload is the SDP session description. Messages can also be sent using email or the WWW (World Wide Web).

The SDP text messages include:

- Session name and purpose
- Time the session is active
- Media comprising the session
- Information to receive the media (address etc.)

Protocol Structure

SDP messages are text messages using the ISO 10646 character set in UTF-8 encoding. SDP Session description (optional fields have an *) is:

v= (protocol version)
o= (owner/creator and session identifier).
s= (session name)
i=* (session information)
u=* (URI of description)
e=* (email address)
p=* (phone number)
c=* (connection information - not required if included in all media)
b=* (bandwidth information)
One or more time descriptions (see below)
z=* (time zone adjustments)
k=* (encryption key)
a=* (zero or more session attribute lines)
Zero or more media descriptions (see below)

Time description

t= (time the session is active)
r=* (zero or more repeat times)

Media description

m= (media name and transport address)
i=* (media title)
c=* (connection information - optional if included at session-level)
b=* (bandwidth information)
k=* (encryption key)
a=* (zero or more media attribute lines)

Related protocols

UDP, TCP, SIP, SAP, RTSP, RTP, H.320, MPEG, H.261, HTTP, MIME

Sponsor Source

SDP is defined by IETF (www.ietf.org) in RFC 2327.

Reference

http://www.javvin.com/protocol/rfc2327.pdf
Session Description Protocol

Protocol Name

SIP: Session Initiation Protocol

Protocol Description

Session Initiation Protocol (SIP) is an application-layer control protocol that can establish, modify, and terminate multimedia sessions such as Internet telephony calls. SIP can also invite participants to already existing sessions, such as multicast conferences. Media can be added to (and removed from) an existing session. SIP transparently supports name mapping and redirection services, which supports personal mobility - users can maintain a single externally visible identifier regardless of their network location.

SIP supports five facets of establishing and terminating multimedia communications:

User location: determination of the end system to be used for communication;

User availability: determination of the willingness of the called party to engage in communications;

User capabilities: determination of the media and media parameters to be used;

Session setup: “ringing”, establishment of session parameters at both called and calling party;

Session management: including transfer and termination of sessions, modifying session parameters, and invoking services.

SIP is a component that can be used with other IETF protocols to build a complete multimedia architecture, such as the Real-time Transport Protocol (RTP) for transporting real-time data and providing QoS feedback, the Real-Time streaming protocol (RTSP) for controlling delivery of streaming media, the Media Gateway Control Protocol (MEGACO) for controlling gateways to the Public Switched Telephone Network (PSTN), and the Session Description Protocol (SDP) for describing multimedia sessions. Therefore, SIP should be used in conjunction with other protocols in order to provide complete services to the users. However, the basic functionality and operation of SIP does not depend on any of these protocols.

SIP provides a suite of security services, which include denial-of-service prevention, authentication (both user to user and proxy to user), integrity protection, and encryption and privacy services.

SIP works with both IPv4 and IPv6. For Internet telephony sessions, SIP works as follows:

Callers and callees are identified by SIP addresses. When making a SIP call, a caller first locates the appropriate server and then sends a SIP request. The most common SIP operation is the invitation. Instead of directly reaching the intended callee, a SIP request may be redirected or may trigger a chain of new SIP requests by proxies. Users can register their location(s) with SIP servers. SIP addresses (URLs) can be embedded in Web pages and therefore can be integrated as part of such powerful implementations as Click to talk.

Protocol Structure

SIP messages can be transmitted either over TCP or UDP SIP messages are text based and use the ISO 10646 character set in UTF-8 encoding. Lines must be terminated with CRLF. Much of the message syntax and header field are similar to HTTP. Messages can be request messages or response messages.

A request message has the following format:

Method	Request URI	SIP version

- Method – The method to be performed on the resource. Possible methods are Invite, Ack, Options, Bye, Cancel, Register.
- Request-URI - A SIP URL or a general Uniform Resource Identifier; this is the user or service to which this request is being addressed.
- SIP version - The SIP version being used.

The format of the Response message header is shown in the following illustration:

SIP version	Status code	Reason phrase

- SIP version - The SIP version being used.
- Status-code - A 3-digit integer result code of the attempt to understand and satisfy the request.
- Reason-phrase – A textual description of the status code.

Related protocols

UDP, TCP, IP, RTSP, RTP, HTTP, SDP, MEGACO

Sponsor Source

SIP is defined by IETF (www.ietf.org) in RFC 3261, 3262, 3263, 3264, and 3265.

Reference

http://www.javvin.com/protocol/rfc3261.pdf
Session Initiation Protocol.
http://www.javvin.com/protocol/rfc3262.pdf
Reliability of Provisional Responses in the Session Initiation Protocol (SIP)
http://www.javvin.com/protocol/rfc3263.pdf

Session Initiation Protocol (SIP): Locating SIP Servers
http://www.javvin.com/protocol/rfc3264.pdf
An Offer/Answer Model with the Session Description Protocol (SDP)
http://www.javvin.com/protocol/rfc3265.pdf
Session Initiation Protocol (SIP)-Specific Event Notification

Protocol Name

SCCP (Skinny): Cisco Skinny Client Control Protocol

Protocol Description

Skinny Client Control Protocol (SCCP or Skinny) is a Cisco proprietary protocol used between Cisco Call Manager and Cisco VOIP phones. It is also supported by some other vendors.

For VOIP solutions, the end station of a LAN or IP- based PBX must be simple to use, familiar and relatively cheap. SCCP defines a simple and easy to use architecture, while the H.323 recommendations produce quite an expensive system. An H.323 proxy can be used to communicate with the Skinny Client using the SCCP. In such a case the telephone is a skinny client over IP, in the context of H.323. A proxy is used for the H.225 and H.245 signalling.

With the SCCP architecture, the vast majority of the H.323 processing power resides in an H.323 proxy known as the Cisco Call Manager. The end stations (telephones) run what is called the Skinny Client, which consumes less processing overhead. The Client communicates with the Call Manager using connection-oriented (TCP/IP-based) communication to establish a call with another H.323-compliant end station. Once the Call Manager has established the call, the two H.323 end stations use connectionless (UDP/IP-based) communication for audio transmissions. Costs and overhead are thus reduced by confining the complexities of H.323 call setup to the Call Manager, and using the Skinny protocol for the actual audio communication into and out of the end stations.

Protocol Structure

The skinny client (i.e. an Ethernet Phone) uses TCP/IP to transmit and receive calls and RTP/UDP/IP to/from a Skinny Client or H.323 terminal for audio. Skinny messages are carried above TCP and use port 2000. The message types are as follows:

Code	Station Message ID Message
0x0000	Keep Alive Message
0x0001	Station Register Message
0x0002	Station IP Port Message
0x0003	Station Key Pad Button Message
0x0004	Station Enbloc Call Message
0x0005	Station Stimulus Message
0x0006	Station Off Hook Message
0x0007	Station On Hook Message
0x0008	Station Hook Flash Message
0x0009	Station Forward Status Request Message
0x11	Station Media Port List Message
0x000A	Station Speed Dial Status Request Message
0x000B	Station Line Status Request Message
0x000C	Station Configuration Status Request Message
0x000D	Station Time Date Request Message
0x000E	Station Button Template Request Message
0x000F	Station Version Request Message
0x0010	Station Capabilities Response Message
0x0012	Station Server Request Message
0x0020	Station Alarm Message
0x0021	Station Multicast Media Reception Ack Message
0x0024	Station Off Hook With Calling Party Number Message
0x22	Station Open Receive Channel Ack Message
0x23	Station Connection Statistics Response Message
0x25	Station Soft Key Template Request Message
0x26	Station Soft Key Set Request Message
0x27	Station Soft Key Event Message
0x28	Station Unregister Message
0x0081	Station Keep Alive Message
0x0082	Station Start Tone Message
0x0083	Station Stop Tone Message
0x0085	Station Set Ringer Message
0x0086	Station Set Lamp Message
0x0087	Station Set Hook Flash Detect Message
0x0088	Station Set Speaker Mode Message
0x0089	Station Set Microphone Mode Message
0x008A	Station Start Media Transmission
0x008B	Station Stop Media Transmission
0x008F	Station Call Information Message
0x009D	Station Register Reject Message
0x009F	Station Reset Message
0x0090	Station Forward Status Message
0x0091	Station Speed Dial Status Message
0x0092	Station Line Status Message
0x0093	Station Configuration Status Message
0x0094	Station Define Time & Date Message
0x0095	Station Start Session Transmission Message
0x0096	Station Stop Session Transmission Message
0x0097	Station Button Template Message
0x0098	Station Version Message
0x0099	Station Display Text Message
0x009A	Station Clear Display Message
0x009B	Station Capabilities Request Message
0x009C	Station Enunciator Command Message
0x009E	Station Server Respond Message
0x0101	Station Start Multicast Media Reception Message
0x0102	Station Start Multicast Media Transmission Message
0x0103	Station Stop Multicast Media Reception Message
0x0104	Station Stop Multicast Media Transmission Message
0x105	Station Open Receive Channel Message

0x0106	Station Close Receive Channel Message
0x107	Station Connection Statistics Request Message
0x0108	Station Soft Key Template Respond Message
0x109	Station Soft Key Set Respond Message
0x0110	Station Select Soft Keys Message
0x0111	Station Call State Message
0x0112	Station Display Prompt Message
0x0113	Station Clear Prompt Message
0x0114	Station Display Notify Message
0x0115	Station Clear Notify Message
0x0116	Station Activate Call Plane Message
0x0117	Station Deactivate Call Plane Message
0x118	Station Unregister Ack Message

Related protocols

RTP, RTSP, SIP, H.323, Megaco, H.248

Sponsor Source

SCCP/Skinny is a Cisco protocol.

Protocol Name

T.120: Multipoint Data Conferencing and Real Time Communication Protocols

Protocol Description

The ITU T.120 standard is made up of a suite of communication and application protocols. T.120 protocols are designed for multipoint Data Conferencing and real time communication including multilayer protocols which considerably enhance multimedia, MCU and codec control capabilities. Depending on the type of T.120 implementations, the resulting product can make connections, transmit and receive data, and collaborate using compatible data conferencing features, such as program sharing, whiteboard conferencing, and file transfer. The key functionalities of T.120 are:

- Establish and maintain conferences without any platform dependence.
- Manage multiple participants and programs.
- Send and receive data accurately and securely over a variety of supported networking connections.

The T.120 protocol suite includes the following protocols:

T.121 provides a template for T.120 resource management that developers should use as a guide for building application protocols. T.121 is mandatory for standardized application protocols and is highly recommended for non-standard application protocols. The template ensures consistency and reduces the potential for unforeseen interaction between different protocol implementations.

T.122 defines the multi-point services available to the developer. Together with T.125, it forms MCS, the multi-point “engine” of the T.120 conference. MCS relies on T.123 to actually deliver the data. MCS is a powerful tool that can be used to solve virtually any multi-point application design requirement. MCS is an elegant abstraction of a rather complex organism. Learning to use MCS effectively is the key to successfully developing real-time applications.

T.123 specifies transport profiles for each of the following: 1) Public Switched Telephone Networks (PSTN) 2) Integrated Switched Digital Networks (ISDN); 3) Circuit Switched Digital Networks (CSDN); 4)Packet Switched Digital Networks (PSDN); 5) Novell Netware IPX (via reference profile); and 6) TCP/IP (via reference profile). T.120 applications expect the underlying transport to provide reliable delivery of its Protocol Data Units (PDUs) and to segment and sequence that data.

T.124 specifies the Generic Conference Control (GCC), which provides a comprehensive set of facilities for establishing and managing the multi-point conference. It is with GCC that we first see features that are specific to the electronic meeting.

T.125 describes the Multipoint Communication Service Protocol (MCS). It defines: 1) Procedures for a single protocol for the transfer of data and control information from one MCS provider to a peer MCS provider; and 2) The structure and encoding of the MCS protocol data units used for the transfer of data and control information.

T.126 defines a protocol for viewing and annotating still images transmitted between two or more applications. This capability is often referred to as document conferencing or shared whiteboarding.

T.127 specifies a means for applications to transmit files between multiple endpoints in a conference. Files can be transferred to all participants in the conference or to a specified subset of the conference. Multiple file transfer operations may occur simultaneously in any given conference and developers can specify priority levels for the file delivery. Finally, T.127 provides options for compressing files before delivering the data.

Protocol Structure

The T.120 architecture relies on a multi-layered approach with defined protocols and service definitions between layers. Each layer presumes the existence of all layers below. The lower level layers (T.122, T.123, T.124, and T.125) specify an application-independent mechanism for providing multi-point data communications services to any application that can use these facilities. The upper level layers (T.126 and T.127) define protocols for specific conferencing applications, such as shared whiteboarding and binary file transfer. These “standardized applications” can co-exist in the same conference with “non-standardized” applications such as a business card exchange program or a textual chat application. The following figure represent the architecture of the T.120.

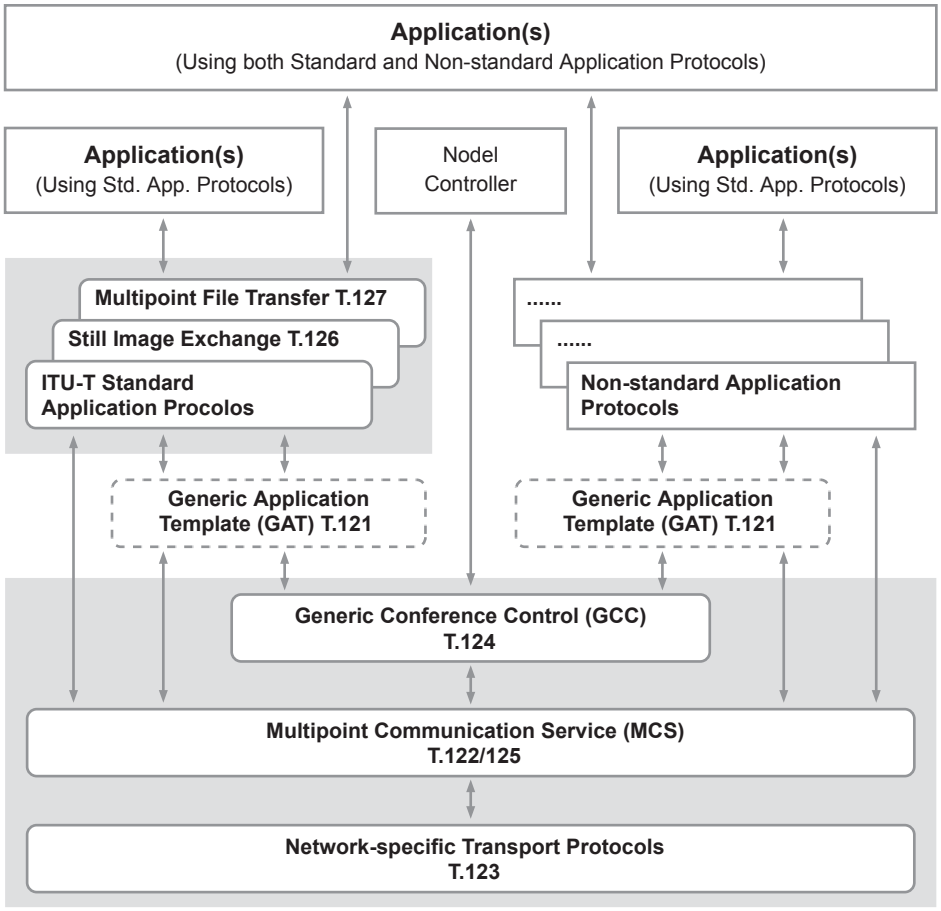


Figure 2-9: T.120 Data Conferencing Protocol Structure

Related protocols

RTP, RTSP, SIP, Megaco, H.248, Q.931, H.323, H.225

Sponsor Source

T.120, T.121, T.122, T.123, T.124, T.125, T.126, T.127 are ITU-T (http://www.itu.int/ITU-T/) standards.

Reference

http://www.packetizer.com/conf/t120/primer/
A Primer on the T.120 Series Standard

Media / CODEC

Protocol Name

G.7xx: Audio (Voice) Compression Protocols

Protocol Description

G.7xx is a suite of ITU-T standards for audio compression and de-commpressions. It is primarily used in telephony. In telephony, there are 2 main algorithms defined in the standard, mu-law algorithm (used in America) and a-law algorithm (used in Europe and the rest of the world). Both are logarithmic, but the later a-law was specifically designed to be simpler for a computer to process. The G.7xx protocol suite is composed of the following protocols:

G.711 Pulse code modulation (PCM) of voice frequencies on a 64 kbps channel.

G.721 32 kbit/s adaptive differential pulse code modulation (ADPCM)

G.722 7 kHz audio-coding within 64 kbit/s

G.722.1 Coding at 24 and 32 kbit/s for hands-free operation in systems with low frame loss

G.722.2 Wideband coding of speech at around 16 kbit/s using adaptive multi-rate wideband (AMR-WB)

G.726 40, 32, 24, 16 kbit/s adaptive differential pulse code modulation (ADPCM)

G.727 5-, 4-, 3- and 2-bit/sample embedded adaptive differential pulse code modulation (ADPCM)

G.728 Coding of speech at 16 kbit/s using low-delay code excited linear prediction

G.729 Coding of speech at 8 kbit/s using conjugate-structure algebraic-code-excited linear-prediction (CS-ACELP)

Protocol Structure

Name	standardized by	description	bit rate (kb/s)	sampling rate (kHz)	frame size (ms)	remarks
(ADPCM) DVI	Intel, IMA	ADPCM	32	8	sample	
G.711	ITU-T	Pulse code modulation (PCM)	64	8	sample	mu-law (US, Japan) and A-law (Europe) companding
G.721	ITU-T	Adaptive differential pulse code modulation (ADPCM)	32	8	sample	Now described in G.726; obsolete.
G.722	ITU-T	7 kHz audio-coding within 64 kbit/s	64	16	sample	Subband-codec that divides 16 kHz band into two subbands, each coded using ADPCM
G.722.1	ITU-T	Coding at 24 and 32 kbit/s for hands-free operation in systems with low frame loss	24/32	16	20	
G.723	ITU-T	Extensions of Recommendation G.721 adaptive differential pulse code modulation to 24 and 40 kbit/s for digital circuit multiplication equipment application	24/40	8	sample	Superceded by G.726; obsolete. This is a completely different codec than G.723.1.
G.723.1	ITU-T	Dual rate speech coder for multimedia communications transmitting at 5.3 and 6.3 kbit/s	5.6/6.3	8	30	Part of H.324 video conferencing. It encodes speech or other audio signals in frames using linear predictive analysis-by-synthesis coding. The excitation signal for the high rate coder is Multipulse Maximum Likelihood Quantization (MP-MLQ) and for the low rate coder is Algebraic-Code-Excited Linear-Prediction (ACELP).
G.726	ITU-T	40, 32, 24, 16 kbit/s adaptive differential pulse code modulation (ADPCM)	16 / 24 / 32 / 40	8	sample	ADPCM; replaces G.721 and G.723.
G.727	ITU-T	5-, 4-, 3- and 2-bit/sample embedded adaptive differential pulse code modulation (ADPCM)	var.	?	sample	ADPCM. Related to G.726.
G.728	ITU-T	Coding of speech at 16 kbit/s using low-delay code excited linear prediction	16	8		CELP. Annex J offers variable-bit rate operation for DCME.
G.729	ITU-T	Coding of speech at 8 kbit/s using conjugate-structure algebraic-code-excited linear-prediction (CS-ACELP)	8	8	10	Low delay (15 ms)
GSM 06.10	ETSI	Regular Pulse Excitation Long Term Predictor (RPE-LTP)	13	8	22.5	Used for GSM cellular telephony.

Related protocols

RTP, RTSP, H.323, H.225

Sponsor Source

G.7xxis a suite of ITU-T (http://www.itu.int/ITU-T/) standards.

Reference

http://www.h323forum.org/papers/
H.323 papers and documents

Protocol Name

H.261: Video Coding and Decoding (CODEC)

Protocol Description

H.261 is the video coding standard of the ITU. It was designed for data rates which are multiples of 64Kbit/s and is sometimes called p x 64Kbit/s (p is in the range 1-30). These data rates suit ISDN lines, for which this video codec was originally designed. H.261 transports a video stream using the real-time transport protocol, RTP, with any of the underlying protocols that carry RTP.

The coding algorithm is a hybrid of inter-picture prediction, transform coding and motion compensation. The data rate of the coding algorithm was designed to be able to be set to between 40 Kbits/s and 2 Mbits/s. INTRA coding encodes blocks of 8x8 pixels each only with reference to themselves and sends them directly to the block transformation process. On the other hand, INTER coding frames are encoded with respect to another reference frame. The inter-picture prediction removes temporal redundancy. The transform coding removes the spatial redundancy. Motion vectors are used to help the codec compensate for motion. To remove any further redundancy in the transmitted bitstream, variable length coding is used.

H261 supports motion compensation in the encoder as an option. In motion compensation a search area is constructed in the previous (recovered) frame to determine the best reference macroblock.

H261 supports two image resolutions, QCIF (Quarter Common Interchange format) which is (144x176 pixels) and CIF (Common Interchange format), which is (288x352).

The video multiplexer structures the compressed data into a hierarchical bitstream that can be universally interpreted. The hierarchy has four layers:

1. Picture layer: corresponds to one video picture (frame)
2. Group of blocks: corresponds to 1/12 of CIF pictures or 1/3 of QCIF
3. Macroblocks : corresponds to 16x16 pixels of luminance and the two spatially corresponding 8x8 chrominance components.
4. Blocks: corresponds to 8x8 pixels

Protocol Structure

H.261 Header Structure:

3	6	7	8	12	17	22	27	32bit
SBIT	EBIT	I	V	GOBN	MBAP	QUANT	HMVD	VMVD

- SBIT - Start bit. Number of most significant bits that are to be ignored in the first data octet.
- EBIT - End bit. Number of least significant bits that are to be ignored in the last data octet.
- I - INTRA-frame encoded data field. Set to 1 if this stream contains only INTRA-frame coded blocks and to 0 if this stream may or may not contain INTRA-frame coded blocks.
- V - Motion Vector flag. Set to 0 if motion vectors are not used in this stream and to 1 if motion vectors may or may not be used in this stream.
- GOBN - GOB number. Encodes the GOB number in effect at the start of the packet. Set to 0 if the packet begins with a GOB header.
- MBAP - Macroblock address predictor. Encodes the macroblock address predictor (i.e., the last MBA encoded in the previous packet). This predictor ranges from 0 to 32 (to predict the valid MBAs 1-33), but because the bit stream cannot be fragmented between a GOB header and MB 1, the predictor at the start of the packet can never be 0.
- QUANT - Quantizer field. Shows the Quantizer value (MQUANT or GQUANT) in effect prior to the start of this packet. Set to 0 if the packet begins with a GOB header.
- HMVD - Horizontal motion vector data field. Represents the reference horizontal motion vector data (MVD). Set to 0 if V flag is 0 or if the packet begins with a GOB header, or when the MTYPE of the last MB encoded in the previous packet was not MC.
- VMVD - Vertical motion vector data (VMVD). Reference vertical motion vector data (MVD). Set to 0 if V flag is 0 or if the packet begins with a GOB header, or when the MTYPE of the last MB encoded in the previous packet was not MC.

Related protocols

RTP, RTSP, H.248, H.323, H.225, H.245, H.263

Sponsor Source

H.261 is an ITU-T (http://www.itu.int/ITU-T/) standards.

Reference

http://www.javvin.com/protocol/rfc2032.pdf
RTP Payload Format for H.261 Video Streams
http://www.h323forum.org/papers/
H.323 papers and documents

Protocol Name

H.263: Video Coding and Decoding (CODEC)

Protocol Description

The H.263, by the International Telecommunications Union (ITU), supports video compression (coding) for video-conferencing and video-telephony applications. H.263 was developed to stream video at bandwidths as low as 20K to 24K bit/sec and was based on the H.261 codec. As a general rule, H.263 requires half the bandwidth to achieve the same video quality as in the H.261. As a result, H.263 has largely replaced H.261. H.263 uses RTP to transport video streams.

The coding algorithm of H.263 is similar to that used by H.261, however with some improvements and changes to improve performance and error recovery. Half pixel precision is used for motion compensation whereas H.261 used full pixel precision and a loop filter. Some parts of the hierarchical structure of the datastream are now optional, so the codec can be configured for a lower datarate or better error recovery. There are now four optional negotiable options included to improve performance: Unrestricted Motion Vectors, Syntax-based arithmetic coding, Advance prediction, and forward and backward frame prediction similar to MPEG called P-B frames.

H.263 supports five resolutions. In addition to QCIF and CIF that were supported by H.261 there are SQCIF, 4CIF, and 16CIF. SQCIF is approximately half the resolution of QCIF. 4CIF and 16CIF are 4 and 16 times the resolution of CIF respectively. The support of 4CIF and 16CIF means the codec can compete with other higher bitrate video coding standards such as the MPEG standards.

The differences between the H.261 and H.263 coding algorithms are listed as follows:

Picture format	Luminance pixels	Luminance lines	H.261 support	H.263 support	Uncompressed bit rate (Mbits/s)			
					10 frames/s		30 frames/s	
					Grey	Color	Grey	Color
SQCIF	128	96		Yes	1.0	1.5	3.0	4.4
QCIF	176	144	Yes	Yes	2.0	3.0	6.1	9.1
CIF	352	288	Optional	Optional	8.1	12.2	24.3	36.5
4CIF	704	576		Optional	32.4	48.7	97.3	146.0
16CIF	1408	1152		Optional	129.8	194.6	389.3	583.9

Protocol Structure

Three formats (mode A, mode B and mode C) are defined for the H.263 payload header. In mode A, an H.263 payload header of four bytes is present before the actual compressed H.263 video bitstream. It allows fragmentation at GOB boundaries. In mode B, an 8-byte H.263 payload header is used and each packet starts at MB boundaries without the PB-frames option. Finally, a 12-byte H.263 payload header is defined in mode C to support fragmentation at MB boundaries for frames that are coded with the PB-frames option.

The format of the header for mode A is shown in the following illustration:

1	2	5	8	11	12	13	14	15	16bit
F	P	SBIT	EBIT	SRC	I	U	S	A	R
R (cont.)		DBQ	TRB	TR					

- F - Flag bit indicates the mode of the payload header. Values are as follows:
  0 - mode A.
  1 - mode B or mode C depending on P bit.
- P - P bit specifies the optional PB-frames mode.
- SBIT - Start bit position specifies number of most significant bits that are ignored in the first data byte.
- EBIT - End bit position specifies number of least significant bits that are ignored in the last data byte.
- SRC - Source format (bit 6,7 and 8 in PTYPE in the standard H.263 compressed bitstream) specifies the resolution of the current picture.
- I - Picture coding type (bit 9 in PTYPE in the standard H.263 compressed bitstream).
- U - Set to 1 if the Unrestricted Motion Vector option (bit 10 in PTYPE in the standard H.263 compressed bitstream) was set to 1 in the current picture header, otherwise 0.
- S - Set to 1 if the Syntax-based Arithmetic Coding option (bit 11 in PTYPE in the standard H.263 compressed bitstream) was set to 1 for the current picture header, otherwise 0.
- A - Set to 1 if the Advanced Prediction option (bit 12 in PTYPE in the standard H.263 compressed bitstream) was set to 1 for current picture header, otherwise 0.
- R - Reserved; set to zero.
- DBQ - Differential quantization parameter used to calculate quantizer for the B frame based on quantizer for the P frame, when PB-frames option is used. The value should be the same as DBQUANT in the standard H.263 compressed bitstream. Set to zero if PB-frames option is not used.
- TRB - Temporal Reference for the B frame in the standard H.263 compressed bitstream. Set to zero if PB-frames option is not used.
- TR - Temporal Reference for the P frame in the standard H.263 compressed bitstream. Set to zero if the PB-frames option is not used.

The format of the header for mode B is shown here:

1	2	5	8	11	16bit
F	P	SBIT	EBIT	SRC	QUANT
GOBN			MBR		R
I	U	S	A	HMV1	VMV1	HMV2	VMV2

- F, P, SBIT, EBIT, SRC, I, U, S and A are defined the same as in mode A.
- QUANT - Quantization value for the first MB coded at the starting of the packet. Set to 0 if the packet begins with a GOB header.
- GOBN - GOB number in effect at the start of the packet. GOB number is specified differently for different resolutions.
- MBA - The address within the GOB of the first MB in the packet, counting from zero in scan order. For example, the third MB in any GOB is given MBA=2.
- R - Reserved, set to zero.
- HMV1, VMV1 - Horizontal and vertical motion vector predictors for the first MB in this packet. When four motion vectors are used for the current MB with advanced prediction option, they are the motion vector predictors for block number 1 in the MB. Each 7-bit field encodes a motion vector predictor in half pixel resolution as a 2's complement number.
- HMV2, VMV2 - Horizontal and vertical motion vector predictors for block number 3 in the first MB in this packet when four motion vectors are used with the advanced prediction option. This is needed because block number 3 in the MB needs different motion vector predictors from other blocks in the MB. These two fields are not used when the MB only has one motion vector. Each 7 bits field encodes a motion vector predictor in half pixel resolution as a 2's complement number.

The format of the header for mode C is shown here:

1	2	5	8	11	16bit
F	P	SBIT	EBIT	SRC	QUANT
GOBN			MBR		R
I	U	S	A	HMV1	VMV1	HMV2	VMV2
RR							
RR(c)	DBR		TRB	TR			

- F, P, SBIT, EBIT, SRC, I, U, S, A, DBQ, TRB and TR are defined as in mode A. QUANT, GOBN, MBA, HMV1, VMV1, HMV2, VNV2 are defined the same as in mode B.
- RR - Reserved, set to zero (19 bits).

Related protocols

RTP, RTSP, H.245, H.323, H.225, H.261

Sponsor Source

H.263 is an ITU-T (http://www.itu.int/ITU-T/) standards.

Reference

http://www.javvin.com/protocol/rfc2190.pdf
RTP Payload Format for H.263 Video Streams
http://www.h323forum.org/papers/
H.323 papers and documents

Protocol Name

RTP: Real-Time Transport Protocol

Protocol Description

The Real-time Transport Protocol (RTP) provides end-to-end delivery services for data with real-time characteristics, such as interactive audio and video or simulation data, over multicast or unicast network services. Applications typically run RTP on top of UDP to make use of its multiplexing and checksum services; both protocols contribute parts of the transport protocol functionality. However, RTP may be used with other suitable underlying network or transport protocols. RTP supports data transfer to multiple destinations using multicast distribution if provided by the underlying network.

RTP itself does not provide any mechanism to ensure timely delivery or provide other quality-of-service guarantees but relies on lower-layer services to do so. It does not guarantee delivery or prevent out-of-order delivery, nor does it assume that the underlying network is reliable. It delivers packets in sequence. The sequence numbers included in RTP allow the receiver to reconstruct the sender's packet sequence but sequence numbers might also be used to determine the proper location of a packet, for example in video decoding, without necessarily decoding packets in sequence.

RTP consists of two closely-linked parts:

The real-time transport protocol (RTP), to carry data that has real-time properties.

The RTP control protocol (RTCP), to monitor the quality of service and to convey information about the participants in an on-going session. The latter aspect of RTCP may be sufficient for "loosely controlled" sessions, i.e., where there is no explicit membership control and set-up, but it is not necessarily intended to support all of an application's control communication requirements.

Protocol Structure

<table>
<tr><td>2</td><td>3</td><td>4</td><td>8</td><td>9</td><td>16bit</td></tr>
<tr><td>V</td><td>P</td><td>X</td><td>CSRC count</td><td>M</td><td>Payload type</td></tr>
<tr><td colspan="4">Sequence number</td><td colspan="2">Timestamp</td></tr>
<tr><td colspan="4">SSRC</td><td colspan="2">CSRC (variable 0 – 15 items 32bits each)</td></tr>
</table>

- V - Version. Identifies the RTP version.
- P - Padding. When set, the packet contains one or more additional padding octets at the end which are not part of the payload.
- X - Extension bit. When set, the fixed header is followed by exactly one header extension, with a defined format.
- CSRC count -Contains the number of CSRC identifiers that follow the fixed header.
- M - Marker. The interpretation of the marker is defined by a profile. It is intended to allow significant events such as frame boundaries to be marked in the packet stream.
- Payload type - Identifies the format of the RTP payload and determines its interpretation by the application. A profile specifies a default static mapping of payload type codes to payload formats. Additional payload type codes may be defined dynamically through non-RTP means.
- Sequence number - Increments by one for each RTP data packet sent, and may be used by the receiver to detect packet loss and to restore packet sequence.
- Timestamp - Reflects the sampling instant of the first octet in the RTP data packet. The sampling instant must be derived from a clock that increments monotonically and linearly in time to allow synchronization and jitter calculations.
- SSRC - Synchronization source. This identifier is chosen randomly, with the intent that no two synchronization sources within the same RTP session will have the same SSRC identifier.
- CSRC - Contributing source identifiers list. Identifies the contributing sources for the payload contained in this packet.

Related protocols

RTCP, RTSP, UDP, TCP, IP

Sponsor Source

RTP is defined by IETF (www.ietf.org) in RFC 3550 and 3551.

Reference

http://www.javvin.com/protocol/rfc3550.pdf
RTP: A Transport Protocol for Real-Time Applications
http://www.javvin.com/protocol/rfc3551.pdf
RTP Profile for Audio and Video Conferences with Minimal Control

Protocol Name

RTCP: RTP Control Protocol

Protocol Description

The RTP control protocol (RTCP) is based on the periodic transmission of control packets to all participants in the session, using the same distribution mechanism as the data packets. The underlying protocol must provide multiplexing of the data and control packets, for example using separate port numbers with UDP. RTCP performs four functions:

1. RTCP provides feedback on the quality of the data distribution. This is an integral part of the RTP's role as a transport protocol and is related to the flow and congestion control functions of other transport protocols.
2. RTCP carries a persistent transport-level identifier for an RTP source called the canonical name or CNAME. Since the SSRC identifier may change if a conflict is discovered or a program is restarted, receivers require the CNAME to keep track of each participant. Receivers may also require the CNAME to associate multiple data streams from a given participant in a set of related RTP sessions, for example to synchronize audio and video.
3. The first two functions require that all participants send RTCP packets, therefore the rate must be controlled in order for RTP to scale up to a large number of participants. By having each participant send its control packets to all the others, each can independently observe the number of participants. This number is used to calculate the rate at which the packets are sent.
4. An OPTIONAL function is to convey minimal session control information, for example participant identification to be displayed in the user interface. This is most likely to be useful in "loosely controlled" sessions where participants enter and leave without membership control or parameter negotiation.

Functions 1-3 SHOULD be used in all environments, but particularly in the IP multicast environment. RTP application designers SHOULD avoid mechanisms that can only work in unicast mode and will not scale to larger numbers. Transmission of RTCP MAY be controlled separately for senders and receivers for cases such as unidirectional links where feedback from receivers is not possible.

Protocol Structure

2	3	8	16bit
Version	P	RC	Packet type
Length			

- Version - Identifies the RTP version which is the same in RTCP packets as in RTP data packets. The version defined by this specification is two (2).
- P - Padding. When set, this RTCP packet contains some additional padding octets at the end which are not part of the control information. The last octet of the padding is a count of how many padding octets should be ignored. Padding may be needed by some encryption algorithms with fixed block sizes. In a compound RTCP packet, padding should only be required on the last individual packet because the compound packet is encrypted as a whole.
- RC- Reception report count, the number of reception report blocks contained in this packet. A value of zero is valid. Packet type Contains the constant 200 to identify this as an RTCP SR packet.
- Length - The length of this RTCP packet in 32-bit words minus one, including the header and any padding. (The offset of one makes zero a valid length and avoids a possible infinite loop in scanning a compound RTCP packet, while counting 32-bit words avoids a validity check for a multiple of 4.)

Related protocols

RTP, RTSP, UDP, TCP, IP

Sponsor Source

RTCP is defined by IETF (www.ietf.org) in RFC 3550.

Reference

http://www.javvin.com/protocol/rfc3550.pdf
RTP: A Transport Protocol for Real-Time Applications

Other Protocols

Protocol Name

COPS: Common Open Policy Service

Protocol Description

The Common Open Policy Service (COPS) protocol is a simple query and response protocol that can be used to exchange policy information between a policy server (Policy Decision Point or PDP) and its clients (Policy Enforcement Points or PEPs). One example of a policy client is an RSVP router that must exercise policy-based admission control over RSVP usage. At least one policy server exists in each controlled administrative domain. The COPS protocol has a simple but extensible design. The main characteristics of the COPS protocol include:

1. COPS employs a client/server model where the PEP sends requests, updates, and deletes to the remote PDP and the PDP returns decisions back to the PEP.
2. COPS uses TCP as its transport protocol for reliable exchange of messages between policy clients and a server.
3. COPS is extensible in that it is designed to leverage off self-identifying objects and can support diverse client specific information without requiring modifications to the COPS protocol itself. COPS was created for the general administration, configuration, and enforcement of policies.
4. COPS provides message level security for authentication, replay protection, and message integrity. COPS can also reuse existing protocols for security such as IPsec or TLS to authenticate and secure the channel between the PEP and the PDP.
5. COPS is stateful in two main aspects: (1) Request/Decision state is shared between client and server and (2) State from various events (Request/Decision pairs) may be inter-associated.
6. Additionally, COPS is stateful in that it allows the server to push configuration information to the client, and then allows the server to remove such state from the client when it is no longer applicable.

Protocol Structure

COPS common header:

4	8	16	32bit
Version	Flags	Op Code	Client-type
Message Length			

- Version - The version field specifies the COPS version number. The current version is 1.
- Flags - The defined flag values is 1 a Solicited Message Flag Bit. This flag is set when the message is solicited by another COPS message. (All other flags MUST be set to 0).
- Op Code - Code identifying the COPS operations: 1 Request (REQ); 2 Decision (DEC); 3 Report State (RPT); 4 Delete Request State (DRQ); 5 Synchronize State Req (SSQ); 6 Client-Open (OPN); 7 Client-Accept (CAT); 8 Client-Close (CC); 9 Keep-Alive (KA); 10 Synchronize Complete (SSC)
- Client-type - The Client-type identifies the policy client. Interpretation of all encapsulated objects is relative to the client-type.
- Message length - Size of message in octets, which includes the standard COPS header and all encapsulated objects.

Related protocols

TCP, RSVP

Sponsor Source

COPS is defined by IETF (www.ietf.org) in RFC 2748.

Reference

http://www.javvin.com/protocol/rfc2748.pdf

The COPS (Common Open Policy Service) Protocol

Protocol Name

SCTP: Stream Control Transmission Protocol

Protocol Description

Stream Control Transmission Protocol (SCTP) is designed to transport PSTN signalling messages (SS7/C7) over IP networks, but is capable of broader applications. SCTP is a reliable transport protocol operating on top of a connectionless packet network such as IP. SCTP is designed to address the limitations and complexity of TCP while transporting real time signaling and data such as PSTN signaling over an IP network. SCTP can also run on top of the UDP layer.

SCTP offers the following services:

- acknowledged error-free non-duplicated transfer of user data;
- data fragmentation to conform to discovered path MTU size;
- sequenced delivery of user messages within multiple streams, with an option for order-of-arrival delivery of individual user messages;
- optional bundling of multiple user messages into a single SCTP packet; and
- network-level fault tolerance through supporting of multi homing at either or both ends of an association.

The design of SCTP includes appropriate congestion avoidance behavior and resistance to flooding and masquerade attacks. The SCTP datagram is comprised of a common header and chunks. The chunks contain either control information or user data.

Protocol Structure

16	32bit
Source Port Number	Destination Port Number
Verification Tag	
Checksum	

- Source Port Number - SCTP sender's port number. It can be used by the receiver, in combination with the source IP Address, to identify the association to which this datagram belongs.
- Destination Port Number – Destination port number where SCTP datagram is intended to go. The receiving host will use this port number to de-multiplex the SCTP datagram to the correct receiving endpoint/application.
- Verification Tag - The receiver uses the Verification tag to identify the association. On transmit, the value of this Verification tag must be set to the value of the Initiate tag received from the peer endpoint during the association initialization.
- Checksum - This field contains an Adler-32 checksum on this SCTP datagram.

Related protocols

UDP, TCP, IP, SS7/C7

Sponsor Source

SCTP is defined by IETF (www.ietf.org) in RFC 2960.

Reference

http://www.javvin.com/protocol/rfc2960.pdf
Stream Control Transmission Protocol

Protocol Name

TRIP: Telephony Routing over IP

Protocol Description

Telephony Routing over IP (TRIP) is a policy driven inter-administrative domain protocol for advertising the reachability of telephony destinations between location servers and for advertising attributes of the routes to those destinations. TRIP's operation is independent of any signaling protocol; hence TRIP can serve as the telephony routing protocol for any signaling protocol.

The primary function of a TRIP speaker, called a location server (LS), is to exchange information with other LSs. This information includes the reachability of telephony destinations, the routes towards these destinations, and information about gateways towards those telephony destinations residing in the PSTN. LSs exchange sufficient routing information to construct a graph of ITAD connectivity so that routing loops may be prevented. In addition, TRIP can be used to exchange attributes necessary to enforce policies and to select routes based on path or gateway characteristics. This specification defines TRIP's transport and synchronization mechanisms, its finite state machine, and the TRIP data. This specification defines the basic attributes of TRIP. The TRIP attribute set is extendible, so additional attributes may be defined in future documents.

TRIP, used to distribute telephony routing information between telephony administrative domains, is modeled after the Border Gateway Protocol 4, which is used to distribute routing information between administrative domains. TRIP is enhanced with some link state features, as in the Open Shortest Path First (OSPF) protocol, IS-IS, and the Server Cache Synchronization Protocol (SCSP). TRIP uses BGP's inter-domain transport mechanism, BGP's peer communication, BGP's finite state machine, and similar formats and attributes to BGP. Unlike BGP however, TRIP permits generic intra-domain LS topologies, which simplifies configuration and increases scalability in contrast to BGP's full mesh requirement of internal BGP speakers. TRIP uses an intra-domain flooding mechanism similar to that used in OSPF, IS-IS, and SCSP.

TRIP runs over a reliable transport protocol. This eliminates the need to implement explicit fragmentation, retransmission, acknowledgment, and sequencing. The error notification mechanism used in TRIP assumes that the transport protocol supports a graceful close, i.e., that all outstanding data will be delivered before the connection is closed.

TRIP's operation is independent of any particular telephony signaling protocol. Therefore, TRIP can be used as the routing protocol for any of these protocols, e.g., H.323 and SIP.

The LS peering topology is independent of the physical topology of the network. In addition, the boundaries of an ITAD are independent of the boundaries of the layer 3 routing autonomous systems. Neither internal nor external TRIP peers need to be physically adjacent.

Protocol Structure

Each TRIP message has a fixed-size header. There may or may not be a data portion following the header, depending on the message type.

16	24bit
Length	Type

- Length - unsigned integer indicating the total length of the message, including the header, in octets. Thus, it allows one to locate, in the transport-level stream, the beginning of the next message. The value of the Length field must always be at least 3 and no greater than 4096, and may be further constrained depending on the message type. No padding of extra data after the message is allowed, so the Length field must have the smallest value possible given the rest of the message.
- Type - unsigned integer indicating the type code of the message. The following type codes are defined:
  1 - OPEN
  2 - UPDATE
  3 - NOTIFICATION
  4 - KEEPALIVE

Related protocols

BGP-4, H.323, SIP, SCSP

Sponsor Source

TRIP is defined by IETF (www.ietf.org) in RFC 3219.

Reference

http://www.javvin.com/protocol/rfc3219.pdf
Telephony Routing over IP (TRIP)

Wide Area Network and WAN Protocols

Description

A Wide Area Network (WAN) is a computer network covering multiple distance areas, which may spread across the entire world. WANs often connect multiple smaller networks, such as local area networks (LANs) or metro area networks (MANs). The world's most popular WAN is the Internet. Some segments of the Internet are also WANs in themselves. A wide area network may be privately owned or rented from a service provider, but the term usually connotes the inclusion of public (shared user) networks.

A virtual private network (VPN) riding on the public switched data network (PSDN) is often used by organizations for their private and secured communications. VPN uses encryption and other techniques to make it appear that the organisation has a dedicated network while making use of the shared infrastructure of the WAN.

WANs generally utilize different networking technologies and equipment than do LANs. Key technologies often found in WANs include SONET, Frame Relay, X.25, ATM, and PPP.

WAN technologies and protocols are mostly data link layer (layer 2) protocols which are defined by many organizations over time. The key organizations in this space are IETF for PPP, ITU-T for ATM, Frame Relay, ISO for X.25 and SONET.

Key Protocols

The key WAN protocols are listed as follows:

WAN	Wide Area Network
ATM	ATM: Asynchronous Transfer Mode Reference Model
	ATM Layer
	AAL: ATM Adaptation Layer Type 0-5 reserved for variable bit rate video transfer.
	LANE-NNI: LAN Emulation - Network to Network Interface
	LANE-UNI: LAN Emulation - User to Network Interface
	MPOA: Multi Protocol Over ATM
	PNNI: Private Network-to-Network Interface
	Q.2931: ATM Signalling User Interface
SONET/SDH	Synchronous Optical Network and Synchronous Digital Hierarchy
Broadband Access	DOCSIS: Data Over Cable Service Interface Specification
	BISDN: Broadband Integrade Service Digital Network
	ISDN: Integrated Services Digital Network
	Q.931: ISDN network layer interface protocol
	LAPD: ISDN Link Access Protocol Channel D (Q.921)

	xDSL: Digital Subscriber Line Technologies (DSL, IDSL, ADSL, HDSL, SDSL, VDSL, G.Lite)
PPP	PPP: Point-to-Point Protocols
	BAP: PPP Bandwidth Allocation Protocol
	BACP: PPP Bandwidth Allocation Control Protocol
	BCP: PPP Bridging Control Protocol
	CHAP: Challenge Handshake Authentication Protocol
	EAP: PPP Extensible Authentication Protocol
	LCP: PPP Link Control Protocol
	MultiPPP: MultiLink PPP (MP)
	NCP: PPP Network Control Protocol
	PAP: Password Authentication Protocol
	PPPoE: PPP over Ethernet
	PPPoA: PPP over ATM AAL5
X.25	HDLC: High Level Data Link Control protocol
	LAPB: Link Access Procedure Balanced for x.25
	X.25: ITU-T WAN communication protocol
Frame Relay	Frame Relay: WAN protocol for internetworking at layer 2
	LAPF: Link Access Procedure/Protocol (ITU Q.922)
Other	IBM SDLC: Synchronous Data Link Control protocol

Related protocols

LAN, MAN, TCP/IP

ATM Protocols

Protocol Name

ATM: Asynchronous Transfer Mode Reference Model

Protocol Description

The Asynchronous Transfer Mode (ATM) comprises a protocol suite under the ATM reference model which establishes a mechanism to carry all traffic on a stream of fixed 53-byte packets (cells). A fixed-size packet can ensure that the switching and multiplexing function could be carried out quickly and easily. ATM is a connection-oriented technology, i.e. two systems on the network should inform all intermediate switches about their service requirements and traffic parameters in order to establish communication.

The ATM reference model is divided into three layers: the ATM adaptation layer (AAL), the ATM layer, and the physical layer. The AAL interfaces the higher layer protocols to the ATM Layer, which relays ATM cells both from the upper layers to the ATM Layer and vice versa. When relaying information received from the higher layers, the AAL segments the data into ATM cells. When relaying information received from the ATM Layer, the AAL must reassemble the payloads into a format the higher layers can understand. This is called Segmentation and Reassembly (SAR). Different AALs are defined in supporting different types of traffic or service expected to be used on ATM networks.

The ATM layer is responsible for relaying cells from the AAL to the physical layer for transmission and from the physical layer to the AAL for use at the end systems. It determines where the incoming cells should be forwarded to, resets the corresponding connection identifiers and forwards the cells to the next link, buffers cells and handles various traffic management functions such as cell loss priority marking, congestion indication, and generic flow control access. It also monitors the transmission rate and conformance to the service contract (traffic policing).

The physical layer of ATM defines the bit timing and other characteristics for encoding and decoding the data into suitable electrical/optical waveforms for transmission and reception on the specific physical media used. In addition, it also provides a frame adaptation function, which includes cell delineation, header error check (HEC) generation and processing, performance monitoring, and payload rate matching of the different transport formats used at this layer. SONET, DS3, Fiber, and twisted-pair are a few of the media often used at the physical layer.

Protocol Structure

ATM Cell Format:

HEADER	GFC or VPI	VPI	
	VPI	VCI	
	VCI		
	VCI	PT (3 Bit)	CLP
	HEC		
IE	Cell Payload (48 Bytes)		

- Header — (5 Bytes) Generic flow control, VPI/VCI, and other control header.
- IE — (48 Bytes) Cell Payload.

Physical Layer Specification – Private UNI:

Frame Format	*Bit Rate/Line Rate*	*Media*
Cell Stream	25.6 Mbps/ 32 Mbaud	UTP-3
STS-1	51.84 Mbps	UTP-3
FDDI	100 Mbps/ 125 Mbaud	Multimode Fiber
STS-3c, STM-1	155.52 Mbps	UTP-5
STS-3c, STM-1	155.52 Mbps	Single-Mode Fiber, Multimode Fiber, Coax pair
Cell Stream	155.52 Mbps/ 194.4Mbaud	Multimode Fiber, STP
STS-3c, STM-1	155.52 Mbps	UTP-3
STS-12, STM-4	622.08 Mbps	SMF, MMF

Physical Layer Specification – Public UNI:

Frame Format	*Bit Rate/Line Rate*	*Media*
DS1	1.544 Mbp	Twisted pair
DS3	44.736 Mbps	Coax pair
STS-3c, STM-1	155.520 Mbps	Single-mode Fiber
E1	2.048 Mbps	Twisted pair, Coax pair
E3	34.368 Mbps	Coax pair
J2	6.312 Mbps	Coax pair
N × T1	N × 1.544 Mbps	Twisted pair

Related protocols

SONET, AAL0-AAL5, LAN Emulation, CES, PNNI and MPOA, Q.2931

Sponsor Source

The ATM protocol reference model is based on standards developed by the ITU.
http://www.atmforum.com/standards/approved.html
ATM Forum approved specifications

Reference

ITU-T Recommendation I.363, “B-ISDN ATM Adaptation Layer (AAL) Specification”
http://www.atmforum.com/standards/approved.html=uni
ATM User-Network Interface Specification
http://www.cisco.com/univercd/cc/td/doc/cisintwk/ito_doc/atm.htm ATM Overview

Protocol Name

ATM Layer: Asynchronous Transfer Mode Layer

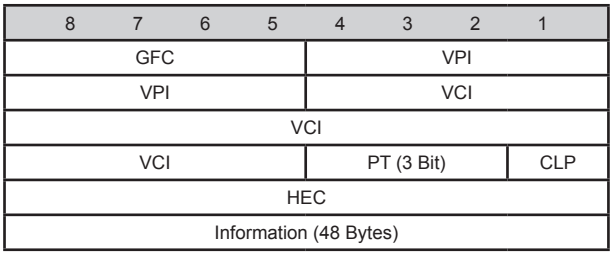
Protocol Description

The ATM layer, the layer 2 in the ATM reference model, provides an interface between the ATM adaption layer (AAL) and the physical layer. This layer is responsible for relaying cells from the AAL to the physical layer, such as SONET, for transmission and from the physical layer to the AAL for use at the end systems. It determines where the incoming cells should be forwarded to, resets the corresponding connection identifiers and forwards the cells to the next link, buffers incoming and outgoing cells and handles various traffic management functions, such as cell loss priority marking, congestion indication and generic flow control access. It also monitors the transmission rate and conformance to the service contract (traffic policing).

The fields in the ATM header define the functionality of the ATM layer. The format of the header for ATM cells has two different forms, one for use at the user-to-network interface (UNI) and the other for use internal to the network, the network-to-node interface (NNI).

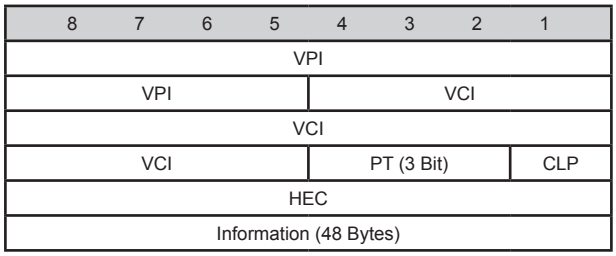
Protocol Structure

UNI Structure:



- GFC— generic flow control, used to limit the amount of data entering the network during periods of congestion.
- VPI — Virtual Path Identifier.
- VCI— Virtual Channel Identifier. The VPI and the virtual channel identifier (VCI) together form the routing field, also called VPCI, which associates each cell with a particular channel or circuit. The VCI is a single-channel identifier. The VPI allows grouping of VCs with different VCIs and allows the group to be switched together as an entity. However, the VPIs and VCIs have significance only on the local link; the contents of the routing field will generally change as the cell traverses from link to link. These fields, in UNI, can support up to 16 million desk (user) to network sessions.
- PT —Payload Type.
- CLP—Cell Loss Priority.
- HEC—Header Error Control.

NNI Structure:



- VPI — Virtual Path Identifier.
- VCI— Virtual Channel Identifier. See above for VPCI detail definition. These fields, which allows 268 millions NNI sessions, represent the network-to-node routing information within the ATM cell.
- PT —Payload Type.
- CLP—Cell Loss Priority.
- HEC—Header Error Control.

Related protocols

ATM, SONET, AAL0-AAL5, LAN Emulation (LANE), CES, UNI, NNI and Q.2931

Sponsor Source

The ATM protocols is based on standards developed by the ITU.
http://www.atmforum.com/standards/approved.html
ATM Forum approved specifications

Reference

http://www.atmforum.com/standards/approved.html
ATM Forum approved specifications
http://www.cisco.com/univercd/cc/td/doc/cisintwk/ito_doc/atm.htm
ATM Overview

Protocol Name

AAL: ATM Adaptation Layer (AAL0, AAL2, AAL3/4, AAL5)

Protocol Description

The ATM Adaptation Layer (AAL) relays the ATM cells between ATM Layer and higher layers. When relaying information received from the higher layers, it segments the data into ATM cells. When relaying information received from the ATM Layer, it must reassemble the payloads into a format the higher layers can understand. This operation, which is called Segmentation and Reassembly (SAR), is the main task of AAL. Different AALs were defined in supporting different traffic or services expected to be used. The service classes and the corresponding types of AALs are as follows:

Class A - Constant Bit Rate (CBR) service: AAL1 supports a connection-oriented service in which the bit rate is constant. Examples of this service include 64 Kbit/sec voice, fixed-rate uncompressed video and leased lines for private data networks.

Class B - Variable Bit Rate (VBR) service: AAL2 supports a connection-oriented service in which the bit rate is variable but requires a bounded delay for delivery. Examples of this service include compressed packetized voice or video. The bounded delay for delivery is necessary for the receiver to reconstruct the original uncompressed voice or video.

Class C - Connection-oriented data service: For connection-oriented file transfer and in general, data network applications where a connection is set up before data is transferred. This type of service has variable bit rate and does not require bounded delay for delivery. Two AAL protocols were defined to support this service class, and have been merged into a single type, called AAL3/4. However, with its high complexity, the AAL5 protocol is often used to support this class of service.

Class D - Connectionless data service: Examples of this service include datagram traffic and in general, data network applications where no connection is set up before data is transferred. Either AAL3/4 or AAL5 can be used to support this class of service.

Operation Administration and Maintenance (OA&M) - OA&M is defined for supervision, testing, and performance monitoring. It uses loop-back for maintenance and ITU TS standard CMIP, with organization into 5 hierarchical levels: Virtual Channel (F5 - Between VC endpoints), Virtual Path (F4- Between VP endpoints), Transmission Path (F3- Between elements that perform assembling, and disassembling of payload, header, or control), Digital Section (F2 Between section end-points, performs frame synchronization) and Regenerator Section (F1- Between regeneration sections).

Protocol Structure

AAL0 PDU:

AAL0 payload, also referred as raw cells, consists of 48 bytes without special fields.

AAL1 PDU:

<table>
<tr><td>1</td><td>3 bits</td><td>3 bits</td><td>1</td><td>47 Bytes</td></tr>
<tr><td colspan="2">SN</td><td colspan="2">SNP</td><td>SAR</td></tr>
<tr><td>CSI</td><td>SC</td><td>CRC</td><td>P</td><td>Payload</td></tr>
</table>

- SN - Sequence number. Numbers the stream of SAR PDUs of a CPCS PDU (modulo 16).
- CSI - Convergence sublayer indicator. Used for residual time stamp for clocking.
- SC - Sequence count.
- NP - Sequence number protection.
- CRC - Cyclic redundancy check calculated over the SAR header.
- P - Parity calculated over the CRC.
- SAR PDU payload - 47-byte user information field.

AAL2 PDU

AAL2, with compression, silent and idle channel suppression, is perfect for low-rate voice traffic. AAL type 2 is subdivided into the Common Part Sublayer (CPS) and the Service Specific Convergence Sublayer (SSCS).

AAL2 CPS Packet

The CPS packet consists of a 3 octet header followed by a payload. The structure of the AAL2 CPS packet is shown in the following illustration.

<table>
<tr><td>8bits</td><td>6bits</td><td>5bits</td><td>5bits</td><td>1-45/64 bytes</td></tr>
<tr><td>CID</td><td>LI</td><td>UUI</td><td>HEC</td><td>Information payload</td></tr>
<tr><td colspan="5">AAL2 CPS packet</td></tr>
</table>

- CID - Channel identification.
- LI - Length indicator: the length of the packet payload associated with each individual user. Value is one less than the packet payload and has a default value of 45 bytes (may be set to 64 bytes).
- UUI - User-to-user indication. Provides a link between the CPS and an appropriate SSCS that satisfies the higher layer application.
- HEC - Header error control.
- Information payload - Contains the CPS/SSCS PDU.

AAL2 CPS PDU

The structure of the AAL2 CPS PDU is shown as follows:

6bits	1bit	1bit	0-47 bytes	0-47 bytes
OSF	SN	P	AAL2 PDU payload	PAD
AAL2 CPS PDU				

- OSF - Offset field. Identifies the location of the start of the next CPS packet within the CPS-PDU.
- SN - Sequence number. Protects data integrity.
- P - Parity. Protects the start field from errors.
- SAR PDU payload - Information field of the SAR PDU.
- PAD - Padding.

AAL2 SSCS Packet

The SSCS conveys narrowband calls consisting of voice, voice-band data or circuit mode data. SSCS packets are transported as CPS packets over AAL2 connections. The CPS packet contains a SSCS payload. There are 3 SSCS packet types: Type 1 Unprotected (This is used by default); Type 2 Partially protected; and Type 3 Fully protected (The entire payload is protected by a 10-bit CRC which is computed as for OAM cells. The remaining 2 bits of the 2-octet trailer consist of the message type field).

AAL2 SSCS Type 3 Packets:

The AAL2 type 3 packets are used for Dialed digits, Channel associated signalling bits, Facsimile demodulated control data, Alarms and User state control operations. The general structure of AAL2 SSCS Type 3 PDUs is shown as follows. The format varies according to the actual message type.

2bits	14bits	16bits	6bits	10bits
Redun-dancy	Time stamp	Message dependant information	Message type	CRC-10
AAL2 SSCS Type 3 PDU – General Structure				

- Redundancy - Packets are sent 3 times to ensure error correction. The value in this field signifies the transmission number.
- Time stamp - Counters packet delay variation and allows a receiver to accurately reproduce the relative timing of successive events separated by a short interval.
- Message dependant information - Packet content that varies, depending on the message type.
- Message type - The message type code.
- CRC-10 - The 10-bit CRC.

AAL3/4 PDU:

2	4	10 bits	44 Bytes	6 bits	10 bits
ST	SN	MID	PDU Payload	LI	CRC

- ST - Segment Type: BOM (Beginning of Message), COM (Continuation of Message), EOM (End of Message), SSM (Single Segment Message).
- SN - Sequence number. Numbers the stream of SAR PDUs of a CPCS PDU (modulo 16).
- MID - Multiplexing Indication
- PDU payload - 44-byte user information field.
- LI - Length indicator.
- CRC - Cyclic redundancy check calculated over the SAR header.

AAL3/4 CS PDU:

1	1	2 bits	40 Bytes	1	1	2 bits
CPI	BTag	BAsize	PDU Payload + PAD	AL	ETag	LEN

- CPI - Common Part Indication
- BTag - Beginning Tag
- BAsize - Buffer Allocation Size
- PDU payload - Variable length user information field up to 40 Bytes
- PAD - Padding (up to 3 bytes) used to cell align the trailer.
- AL - Alignment. A filling byte coded with zero
- ETag - End Tag.
- LEN - Length of Information Field

AAL5 CS PDU:

0-48 Bytes	0-47	1	1	2	4 Bytes
PDU payload	PAD	UU	CPI	LI	CRC-32

AAL5 is the simple and efficient AAL (SEAL) most used for data traffic. It has no per-cell length or per-cell CRC fields.

- PDU payload - Variable length user information field
- PAD - Padding used to cell align the trailer which may be between 0 and 47 bytes long.
- UU - CPCS user-to-user indication to transfer one byte of user information
- CPI - Common Part Indication
- LI - Length indicator.

For OA&M cells, there are pre-defined (reserved) VPI/VCI numbers:

- 0/0 Unassigned or Idle 0/1 Meta-signaling
- 0/3 Segment F4 Flow 0/4 End-to-end F4 flow
- 0/5 Signaling 0/15 SMDS
- 0/16 Interim Layer Management Interface (ILMI)

F4/F5 OA&M PDU format:

4 bits	4 bits	45 Bytes	6 bits	10 bits
OAM Type	Function Type	Function Spec	Re-serve	CRC-10

- OAM type / Function type- The possible values for OAM type and function type are defined for Fault, Performance, Activation/Deactivation
- CRC-10 - Cyclic redundancy check calculated over the SAR header. $G(x) = x^{10}+x^9+x^5+x^4+x+1$

Related protocols

SONET, AAL0-AAL5, LAN Emulation, CES, PNNI and MPOA, Q.2931

Sponsor Source

ATM Adaptation Layers are defined by ITU in document I.366.2.

Reference

http://www.atmforum.com/standards/approved.html
ATM Forum approved specifications
http://www.atmforum.com/standards/approved.html#uni
ATM User-Network Interface Specification
http://www.cisco.com/univercd/cc/td/doc/cisintwk/ito_doc/atm.htm
ATM Overview

Protocol Name

ATM UNI: ATM Signaling User-to-Network Interface

Protocol Description

Signalling is the process by which ATM users and the network exchange the control of information, request the use of network resources, or negotiate for the use of circuit parameters. The VPI/VCI pair and requested bandwidth are allocated as a result of a successful signalling exchange. These messages are sent over the Signalling ATM Adaptation Layer (SAAL), which ensures their reliable delivery. The SAAL is divided into a Service Specific Part and a Common Part. The Service Specific Part is further divided into a Service Specific Coordination Function (SSCF), which interfaces with the SSCF user; and a Service Specific Connection-Oriented Protocol (SSCOP), which assures reliable delivery.

The UNI signalling protocols within the SAAL are responsible for ATM call and connection control, including call establishment, call clearing, status enquiry, and point-to-multipoint control. ATM UNI signalling message uses the Q.931 message format, which is made up of a message header and a variable number of Information Elements.

The VPI/VCI pair and requested bandwidth are allocated as a result of a successful signalling exchange. Two levels of virtual connections can be supported at the UNI:

1) A point-to-point or point-to-multipoint Virtual Channel Connection (VCC) which consists of a single connection established between two ATM VCC end-points; and 2) A point-to-point or point-to-multipoint Virtual Path Connection (VPC) which consists of a bundle of VCCs carried transparently between two ATM VPC end-points. Note: For VPC at the Public UNI, traffic monitoring and throughput enforcement will be performed across all cells carried on the same VPI independently of the VCI values.

Protocol Structure

SAAL protocol stacks illustrated below support UNI connection control signaling:

	User-Network Signalling
SAAL	UNI SSCF
	SSCOP
	AAL Type 5 Common Part
	ATM Layer
	Physical Layer

UNI Signaling Message:

8	7	6	5	4	3	2	1	bit/Octet
Protocol discriminator								1
0	0	0	0	Length of call reference value				2
Flag	Call reference value							3
Call reference value (continued)								4
								5
Message type								6
Message type (continued)								7
Message length								8
Message length (continued)								9
Variable length Information Elements as required								etc.

- Protocol discriminator -Distinguishes Messages for user-network call control from other messages. (9 for Q.2931 messages)
- Call reference - Unique number for every ATM connection which serves to link all signalling messages relating to the same connection. It is comprised of the call reference value and the call reference flag. The call reference flag indicates who allocated the call reference value.
- Message type - The message may be of the following types:
    1. Call establishment messages: such as CALL PROCEEDING, sent by the called user to the network or by the network to the calling user to indicate initiation of the requested call. CONNECT, sent by the called user to the network and by the network to the calling user to indicate that the called user accepted the call. CONNECT ACKNOWLEDGE, sent by the network to the called user to indicate that the call was awarded and by the calling user to the network; and SETUP, sent by the calling user to the network and by the network to the calling user to initiate a call.
    2. Call clearing messages: such as RELEASE, sent by the user to request that the network clear the connection or sent by the network to indicate that the connection has cleared. RELEASE COMPLETE, sent by either the user or the network to indicate that the originator has released the call reference and virtual channel. RESTART, sent by the user or the network to restart the indicated virtual channel. RESTART ACKNOWLEDGE, sent to acknowledge the receipt of the RESTART message.
    3. Miscellaneous messages: such as STATUS, sent by the user or network in response to a STATUS ENQUIRY message. STATUS EN-

QUIRY, sent by the user or the network to solicit a STATUS message.

4. Point-to-Multipoint messages: such as ADD PARTY, which adds a party to an existing connection. ADD PARTY ACKNOWLEDGE, which acknowledges a successful ADD PARTY. ADD PARTY REJECT, which indicates an unsuccessful ADD PARTY. DROP PARTY, which drops a party from an existing point-to-multipoint connection. DROP PARTY ACKNOWLEDGE, which acknowledges a successful DROP PARTY.

- Message length - The length of the contents of a message.
- Information Elements - There are several types of information elements. Some may appear only once in the message; others may appear more than once. Depending on the message type, some information elements are mandatory and some are optional. The order of the information elements does not matter to the signalling protocol. The information elements in UNI 3.0 are listed in the following table:

IE	Description	Max. No.
Cause	Gives the reason for certain messages. For example, the Cause IE is part of the release message, indicating why the call was released.	2
Call state	Indicates the current state of the call.	1
Endpoint reference	Identifies individual endpoints in a point-to-multipoint call.	1
Endpoint state	Indicates the state of an endpoint in a point-to-multipoint call.	1
AAL parameters	Includes requested AAL type and other AAL parameters.	1
ATM user cell rate	Specifies traffic parameters.	1
Connection identifier	Identifies the ATM connection and gives the VPI and VCI values.	1
Quality of Service parameter	Indicates the required Quality of Service class for the connection.	1
Broadband high-layer information	Gives information about the high-layer protocols for compatibility purposes.	1
Broadband bearer capacity	Requests a service from the network (such as CBR or VBR link, point-to-point and point-to-multipoint link).	1
Broadband low-layer information	Checks compatibility with layer 2 and 3 protocols.	3
Broadband locking shift	Indicates a new active codeset.	-
Broadband non-locking shift	Indicates a temporary codeset shift.	-
Broadband sending complete	Indicates the competition of sending the called party number.	1
Broadband repeat indicator	Indicates how IEs which are repeated in the message should be handled.	1
Calling party number	Origin of the call.	1
Calling party subaddress	Subaddress of calling party.	1
Called party number	Destination of the call.	1
Called party subaddress	Subaddress of the called party.	1
Transit network selection	Identifies one requested transit network.	1
Restart indicator	Identifies which facilities should be restarted (e.g., one VC, all VCs).	1

Related protocols

ATM, AAL0-AAL5, LAN Emulation (LANE), CES, PNNI, MPOA, NNI and Q.2931.

Sponsor Source

The ATM protocols are based on standards developed by the ITU.
http://www.atmforum.com/standards/approved.html
ATM Forum approved specifications

Reference

http://www.atmforum.com/standards/approved.html
ATM Forum approved specifications
http://www.cisco.com/univercd/cc/td/doc/cisintwk/ito_doc/atm.htm
ATM Overview

Protocol Name

LANE NNI: ATM LAN Emulation NNI

Protocol Description

The ATM LAN Emulation (LANE) enables the implementation of emulated LANs over an ATM network. An emulated LAN provides communication of user data frames among all its users, similar to a physical LAN. One or more emulated LANs could run on the same ATM network. However, each of the emulated LANs is logically independent of the others. Communication between emulated LANs requires some type of interconnection device (bridge, router, etc.), even though direct ATM connections between emulated LANs are explicitly allowed in some circumstances. The LAN Emulation LUNI defines the protocols and interactions between LAN Emulation Clients (LE Clients) and the LAN Emulation Service. Each LE Client connects across the LUNI to a single LES and BUS, may connect to a single LECS, and may have connections to multiple SMSs.

The LAN Emulation NNI (LNNI) defines the behavior of these LANE service components as seen by each other, the procedures necessary to provide a distributed and reliable LAN Emulation Service. A single ELAN may be served by multiple LECSs, LESs, BUSs and SMSs. Each LES, BUS, and SMS serves a single ELAN, while an LECS may serve multiple ELANs. LANE service components interconnect with multiple VCCs for Configuration, Status, Database Synchronization, Control and Data forwarding. The LNNI specification provides multivendor interoperability among the components serving an ELAN so that consumers may mix and match the LANE Service implementations of different vendors.

LANE service consists of four major components:

- LAN emulation client (LEC) - located in ATM end systems, implements the LUNI interface, serves as a proxy for LAN systems to perform data forwarding and address resolution, provides a MAC level emulated Ethernet/IEEE 802.3 or IEEE 802.5 service interface to higher level software.
- LAN emulation server (LES) - supports the address resolution protocol (LE-ARP), and is used to determine the ATM address of the target LEC responsible for a certain destination MAC address. An LE Client is connected to only one LE Server. An LE Client may register LAN Destinations it represents and/or multicast MAC addresses it wishes to receive with its LE Server. An LE Client will also query its LE Server to resolve a MAC address or route descriptor of an ATM address.
- Broadcast/Unknown Server (BUS) - handles all multicast traffic forwarding to all attached LECs. An LE Client sees a single Broadcast and Unknown Server.
- Selective Multicast Server (SMS) - may be used to offload much of the multicast processing from the BUSs, which also have to forward broadcast frames and frames for unresolved LAN destinations, to efficiently forwaqrd multicast frames.

The multiple LANE Service entities serving an ELAN need to cooperate and communicate in order to provide a distributed and reliable LAN Emulation Service. The communications required for LNNI may be partitioned as follows:

a) Control Plane

- Configuration and Status Communications - LESs and SMSs obtain configuration information from an LECS over Configuration Direct VCCs. LECSs obtain the status of LESs and SMSs over the same connection.
- LANE Control Communications - Each LES is responsible for distributing LE_ARP requests for unregistered destinations from local LE Clients to local LE Clients and to other LESs. LESs must also forward LE_ARP responses back to the originator. Additionally, LESs must be able to forward LE_FLUSH responses and LE_TOPOLOGY requests to the correct destination(s).b) Synchronization Plane
- LECS Synchronization - A particular LECS may not directly receive status from all service components. Thus, LECSs must exchange LES and SMS status information among themselves. In order to distribute this status information, all LECSs participating in an ELAN must maintains an LECS Synchronization VCC to all other LECSs in the network.
- LES-SMS Database Synchronization – LESs and SMSs use SCSP to synchronize their databases.

c) Data Plane

- BUS Data Communications - Each BUS is assumed to be logically paired with an LES, and the BUS is assumed to have access to the registration database maintained by the LES, which includes the ATM address of all BUSs. No protocol is defined between a paired LES and BUS.
- SMS Data Communications – Every SMS (and LES) obtains a complete copy of the registration database for the entire ELAN via SCSP, so every SMS knows of every other SMS and BUS. When an LE Client LE_ARPs for a multicast address, the LES should assign the client to an SMS as a sender if an SMS is available for that destination, otherwise a BUS's ATM address is returned in the LE_ARP response. An ELAN, and hence all the ELAN's SMSs, may operate in either distributed or stand-alone mode, as determined by the network administrator.

Protocol Structure

LANE Data Frames:

The LNNI Control frame format is shown below:

	0	LLC = X"AAAA03"			OUI
	4	OUI	Frame Type		
	8	ELAN-ID			
LANE Control Frame	12	REQUESTER-LECID		FLAGS	
	16	SOURCE-LAN-DESTINATION			
	24	TARGET-LAN-DESTINATION			
	32	SOURCE-ATM-ADDRESS			
	52	LAN-Type	MAX-Frame-Size	Number-TLVS	ELAN-Name-Size
	56	TARGET-ATM-ADDRESS			
	76	ELAN-NAME			
	108	TLVs BEGIN			

- LLC - Logical Link Control: The Control Coordinate VCCs are all LLC encapsulated.
- OUI - Organizationally Unique Identifier = X"00A03E" which indicates ATM Forum.
- FRAME-TYPE = X"000F"
- ELAN-ID - Emulated LAN ID
- OP-CODE (2 Bytes) - Control frame Operation type. Some defined OP-Code are:

OP-CODE Value	OP-CODE Function
X"000b"	LNNI_CONFIGURE_TRIGGER
X"000C"	LNNI_LECS_SYNC_REQUEST
X"000d"	LNNI_KEEP_ALIVE_REQUEST
X"000d"	LNNI_KEEP_ALIVE_RESPONSE
X"000e"	LNNI_VALIDATE_REQUEST
X"000e"	LNNI_VALIDATE_RESPONSE

- Status - (2 Bytes) Control frame Operation status.
- TLV - Type/ Length / Value Encoded Parameter, Examples of LNNI TLVs are:

Item	Type	LEN	Description
ServerId	00-A0-3E-14	2	Unique identifier for a server within an ELAN
ServerGroupId	00-A0-3E-15	2	Uniquely correlates to an ELAN-ID. Required for SCSP.
SynchronizationPeerServer	00-A0-3E-16	20	Multiplexed ATM Address of ES or SMS to synchronize DB using SCSP.
SmsModeOfOperation	00-A0-3E-19	1	Indicates SMS operational mode. 0 = STAND_ALONE 1 = DISTRIBUTED

Related protocols

ATM, SONET, AAL0-AAL5, LAN Emulation (LANE), CES, UNI, NNI and Q.2931.

Sponsor Source

The ATM protocols are based on standards developed by the ITU.
http://www.atmforum.com/standards/approved.html
ATM Forum approved specifications

Reference

http://www.atmforum.com/standards/approved.html
ATM Forum approved specifications
http://www.cisco.com/univercd/cc/td/doc/cisintwk/ito_doc/atm.htm
ATM Overview

Protocol Name

LANE UNI: ATM LAN Emulation UNI

Protocol Description

The ATM LAN emulation (LANE) specification defines how an ATM network can emulate a sufficient set of the medium access control (MAC) services of existing LAN technologies such as Ethernet and Token Ring, so that higher layer protocols can be used without modification. An emulated LAN (ELAN), which provides the appearance of either an Ethernet or Token-Ring LAN segment over a switched ATM network, is composed of a collection of LE Clients and a set of co-operating service entities: LAN Emulation Configuration Servers (LECSs), LAN Emulation Servers (LESs), Broadcast and Unknown Servers (BUSs), and Selective Multicast Servers (SMSs).

The LAN Emulation LUNI defines the protocols and interactions between LAN Emulation Clients (LE Clients) and the LAN Emulation Service, including initialization, registration, address resolution, and data transfer procedures. Each LE Client connects across the LUNI to a single LES and BUS, may connect to a single LECS, and may have connections to multiple SMSs.

Communication among LE Clients and between LE Clients and the LE Service is performed over ATM virtual channel connections (VCCs). Each LE Client must communicate with the LE Service over control and data VCCs. LANE assumes the availability of point-to-point and point-to-multipoint Switched Virtual Circuits (SVCs). Multicast Forward and Control Distribute flows are carried on point-to-multipoint VCCs. Data Direct, Control Direct, Configure Direct, Default Multicast Send and Selective Multicast Send flows are carried on point-to-point VCCs. Only Data Direct flows may be LLC-multiplexed. All other flows are non-multiplexed.

LAN Emulation encompasses both Ethernet and Token Ring emulation. In Ethernet emulation, a LAN Emulation component need examine only a data frame's destination MAC address in order to direct the frame towards its ultimate destinations. In Token Ring emulation, however, a LAN emulation component may have to use a "Route Descriptor" extracted from the data frame's Routing Information Field (RIF) in order to properly direct the frame over the Emulated LAN.

Most LAN emulation services would be implemented as device drivers below the network layer in ATM-to-legacy LAN bridges and ATM end systems. In LANE, the bandwidth management capability is currently supported by the "available bit rate" (ABR) service.

Protocol Structure

LE Data Frames:

1) For 802.3 (Ethernet) Frame – Non-multiplexed data frame:

0	LE Header	Destination Address
4	Destination Address	
8	Source Address	
12	Source Address	Type / Length
16 and on	User Info	

2) For 802.5 (Token Ring) Frame– Non-multiplexed data frame:

0	LE Header	AC PAD	FC
4	Destination Address		
8	Destination Address	Source Address	
12	Source Address	Type / Length	
16-46	Routing Information Field		
	User Info		

- LE Header— LAN Emulation header which contains either the LAN Emulation client identifier value, the sending client, or X'0000'.

LE Control Frame:

Except for LLC multiplexed Data with Direct VCCs, all LAN Emulation control frames, such as LE_FLUSH_REQUESTs, READY_IND and READY_QUERY, use the format described below:

0	MARKER = X"FF00"		PROTOCOL = X"01"	VERSION = X"01"
4	OP-CODE		STATUS	
8	TRANSACTION-ID			
12	REQUESTER-LECID		FLAGS	
16	SOURCE-LAN-DESTINATION			
24	TARGET-LAN-DESTINATION			
32	SOURCE-ATM-ADDRESS			
52	LAN-Type	MAX-Frame-Size	Number-TLVS	ELAN-Name-Size
56	TARGET-ATM-ADDRESS			
76	ELAN-NAME			
108	TLVs BEGIN			

- OP-CODE – (2 Bytes) Control frame Operation type. Some defined OP-Code are:

OP-CODE Value	OP-CODE Function
X"0001" & X"0101"	LE_CONFIGURE_REQUEST & LE_CONFIGURE_RESPONSE
X"0002" & X"0102"	LE_JOIN_REQUEST & LE_JOIN_RESPONSE
X"0003" & X"0103"	READY_QUERY & READY_IND

X”0004” & X”0104”	LE_REGISTER_REQUEST & LE_REGISTER_RESPONSE
X”0005” & X”0105”	LE_UNREGISTER_REQUEST & LE_UNREGISTER_RESPONSE
X”0006” & X”0106”	LE_ARP_REQUEST & LE_ARP_RESPONSE
X”0007” & X”0107”	LE_FLUSH_REQUEST LE_FLUSH_RESPONSE
X”0008” & X”0108”	LE_NARP_REQUEST & Undefined
X”0009” & X”0109”	LE_TOPOLOGY_REQUEST & Undefined
X”000A” & X”010A”	LE_VERIFY_REQUEST & LE_VERIFY_RESPONSE

- Status – (2 Bytes) Control frame Operation status. Some defined Status codes are:

Code (dec)	Name	Code (dec)	Name
0	Success	1	Version Not Supported
2	Invalid request parameters	4	Duplicate LAN Destination registration
5	Duplicate ATM address	6	Insufficient resources to grant request
7	Access denied	8	Invalid REQUESTOR-ID
9	Invalid LAN Destination	10	Invalid ATM Address
20	No Configuration	21	LE_CONFIGURE Error
22	Insufficient Information	24	TLV Not Found

- TLV - Type / Length / Value Encoded Parameter.

LANE LLC-multiplexed Frame - has a 12-octet LLC multiplexing header:

0	LLC- X“AA”	LLC- X“AA”	LLC X“03”	OUI-X“00”
4	OUI-X“A0”	OUI-X“3E”	Frame-Type	
8	ELAN-ID			
12-28/ 58	LANE Data Frame Header (802.3 or 802.5)			
	User Info			

LLC field is three octets, containing the constant value X“AAAA03”, indicating that an OUI follows.

OUI field is three octets, containing the constant value X“00A03E”, indicating “ATM Forum”.

The next two octets are a FRAME-TYPE field containing the value X“000C” for IEEE 802.3 data frame, X“000D” for IEEE 802.5 data frame, X“000E” for for LANE LLC-multiplexed READY_IND and READY_QUERY control frames.

The ELAN-ID field identifies the emulated LAN for this data frame.

Related protocols

ATM, SONET, AAL0-AAL5, LAN Emulation (LANE), CES, UNI, NNI and Q.2931.

Sponsor Source

The ATM protocols is based on standards developed by the ITU.
http://www.atmforum.com/standards/approved.html
ATM Forum approved specifications

Reference

http://www.atmforum.com/standards/approved.html
ATM Forum approved specifications
http://www.cisco.com/univercd/cc/td/doc/cisintwk/ito_doc/atm.htm
ATM Overview

Protocol Name

MPOA: Multi-Protocol Over ATM

Protocol Description

The Multi Protocol Over ATM (MPOA) deals with the efficient transfer of inter-subnet unicast data in a LANE environment. MPOA integrates LANE and NHRP to preserve the benefits of LAN Emulation, while allowing inter-subnet, internetwork layer protocol communication over ATM VCCs without requiring routers in the data path. MPOA provides a framework for effectively synthesizing bridging and routing with ATM in an environment of diverse protocols, network technologies, and IEEE 802.1 virtual LANs. MPOA is capable of using both routing and bridging information to locate the optimal exit from the ATM cloud. It allows the physical separation of internetwork layer route calculation and forwarding, a technique known as virtual routing.

Based on ATM UNI signaling, LAN Emulation, and Next Hop Resolution Protocol (NHRP), MPOA defines two components: MPOA Clients (MPCs) and MPOA Servers (MPSs), and the protocols that are required to communicate and receive services.

The MPS is a component of a router, and is only useful in a router that has a Next Hop Server (NHS) and interfaces to one or more LECs. The data and control path from the router through the LEC(s) to LANE is unaltered by MPOA. The MPS does, however, interact with the router, its LEC(s), the NHS, and other MPOA components. A LEC is associated with a single MPS.

MPOA uses a protocol based on the Next Hop Resolution Protocol [NHRP] to manage caches and establish shortcuts. It performs the following operations:

- Configuration- Obtaining the appropriate configuration information.
- Discovery- MPCs and MPSs learning of each others' existence.
- Target Resolution - Determining the mapping of a Target to an egress ATM address, an optional Tag, and a set of parameters used to set up a Shortcut VCC to forward packets across subnet boundaries.
- Connection Management - VCCs creating, maintaining, and terminating for the purpose of transferring control information and data.
- Data Transfer - Forwarding internetwork layer data across a Shortcut.

MPOA components must support the use of LLC/SNAP encapsulation for all PDUs. By default VCCs must be signaled to use LLC encapsulation. An MPOA component must be capable of establishing, receiving and maintaining a VCC to any entity that conforms to the connection management procedures, whether or not that entity is an MPOA component

Protocol Structure

MPOA tagged encapsulation format:

0	LLC- X"AA"	LLC- X"AA"	LLC X"03"	OUI-X"00"
4	OUI-X"00"	OUI-X"00"	Frame-Type = 0x884C	
8	MPOA Tag			
12-n	Internetwork Layer PDU (up to 2^16 - 13 octets)			

MPOA Control Frame – MPOA tagged encapsulation format:

0	LLC- X"AA"	LLC- X"AA"	LLC X"03"	OUI-X"00"
4	OUI-X"00"	OUI-X"5E"	Frame-Type = 0x0003	
8-n	MPOA PDU (up to 2^16 - 9 octets)			

By default, MPOA uses LLC encapsulation for all control flows as defined in [NHRP], with the same fixed header as an NHRP packet described below:

0	ar$afn		ar$pro.type	
4	ar$pro.snap			
8	ar$pro.snap	ar$hopcnt	ar$pkstz	
12	ar$chksum		ar$extoff	
16	ar$op.version	ar$op.type	ar$shtl	ar$sstl

- ar$afn - Defines the type of "link layer" address being carried.
- ar$pro.type – Protocol Type. This field is a 16 bit unsigned integer.
- ar$pro.snap - When ar$pro.type field equals to 0x0080, a snap encoded extension, which is placed in the ar$pro.snap field. is used to encode the protocol type. By default this field should be set to 0.
- ar$hopcnt - Hop count- the maximum number of NHSs that an MPOA packet is allowed to traverse.
- ar$pktsz - The total length of the MPOA packet in octets.
- ar$chksum - The standard IP 16-bit checksum over the entire MPOA packet.
- ar$extoff - This field identifies the existence and location of MPOA extensions.
- ar$op.version - Version of generic address mapping and management protocol, set to X"01" NHRP
- ar$op.type - The MPOA packet type. Some values for packet types are:

128	MPOA Cache Imposition Request.	129	MPOA Cache Imposition Reply.
130	MPOA Egress Cache Purge Request.	131	MPOA Egress Cache Purge Reply.
132	MPOA Keep-Alive.	133	MPOA Trigger.
134	MPOA Resolution Request.	135	MPOA Resolution Reply.

136	MPOA Error Indicator		

- ar$shtl - The type and length of the source NBMA address interpreted.
- ar$sstl - The type and length of the source NBMA subaddress interpreted.

Related protocols

ATM, SONET, AAL0-AAL5, LAN Emulation (LANE), CES, UNI, NNI and Q.2931.

Sponsor Source

The ATM protocols is based on standards developed by the ITU.
http://www.atmforum.com/standards/approved.html
ATM Forum approved specifications

Reference

http://www.atmforum.com/standards/approved.html
ATM Forum approved specifications
http://www.cisco.com/univercd/cc/td/doc/cisintwk/ito_doc/atm.htm
ATM Overview

Protocol Name

ATM PNNI: ATM Private Network-to-Network Interface

Protocol Description

The ATM Private Network-Node Interface (PNNI), an ATM network-to-network signaling protocol, provides mechanisms to support scalable, QoS-based ATM routing and switch-to-switch switched virtual connection (SVC) interoperability.

The PNNI (Private Network-to-Network Interface) is a hierarchical, dynamic link-state routing protocol. It is designed to support large-scale ATM networks. The PNNI protocol uses VPI/VCI 0,18 for its messages. In addition, it uses signalling messages to support connection establishment across multiple networks. PNNI is based on UNI 4.0 and Q.2931. Specific information elements were added to UNI 4.0 in order to support the routing process of PNNI. PNNI Signalling contains the procedure to dynamically establish, maintain and clear ATM connections at the private network to network interface or network node interface between 2 ATM networks or 2 ATM network nodes. The PNNI signalling protocol is based on the ATM forum UNI specification and on Q.2931.

PNNI Messages include:

ALERTING, CALL PROCEEDING, CONNECT, SETUP, RELEASE, RELEASE COMPLETE, NOTIFY, STATUS, STATUS ENQUIRY, RESTART, RESTART ACKNOWLEDGE, STATUS, ADD PARTY, ADD PARTY ACKNOWLEDGE, PARTY ALERTING, ADD PARTY REJECT, DROP PARTY, DROP PARTY ACKNOWLEDGE

Protocol Structure

The structure of the PNNI header is shown in the following illustration:

2	2	1	1	1	1
Packet type	Packet length	Prot ver	Newest ver	Oldest ver	Reserved

- Packet type: The following packet types are defined:
  1. Hello - Sent by each node to identify neighbor nodes belonging to the same peer group.
  2. PTSP - PNNI Topology State Packet. Passes topology information between groups.
  3. PTSE - PNNI Topology State Element (Request and Ack). Conveys topology parameters such as active links, their available bandwidth, etc.
  4. Database Summary - Used during the original database exchange between two neighboring peers.
- Packet length - The length of the packet.
- Prot ver - Protocol Version. The version according to which this packet was formatted.
- Newest ver / Oldest ver - Newest version supported / oldest version supported. The newest version supported and the oldest version supported fields are included in order for nodes to negotiate the most recent protocol version that can be understood by both nodes exchanging a particular type of packet.

Related protocols

ATM, BISDN, SONET, AAL0-AAL5, LAN Emulation (LANE), CES, UNI, NNI, MPOA and Q.2931

Sponsor Source

The ATM protocols are based on standards developed by the ITU.
http://www-comm.itsi.disa.mil/atmf/sig.html#af10.1
UNI 4.0 Specification
http://www.atmforum.com/standards/approved.html
ATM Forum approved specifications

Reference

http://www.atmforum.com/standards/approved.html#uni
ATM User-Network Interface Specification
http://www.cisco.com/univercd/cc/td/doc/cisintwk/ito_doc/atm.htm
ATM Overview

Protocol Name

Q.2931: ATM Signaling for B-ISDN

Protocol Description

Signalling is the process by which ATM users and the network exchange the control of information, request the use of network resources, or negotiate for the use of circuit parameters. Q.2931 based on Q.931 is the ITU version of signaling protocol. Q.2931 specifies the procedures for the establishment, maintenance and clearing of network connections at the B-ISDN user network interface. The UNI 3.1 specification is based on Q.2931. The procedures are defined in terms of messages exchanged.

The basic capabilities supported by Q.2931 Signaling are as follows:

- Demand (switched virtual) channel connections.
- Point-to-point switched channel connections.
- Connections with symmetric or asymmetric bandwidth requirements.
- Single-connection (point-to-point) calls.
- Basic signalling functions via protocol messages, information elements and procedures.
- Class X, Class A and Class C ATM transport services.
- Request and indication of signalling parameters.
- VCI negotiation.
- Out-of-band signalling for all signalling messages.
- Error recovery.
- Public UNI addressing formats for unique identification of ATM endpoints.
- End-to-end compatibility parameter identification.
- Signalling interworking with N-ISDN and provision of N-ISDN services.
- Forward compatibility.

The message types for Q.2931 are the same as in UNI, with a few exceptions, such as point-to-multipoint messages (which are not supported in version 3.0/3.1) and SETUP ACKNOWLEDGE and INFORMATION in version 4.0. New signaling messages specific to Q.2931are ALERTING, PROGRESS, SETUP ACKNOWLEDGE, INFORMATION, and NOTIFY.

The VPI/VCI pair and requested bandwidth are allocated as a result of a successful signalling exchange. These messages are sent over the Signalling ATM Adaptation Layer (SAAL), which ensures their reliable delivery.

Protocol Structure

Protocol stacks illustrated below support User Networking connection control signaling:

	User-Network Signalling
SAAL	UNI SSCF
	SSCOP
	AAL Type 5 Common Part
	ATM Layer
	Physical Layer

Q.2931Signaling Message:

8	7	6	5	4	3	2	1	bit/Octet
Protocol discriminator								1
0	0	0	0	Length of call reference value				2
Flag	Call reference value							3
Call reference value (continued)								4
								5
Message type								6
Message type (continued)								7
Message length								8
Message length (continued)								9
Variable length Information Elements as required								etc.

- Protocol discriminator -Distinguishes Messages for user-network call control from other messages. (9 for Q.2931 messages)
- Call reference - Unique number for every ATM connection which serves to link all signalling messages relating to the same connection. It is comprised of the call reference value and the call reference flag. The call reference flag indicates who allocated the call reference value.
- Message type - Connection control message types.
- Message length - The length of the contents of a message.
- Information Elements - There are several types of information elements. Some may appear only once in the message; others may appear more than once. Depending on the message type, some information elements are mandatory and some are optional. The order of the information elements does not matter to the signalling protocol. Information elements defined in Q.2931 are as follows:
  - Called party number.
  - Called party sub-address.
  - Transit network selection.
  - Restart indicator.
  - Narrow-band low layer compatibility.
  - Narrow-band high layer compatibility.
  - Broadband locking shift.
  - Broadband non-locking shift.
  - Broadband sending complete.
  - Broadband repeat indicator.

- Calling party number.
- Calling party sub-address.
- ATM adaptation layer parameters.
- ATM traffic descriptor.
- Connection identifier.
- OAM traffic descriptor.
- Quality of Service parameter.
- Broadband bearer capability.
- Broadband Low Layer Information (B-LLI).
- Broadband High Layer Information (B-HLI).
- End-to-end transit delay.
- Notification indicator.
- Call state.
- Progress indicator.
- Narrow-band bearer capability.
- Cause

Related protocols

ATM, BISDN, SONET, AAL0-AAL5, LAN Emulation (LANE), CES, UNI, NNI, and MPOA

Sponsor Source

http://members.tripod.com/ATM_protocols/PtoP/Q_2931.html
Q.2931 Recommendation
http://www-comm.itsi.disa.mil/atmf/sig.html#af10.1
UNI 4.0 Specification
http://www.atmforum.com/standards/approved.html
ATM Forum approved specifications

Reference

http://members.tripod.com/ATM_protocols/PtoP/Q_2931.html
Q.2931 Recommendation
http://www.atmforum.com/standards/approved.html#uni
ATM User-Network Interface Specification
http://www.cisco.com/univercd/cc/td/doc/cisintwk/ito_doc/atm.htm
ATM Overview

Protocol Name

SONET/SDH: Synchronous Optical Network and Synchronous Digital Hierarchy

Protocol Description

The Synchronous Optical Network (SONET), also calledand Synchronous Digital Hierarchy (SDH), are a set of related standards for synchronous data transmission over fiber optic networks that are often used for framing and synchronization at the physical layer. SONET is the United States version of the standard published by the American National Standards Institutue (ANSI). SDH is the international version of the standard published by the International Telecommunications Union (ITU).

SONET/SDH can be used in an ATM or non-ATM environment. Packet Over SONET/SDH (POS) maps IP datagrams into the SONET frame payload using Point-to-Point Protocol (PPP). In the ATM enviroment, connections to SONET/SDH lines may be via multi-mode, single-mode or UTP.

SONET is based on transmission at speeds of multiples of 51.840 Mbps (STS-1) and SDH is based on STM-1 which has a data rate of 155.52 Mbps, equivalent to STS-3.

The following table lists the hierarchy of the most common SONET/SDH data rates:

SONET Signal	Bit Rate (Mbp)	SDH Signal	SONET Capacity	SDH Capacity
STS-1/ OC-1	51.840	STM-0	28 DS1 or DS3	21 E1s
STS-3/ OC-3	155.520	STM-1	84 DS1s or 3 DS3s	63 E1s or E4
STS-12/ OC-12	622.080	STM-4	336 DS1s or 12 DS3s	252 E1s or 4 E4s
STS-48 / OC-48	2.488.32	STM-16	1,344 DS1s or 48 DS3s	1,008 E1s or 16 E4s
STS-192/ OC-192	9,953.280	STM-64	5,376 DS1s or 192 DS3s	4,032 E1s or 64 E4s
STS-768/ OC-768	39,813.120	STM-256	21,504 DSs or 786 DS3	16,128 E1s or 256 E4s

Other rates such as OC-9, OC-18, OC-24, OC-36, OC-96 and OC-768 are referenced in some of the standards documents but were not widely implemented. Higher rate maybe defined for future implementations.

Protocol Structure

The frame structure of STS and STM is different. We only display the details of the STS-1 frame structure here. The STS-1 frame is composed of octets which are nine rows high and 90 columns wide. The first three columns are used by the Transport Overhead (TOH) and contain framing, error monitoring, management and payload pointer information. The data (Payload) uses the remaining 87 columns, of which the first column is used for Path Overhead (POH). A pointer in the TOH identifies the start of the payload, which is referred to as the Synchronous Payload Envelope or SPE.

9 Columns	POH	260 Columns	
	J1		9 Rows
	B3		
	C2		
	G1		
	F2		
	H4		
	Z3		
	Z4		
	Z5		

- SOH— Section Overhead.
  A1, A2— Frame alignment. These octets contain the value of 0xF628. The receiver searches for these values in the incoming bit stream. These bytes are not scrambled.
  C1— STS-1 identification. Since OC-3c and STM-1 contain three STS-1 streams, the three C1 bytes contain 0x01, 0x02 and 0x03, respectively.
  B1— Section error monitoring. Contains BIP-8 of all bits in the previous frame using even parity, before scrambling.
- LOH— Line Overhead
  B2— Line error monitoring. Contains BIP-24 calculated over all bits of the line overhead of the previous frame with even parity.
  H1 (bits 1-4)— New data flag (specifies when the pointer has changed), path AIS.
  H1 and H2 (bits 7-16)— Pointer value, path AIS. These bytes specify the offset between the pointer and the first payload byte. A change in this value is ignored until received at least three consecutive times.
  H1* and H2*— Concatenation indication, path AIS.
  H3— Pointer action (used for frequency justification), path AIS.
  K2 (bits 6-8)— Line AIS, line FERF, removal of line FERF.
  Z2— Line FEBE. This contains the number of B2 (BIP-24) errors detected in the previous interval.
- POH— Path Overhead
  J1— STS path trace. This byte is used repetitively to transmit a 64-byte fixed string so that the receiving terminal in a path can verify its continued connection to the transmitter. Its contents

are unspecified.
B3— Path error monitoring. Path BIP-8 over all bits of the payload of the previous frame, using even parity before scrambling.
C2— Path signal level indicator. Contains one of two codes:
Code 0: indicates STS payload unequipped: no path originating equipment.
Code 1: indicates STS payload equipped: non-specific payload for payloads that need no further differentiation.
G1 (bits 1-4)— Path FEBE. Allows monitoring of complete full-duplex path at any point along a complex path.
G1 (bit 5)— Path yellow alarm, path RDI (Remote Defect Indicator).

Related protocols

ATM, STS, STM

Sponsor Source

SONET is an ANSI standard defined in documents T1.105. xx and T1.119.xx and SDH is defined by ITU-T in documents G.707, G.781, G.782, G.783 and G.803.

Reference

http://www.iec.org/online/tutorials/sonet/: Synchronous Optical Network (SONET)

Broadband Access Protocols

Protocol Name

BISDN: Broadband Integrated Services Digital Network (Broadband ISDN)

Protocol Description

Broadband Integrated Services Digital Network (BISDN or Broadband ISDN) is designed to handle high-bandwidth applications. BISDN currently uses ATM technology over SONET-based transmission circuits to provide data rates from 155 to 622Mbps and beyond, contrast with the traditional narrowband ISDN (or N-ISDN), which is only 64 kps basically and up to 2 Mbps maximum.

The designed Broadband ISDN (BISDN) services can be categorised as follows:

- Conversational services such as telephone-like services, which are also supported by N-ISDN. The additional bandwidth offered also allows such services as video telephony, video conferencing and high volume, high speed data transfer.
- Messaging services, which are mainly store-and-forward services. Applications include voice and video mail, as well as multi-media mail and traditional electronic mail.
- Retrieval services which provide access to (public) information stores, information being sent to the user on demand only.
- No user control of presentation. This would be for instance, a TV broadcast, where the user can choose simply either to view or not.
- User controlled presentation. This would apply to broadcast information that the user can partially control.

The B-ISDN is designed to offer both connection oriented and connectionless services. In both cases, the broadband information transfer is provided by the use of asynchronous transfer mode (ATM), using end-to-end logical connections or virtual circuits. Broadband ISDN uses out-of-band signaling (as does N-ISDN). Instead of using a D Channel as in N-ISDN, a special virtual circuit channel can be used for signaling. However, B-ISDN has not widely been deployed so far.

Protocol Structure

Broadband ISDN protocol reference model is based on the ATM reference model.

ATM adaptation layer (AAL). This layer is responsible for mapping the service offered by ATM to the service expected by the higher layers. It has two sublayers.

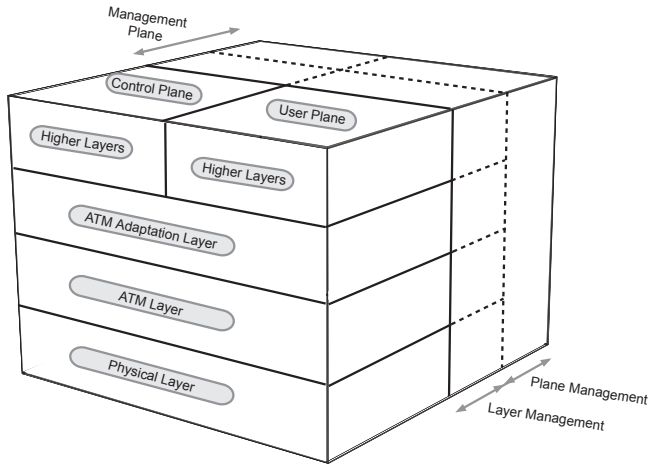


Figure 2-10: ATM Reference Model

ATM Layer. This layer is independent of the physical medium over which transmission is to take place. It has those functions: Generic flow control (GFC) function, Cell header generation and extraction, Cell multiplex and demultiplex.

Physical layer. This consists of two sublayers: Transport Convergence (TC) and Physical medium (PM)

The management plane consists of two functions to perform layer management and plane management. The plane management is not layered as the other layers are. This is because it needs information on all aspects of the system to provide management facilities for the systems as a whole. The layer management provides information and control facilities for the protocol entities that exists in each individual layer. This includes operation and maintenance (OAM) functions for each layer.

The control plane is responsible for the supervision of connections, including call set-up, call release and maintenance.

The user plane provides for the transfer of user information. It also includes mechanisms to perform error recovery, flow control etc.

Related protocols

ISDN, ATM, B-ICI

Sponsor Source

BISDN protocol is defined by ITU-T.

Reference

http://www.cs.ucl.ac.uk/staff/S.Bhatti/D51-notes/node35.html
Broadband ISDN

Protocol Name

ISDN: Integrated Services Digital Network

Protocol Description

Integrated Services Digital Network (ISDN) is a system with digitized phone connections. For decades, telephony has used purely analogue connections. ISDN is the first protocol to define a digital communications line that allows for the transmission of voice, data, video and graphics, at high speeds, over standard communication lines. The various media are simultaneously carried by bearer channels (B channels) occupying a bandwidth of 64 kbits per second (some switches limit bandwidth to 56 kb/s). A defined data channel (D channel) handles signaling at 16 kb/s or 64 kb/s, depending on the service type. ISDN is not restricted to public telephone networks alone; it may be transmitted via packet switched networks, telex, CATV networks, etc. There are two basic types of ISDN service:

- Basic Rate Interface (BRI) - consists of two 64 kb/s B channels and one 16 kb/s D channel for a total of 144 kb/s. This basic service is intended to meet the needs of most individual users. The U interface provided by the telco for BRI is a 2-wire, 160 kb/s digital connection. Echo cancellation is used to reduce noise, and data encoding schemes (2B1Q in North America, 4B3T in Europe) permit this relatively high data rate over ordinary single-pair local loops.
- Primary Rate Interface (PRI) is intended for users with greater capacity requirements. Typically the channel structure is 23 B channels plus one 64 kb/s D channel for a total of 1536 kb/s. In Europe, PRI consists of 30 B channels plus one 64 kb/s D channel for a total of 1984 kb/s. It is also possible to support multiple PRI lines with one 64 kb/s D channel using Non-Facility Associated Signaling (NFAS).

The CCITT (now ITU-T) study group responsible for ISDN first published a set of ISDN recommendations in 1984. Prior to this publication, various geographical areas had developed different versions of ISDN. The use of nation-specific information elements is enabled by using the Codeset mechanism which allows different areas to use their own information elements within the data frames. A common nation-specific ISDN variant is National ISDN by Bellcore, used in the USA. It has four network-specific message types. It does not have any single octet information elements. Other changes are the addition of the SEGMENT, FACILITY and REGISTER message types and the Segmented Message and Extended Facility information elements. Also, some meanings of field values have changed and some new accepted field values have been added.

Due to its limitation of bandwidth and services, this traditional ISDN is called narrowband ISDN, in contrast to the BISDN (Broadband ISDN).

Protocol Structure

Below is the general structure of the ISDN frame:

8	7	6	5	4	3	2	1
Protocol discriminator							
0	0	0	0	Length of reference call value			
Flag	Call reference value						
0	Message type						
Other information elements as required							

- Protocol discriminator - The protocol used to encode the remainder of the Layer.
- Length of call reference value - Defines the length of the next field. The Call reference may be one or two octets long depending on the size of the value being encoded.
- Flag - Set to zero for messages sent by the party that allocated the call reference value; otherwise set to one.
- Call reference value - An arbitrary value that is allocated for the duration of the specific session, which identifies the call between the device maintaining the call and the ISDN switch.
- Message type - Defines the primary purpose of the frame. The message type may be one octet or two octets (for network specific messages). When there is more than one octet, the first octet is coded as eight zeros. A complete list of message types is given in ISDN Message Types below.
- ISDN Information Elements - There are two types of information elements: single octet and variable length.
- Single octet information elements - The single octet information element appears as follows:

8	7	6	5	4	3	2	1
1	Information element identifier				Information element		

- Variable length information elements - The following is the format for the variable length information element:

8	7	6	5	4	3	2	1
0	Information element identifier						
Length of information elements							
Information elements (multiple bytes)							

- The information element identifier identifies the chosen element and is unique only within the given Codeset. The length of the information element informs the receiver as to the amount of the following

octets belonging to each information element.

- ISDN Message Types - The possible ISDN message types are: Call Establishment, Call Information Phase, Call Clearing, and Miscellaneous.
- Codeset - Three main Codesets are defined. In each Codeset, a section of the information elements are defined by the associated variant of the protocol:

Codeset 0	The default code, referring to the CCITT set of information elements.
Codeset 5	The national specific Codeset.
Codeset 6	The network specific Codeset.

- CPE - Customer Premises Equipment; Refers to all ISDN compatible equipment connected at the user site. Examples of devices are telephone, PC, Telex, Facsimile, etc. The exception is the FCC definition of NT1. The FCC views the NT1 as a CPE because it is on the customer site, but the CCITT views NT1 as part of the network. Consequently the network reference point of the network boundary is dependent on the variant in use.
- ISDN Channels B, D and H - The three logical digital communication channels of ISDN perform the following functions:

B-Channel	Carries user service information including: digital data, video, and voice.
D-Channel	Carries signals and data packets between the user and the network
H-Channel	Performs the same function as B-Channel, but operates at rates exceeding DS-0 (64 Kbps). They are implemented as H0 (384 kb/s (6 B channels), H10 (1472 kb/s -23 B channels), H11 (1536 kb/s; 24 B channels), and H12 (1920 kb/s for International -E1 only).

Related protocols

LAP-D, BRI, PRI, Q.920-Q.923, LAP-B, X.25, Q.931, ATM

Sponsor Source

ISDN protocol is defined by ITU I-series and G-series documents (Physical Layer) and Q-series documents (Data-link and Network layers).

Reference

http://www.nationalisdncouncil.com/isdnassistance
The National ISDN Council (NIC) specification
http://www.catcouncil.org/isdn/index
The Council for Access Technologies (CAT formerly NIC) specification and documents.

Protocol Name

LAP-D: ISDN Link Access Protocol-Channel D

Protocol Description

LAP -D is the Layer 2 protocol in the ISDN suite, almost identical to the X.25 LAP-B protocol. The three logical digital communication channels of ISDN perform the following functions:

- B-Channel - Carries user service information including: digital data, video, and voice.
- D-Channel - Carries signals and data packets between the user and the network
- H-Channel - Performs the same function as B-Channels, but operates at rates exceeding DS-0 (64 Kbps).

The Link Establishment process in ISDN performed by LAP-D is as follows:

1. The TE (Terminal Endpoint) and the Network initially exchange Receive Ready (RR) frames, listening for someone to initiate a connection
2. The TE sends an Unnumbered Information (UI) frame with a SAPI of 63 (management procedure, query network) and TEI of 127 (broadcast)
3. The Network assigns anavailable TEI(in the range 64-126)
4. The TE sends a Set Asynchronous Balanced Mode (SABME) frame with a SAPI of 0 (call control, used to initiate a SETUP) and a TEI of the value assigned by the network
5. The network responds with an Unnumbered Acknowledgement (UA), SAPI=0, TEI=assigned.

The LAPD is defined in CCITT Q.920/921. LAPD works in the Asynchronous Balanced Mode (ABM). This mode is totally balanced (i.e., no master/slave relationship). Each station may initialize, supervise, recover from errors, and send frames at any time. The protocol treats the DTE and DCE as equals.

Protocol Structure

The format of a standard LAPD frame is as follows:

Flag	Address field	Control field	Information	FCS	Flag

- Flag - The value of the flag is always (0x7E).” Bit Stuffing” technique is used in order to ensure that the bit pattern of the frame delimiter flag does not appear in the data field of the frame.
- Address field - The first two bytes of the frame after the header flag are known as the address field. The format of the address field is as follows:

1	2	3	4	5	6	7	8
SAPI (6 bits)						C/R	EA0
TEI (7 bits)							EA1

  - SAPI (Service access point identifier), 6-bits (see below)
  - C/R (Command/Response) bit indicates if the frame is a command or a response
  - EA0 (Address Extension) bit indicates whether this is the final octet of the address or not
  - TEI (Terminal Endpoint Identifier) 7-bit device identifier (see below)
  - EA1 (Address Extension) bit, same as EA0

- Control field - The field following the Address Field is called the Control Field and serves to identify the type of the frame. In addition, it includes sequence numbers, control features and error tracking according to frame type. The following are the Supervisory Frame Types defined in LAPD:

RR	Information frame acknowledgement and indication to receive more.
REJ	Request for retransmission of all frames after a given sequence number.
RNR	Indicates a state of temporary occupation of station (e.g., window full).

- Some Unnumbered Frame Types supported in LAPD are – DISC (Request disconnection), UA (Acknowledgement frame), DM (Response to DISC indicating disconnected mode), FRMR (Frame reject), SABM, SABME, UI and XID.
- FCS -The Frame Check Sequence (FCS) enables a high level of physical error control by allowing the integrity of the transmitted frame data to be checked. The sequence is first calculated by the transmitter using an algorithm based on the values of all the bits in the frame. The receiver then performs the same calculation on the received frame and compares its value to the CRC.
- Window size - LAPD supports an extended window size (modulo 128) where the number of possible outstanding frames for acknowledgement is raised from 8 to 128. This extension is generally used for satellite transmissions where the acknowledgement delay is significantly greater than the frame transmission time. The type of the link initialization frame determines the modulo of the session and an “E” is added to the basic frame type name (e.g., SABM becomes SABME).

Related protocols

LAP-D, BRI, PRI, Q.920-Q.923, LAP-B, X.25, Q.931, ATM

Sponsor Source

The LAP-D protocol is based on standards developed by the ITU Q-series documents.

Reference

http://www.nationalisdncouncil.com/isdnassistance
The National ISDN Council (NIC) specification
http://www.catcouncil.org/isdn/index
The Council for Access Technologies (CAT formerly NIC) specification and documents.

Protocol Name

Q.931: ISDN Network Layer Protocol for Signaling

Protocol Description

Q.931, the Network Layer (layer 3) protocol in the telecommunication architecture, is used in ISDN for call establishment and the maintenance, and termination of logical network connections between two devices. Q.931 is one of the layer 3 protocols in the telecommunication architecture specified by ITU in Q series documents Q.930 to Q.939.

During the layer 3 call setup, messages sent and received among three parties: 1) the Caller, 2) the ISDN Switch, and 3) the Receiver. Following is an example of call setup steps:

- Caller sends a SETUP to the Switch.
- If the SETUP is OK, the switch sends a CALL PROCeeding to the Caller, and then a SETUP to the Receiver.
- The Receiver gets the SETUP. If it is OK, it rings the phone and sends an ALERTING message to the Switch.
- The Switch forwards the ALERTING message to the Caller.
- When the receiver answers the call, it sends a CONNECT message to the Switch
- The Switch forwards the CONNECT message to the Caller.
- The Caller sends a CONNECT ACKnowledge message to the Switch
- The Switch forwards the CONNECT ACK message to the Receiver.
- Done. The connection is now up.

What services and features the telco switch provides to the attached ISDN device are specified in the optional field - Service Profile IDs (SPIDs); when they are used, they are only accessed at device initialization time, before the call is set up. The format of the SPID is usually the 10-digit phone number of the ISDN line, plus a prefix and a suffix that are sometimes used to identify features on the line but can be whatever the Telco decides they should be. Details can be found in the Q series documents.

Protocol Structure

Information Field Structure - The Information Field is a variable length field that contains the Q.931 protocol data:

1	2	3	4	5	6	7	8
Protocol Discriminator							
0	0	0	0	Length of CRV			
Call Reference Value (1 or 2 octets)							
0	Message Type						
Mandatory & Optional Information Elements (variable)							

- Protocol Discriminator (1 octet) - identifies the Layer 3 protocol. If this is a Q.931 header, this value is always 0816.
- Length(1 octet) - indicates the length of the next field, the CRV.
- Call Reference Value (CRV) (1 or 2 octets) - used to uniquely identify each call on the user-network interface. This value is assigned at the beginning of a call and becomes available for another call when the call is cleared.
- Message Type (1 octet) - identifies the message type (i.e., SETUP, CONNECT, etc.). This determines what additional information is required and allowed.
- Mandatory and Optional Information Elements (variable length) - are options that are set depending on the Message Type.

Related protocols

LAP-D, BRI, PRI, Q.920-Q.923, LAP-B, X.25, Q.931, ATM

Sponsor Source

The Q.931 protocol is based on standards developed by the ITU Q-series documents.

Reference

http://www.nationalisdncouncil.com/isdnassistance
The National ISDN Council (NIC) specification
http://www.catcouncil.org/isdn/index
The Council for Access Technologies (CAT formerly NIC) specification and documents.

Protocol Name

DOCSIS: Data Over Cable Service Interface Specification

Protocol Description

Data Over Cable Service Interface Specification(DOCSIS), developed by CableLabs and approved by the ITU, defines interface requirements for cable modems involved in high-speed data distribution (both MPEG and IP data) over cable television system networks. Other devices that recognize and support the DOCSIS standard include HDTVs and Web enabled set-top boxes for regular televisions.

There are two key components in the DOCSIS architecture: a Cable Modem (CM) which is located at the customer's premise, and the Cable Modem Termination System (CMTS), which is located at the head end of the service provider and used to aggregate traffic from multiple Cable Modems and then communicate with the backbone network. DOCSIS specifies modulation schemes and the protocol for exchanging bidirectional signals between these two components over cable.

Three versions of DOCSIS are now implemented and deployed:

DOCSIS 1.0 - High Speed Internet Access. Key features: Downstream traffic transfer rates between 27 and 36 Mbps over a radio frequency (RF) path in the 50 MHz to 750+ MHz range, and upstream traffic transfer rates between 320 Kbps and 10 Mbps (average 5 Mbps) over an RF path between 5 and 42 MHz. However, because data over cable travels on a shared loop, individuals will see transfer rates drop as more users gain access.

DOCSIS 1.1 – Data, Voice, Gaming and Streaming. Key features: DOSCIS 1.1 is interoperable with DOCSIS 1.0. It provides enhanced QoS for multiple services such as voice and streaming; improved security over DOCSIS 1.0; and more robust upstream data transmission (average 10 Mbps).

DOCSIS 2.0 – Has added capacity for symmetric services by operating at 64 QAM and having new 6.4 MHz wide channel. It has increased bandwidth for IP traffic by using enhanced modulation and improved error correction. The result for upstream transmission is 30 Mbps, which is 3 times better than DOCSIS 1.1 and 6 times than DOSCIS 1.0. DOSCIS 2.0 is interoperable and backward compatible with DOCSIS 1.x.

The latest DOCSIS specification eDOCSIS has been published to the industry. eDOCSIS, embedded DOCSIS, provides a subordinate function at the core chip level to the host device. Rather than leveraging a home networking protocol, an embedded DOCSIS device feeds directly into a cable network's DOCSIS channel. eDOCSIS is intended to solve end device (and traffic) management, configuration and security issues, to significantly reduce cost in the service operation and to improve speed and quality of end customer services.

Protocol Structure

The specifications of the various DOCSIS versions can be found in the attached reference documents.

Related protocols

IP, MPEG, DOCSIS 1.0, DOCSIS 1.1, DOCSIS 2.0, eDOCSIS

Sponsor Source

DOCSIS is defined by CableLabs (http://www.cablemodem.com/) and approved by ITU.

Reference

http://www.cablemodem.com/downloads/specs/SP_CMTS_NSII01-960702.pdf
DOCSIS - CMTS Network Site Interface Specification
http://www.cablemodem.com/downloads/specs/SP-CMCI-I09-030730.pdf
DOCSIS 1.1 Specification
http://www.cablemodem.com/downloads/specs/SP-RFIv2.0-I04-030730.pdf
DOCSIS 2.0 - Radio Frequency Interface Specification
http://www.cablemodem.com/downloads/specs/SP-OSSIv2.0-I04-030730.pdf
DOCSIS 2.0 - Operation Support System Interface Specification
http://www.cablemodem.com/downloads/specs/SP-eDOCSIS-I02-031117.pdf
eDOCSIS Specification

Protocol Name

xDSL: Digital Subscriber Line Technologies (DSL, IDSL, ADSL, HDSL, SDSL, VDSL, G.Lite)

Protocol Description

DSL (Digital Subscriber Line) is a modem technology for broadband data access over ordinary copper telephone lines (POTS) from homes and businesses. xDSL refers collectively to all types of DSL, such as ADSL (and G.Lite), HDSL, SDSL, IDSL and VDSL etc. They are sometimes referred to as last-mile (or first mile) technologies because they are used only for connections from a telephone switching station to a home or office, not between switching stations.

xDSL is similar to ISDN in as much as both operate over existing copper telephone lines (POTS) using sophisticated modulation schemes and both require short runs to a central telephone office (usually less than 20,000 feet). However, xDSL offers much higher speeds - up to 32 Mbps for upstream traffic, and from 32 Kbps to over 1 Mbps for downstream traffic.

Several modulation technologies are used by various kinds of DSL:

- Discrete Multitone Technology (DMT)
- Simple Line Code (SLC)
- Carrierless Amplitude Modulation (CAP)
- Multiple Virtual Line (MVL)
- Discrete Wavelet Multitone (DWMT).

To interconnect multiple DSL users to a high-speed backbone network, the telephone company uses a Digital Subscriber Line Access Multiplexer (DSLAM). The DSLAM aggregates data transmission from all access DSL lines and then connects to an asynchronous transfer mode (ATM) network. At the other end of each transmission, a DSLAM demultiplexes the signals and forwards them to appropriate individual DSL connections.

Most DSL technologies require that a signal splitter be installed at a customer's premises. However, it is possible to manage the splitting remotely from the central office. This is known as splitterless DSL, "DSL Lite," G.Lite, or Universal ADSL.

Protocol Structure

The following table provides a summary of various DSL specifications.

Type	Description	Data Rate	Mode	Distance	Applications
IDSL	ISDN Digital Subscriber Line	128 kbps	Duplex	18k ft on 24 gauge wire	ISDN service Voice and data communication
HDSL	High data rate Digital Subscriber Line	1.544 Mbps to 42.048 Mbps	Duplex	12k ft on 24 gauge wire	T1/E1 service Feeder plant, WAN, LAN access, server access
SDSL	Single Line Digital Subscriber Line	1.544 Mbps to 2.048 Mbps	Duplex	12k ft on 24 gauge wire	Same as HDSL plus premises access for symmetric services
ADSL	Asymmetric Digital Subscriber Line	1.5 to 9 Mbps 16 to 640 kbps	Down Up	Up to 18k ft on 24 gauge wire	Internet access, video on-demand, simplex video, remote LAN access, interactive multimedia
DSL Lite (G.Lite)	"Splitterless" DSL	1.544 Mbps to 6 Mbps 16 to 640 kbps	Down Up	18k ft on 24 gauge wire	The standard ADSL; sacrifices speed for not having to install a splitter at the user's premises.
VDSL	Very high data rate Digital Subscriber Line	13 to 52 Mbps 1.5 to 2.3 Mbps	Down Up	1k to 4.5k ft depending on data rate	Same as ADSL plus HDTV

Related protocols

ISDN, DSL, ADSL, HDSL, VDSL, SDSL, G.Lite, IDSL, ATM

Sponsor Source

DSL is defined by ITU-T (www.itu.org) and DSL Forum (www.dslforum.org)

Reference

http://www.dslforum.org/
Educate yourself about DSL
http://www.cisco.com/univercd/cc/td/doc/cisintwk/ito_doc/adsl.htm
Digital Subscriber Line

PPP Protocols

Protocol Name

PPP: Point-to-Point Protocols

Protocol Description

The Point-to-Point Protocol (PPP) suite provides a standard method for transporting multi-protocol datagrams over point-to-point links. PPP was originally devised as an encapsulation protocol for transporting IP traffic between two peers. It is a data link layer protocol (layer 2 in the OSI model) in the TCP-IP protocol suite over synchronous modem links, as a replacement for the non-standard layer 2 protocol SLIP. However, other protocols other than IP can also be carried over PPP, including DECnet and Novell's Internetwork Packet Exchange (IPX).

PPP is comprised of the following main components:

Encapsulation: A method for encapsulating multi-protocol datagrams. The PPP encapsulation provides for multiplexing of different network-layer protocols simultaneously over the same link. The PPP encapsulation has been carefully designed to retain compatibility with most commonly used supporting hardware.

Link Control Protocol: The LCP provided by PPP is versatile and portable to a wide variety of environments. The LCP is used to automatically agree upon the encapsulation format options, handle varying limits on sizes of packets, detect a looped-back link and other common misconfiguration errors, and terminate the link. Other optional facilities provided are authentication of the identity of its peer on the link, and determination when a link is functioning properly and when it is failing.

Network Control Protocol: An extensible Link Control Protocol (LCP) for establishing, configuring, and testing and managing the data-link connections.

Configuration: Easy and self configuration mechanisms using Link Control Protocol. This mechanism is also used by other control protocols such as Network Control Protocols (NCPs).

In order to establish communications over a point-to-point link, each end of the PPP link must first send LCP packets to configure and test the data link. After the link has been established and optional facilities have been negotiated as needed by the LCP, PPP must send NCP packets to choose and configure one or more network-layer protocols. Once each of the chosen network-layer protocols has been configured, datagrams from each network-layer protocol can be sent over the link.

The link will remain configured for communications until explicit LCP or NCP packets close the link down, or until some external event occurs (an inactivity timer expires or network administrator intervention).

Protocol Structure

1byte	2bytes	3bytes	5bytes	Variable…	2 – 4bytes
Flag	Address	Control	Protocol	Information	FCS

- Flag— indicates the beginning or end of a frame, consists of the binary sequence 01111110.
- Address— contains the binary sequence 11111111, the standard broadcast address. (Note: PPP does not assign individual station addresses.)
- Control— contains the binary sequence 00000011, which calls for transmission of user data in an unsequenced frame.
- Protocol— identifies the protocol encapsulated in the information field of the frame.
- Information—zero or more octet(s) contains the datagram for the protocol specified in the protocol field.
- FCS—Frame Check Sequence (FCS) Field, normally 16 bits. By prior agreement, consenting PPP implementations can use a 32-bit FCS for improved error detection.

Related protocols

PPPoE, PPPoA, SLIP, CHAP, HDLC, LCP, NCP, L2TP, CHAP, PAS, IPCP, IPv6CP, IPX, DECNet

Sponsor Source

PPP is defined by IETF (http://www.ietf.org) RFC1661 with an update RFC2153.

Reference

http://www.javvin.com/protocol/rfc1661.pdf
The Point-to-Point Protocol (PPP)
http://www.javvin.com/protocol/rfc2153.pdf
PPP Vendor Extensions

Protocol Name

BAP: PPP Bandwidth Allocation Protocol (BAP)
BACP: PPP Bandwidth Allocation Control Protocol (BACP)

Protocol Description

The Bandwidth Allocation Protocol (BAP) can be used to manage the number of links in a multi-link bundle. BAP defines datagrams to coordinate adding and removing individual links in a multi-link bundle, as well as specifying which peer is responsible for various decisions regarding managing bandwidth during a multi-link connection. The Bandwidth Allocation Control Protocol (BACP) is the associated control protocol for BAP. BACP defines control parameters for the BAP protocol to use.

As PPP multilink implementations become increasingly common, there is a greater need for some conformity in how to manage bandwidth over such links. BACP and BAP provide a flexible yet robust way of managing bandwidth between 2 peers. BAP does this by defining Call-Control packets and a protocol that allows peers to co-ordinate the actual bandwidth allocation and de-allocation. Phone number deltas may be passed in the Call-Control packets to minimize the end user's configuration.

BAP defines packets, parameters and negotiation procedures to allow two endpoints to negotiate gracefully adding and dropping links from a multilink bundle. BAP allows two peer implementations to manage the bandwidth available to the protocols using the multilink bundle by negotiating when to add and drop links. Use of the negotiation features of BAP makes it unnecessary to require a 'common' algorithm for determining when to add and remove links in a multilink bundle.

After BACP reaches the opened state, either peer MAY request that another link be added to the bundle by sending a BAP Call- or Callback-Request packet. A Call-Request packet is sent if the implementation wishes to originate the call for the new link, and a Callback-Request packet is sent if the implementation wishes its peer to originate the call for the new link. The implementation receiving a Call- or Callback-Request MUST respond with a Call- or Callback-Response with a valid Response Code.

Protocol Structure

BAP Packet structure:

8	16bit	Variable
Type	Length	Data

- Type - Indicates the type of the BAP Datagram Option. This field is binary coded Hexadecimal.

BACP packet structure:

8	16	32bit	Variable
Code	Identifier	Length	Data

- Code - Decimal value which indicates the type of BACP packet.
- Identifier - Decimal value which aids in matching requests and replies.
- Length - Length of the BACP packet, including the Code, Identifier, Length and Data fields.
- Data - Variable length field which may contain one or more configuration options.

Related protocols

PPP, PPPoE, PPPoA, SLIP, CHAP, HDLC, LCP, NCP

Sponsor Source

BAP and BACP are defined by IETF (http://www.ietf.org) .

Reference

http://www.javvin.com/protocol/rfc2125.pdf
The PPP Bandwidth Allocation Protocol (BAP) / The PPP Bandwidth Allocation Control Protocol (BACP)

Protocol Name

BCP: PPP Bridging Control Protocol

Protocol Description

The Bridging Control Protocol (BCP) is responsible for configuring the bridging protocol parameters on both ends of the point-to-point link. BCP uses the same packet exchange mechanism as the Link Control Protocol. BCP packets can not be exchanged until PPP has reached the Network-Layer Protocol phase. BCP packets received before this phase is reached are discarded.

The Point-to-Point Protocol (PPP) provides a standard method for transporting multi-protocol datagrams over point-to-point links. PPP defines an extensible Link Control Protocol (LCP) and proposes a family of Network Control Protocols (NCP) for establishing and configuring different network-layer protocols. This document defines the NCP for establishing and configuring Remote Bridging for PPP links.

BCP compares the configurations of two devices and seeks to negotiate an acceptable subset of their intersection so as to enable correct interoperation even in the presence of minor configuration or implementation differences. In the event that a major misconfiguration is detected, the negotiation will not complete successfully, resulting in the link coming down or not coming up. It is possible that if a bridged link comes up with a rogue peer, network information may be learned from forwarded multicast traffic, or denial of service attacks may be created by closing loops that should be detected and isolated or by offering rogue load.

Such attacks are not isolated to BCP. Any PPP NCP is subject to attack when connecting to a foreign or compromised device. However, no situations arise which are not common to all NCPs; any NCP that comes up with a rogue peer is subject to snooping and other attacks. Therefore, it is recommended that links on which this may happen should be configured to use PPP authentication during the LCP start-up phase.

Protocol Structure

8	16	32bit	Variable
Code	Identifier	Length	Data

- Code - Decimal value which indicates the type of BCP packet.
- Identifier - Decimal value which aids in matching requests and replies.
- Length - Length of the BCP packet, including the Code, Identifier, Length and Data fields.
- Data - Variable length field which may contain one or more configuration options. The following is a list of BCP configuration options:
  - Bridge-Identification
  - Line-Identification
  - MAC-Support
  - Tinygram-Compression
  - MAC-Address
  - Spanning Tree Protocol (old formatted)
  - IEEE-802-Tagged-Frame
  - Management-Inline
  - Bridge-Control-Packet-Indicator

Related protocols

PPP, PPPoE, PPPoA, SLIP, CHAP, HDLC, LCP, NCP

Sponsor Source

BCP is defined by IETF (http://www.ietf.org) .

Reference

http://www.javvin.com/protocol/rfc3518.pdf
Point-to-Point Protocol (PPP) Bridging Control Protocol (BCP)

Protocol Name

EAP: PPP Extensible Authentication Protocol

Protocol Description

The PPP Extensible Authentication Protocol (EAP) is for PPP authentication. EAP supports multiple authentication mechanisms. EAP does not select a specific authentication mechanism at Link Control Phase, but rather postpones this until the Authentication Phase. This allows the authenticator to request more information before determining the specific authentication mechanism. This also permits the use of a “back-end” server which actually implements the various mechanisms while the PPP authenticator merely passes through the authentication exchange.

1. After the Link Establishment phase is complete, the authenticator sends one or more Requests to authenticate the peer. The Request has a type field to indicate what is being requested. Examples of Request types include Identity, MD5-challenge, One-Time Passwords, Generic Token Card, etc. The MD5-challenge type corresponds closely to the CHAP authentication protocol. Typically, the authenticator will send an initial Identity Request followed by one or more Requests for authentication information. However, an initial Identity Request is not required, and MAY be bypassed in cases where the identity is presumed (leased lines, dedicated dial-ups, etc.).
2. The peer sends a Response packet in reply to each Request. The Response packet contains a type field which corresponds to the type field of the Request.
3. The authenticator ends the authentication phase with a Success or Failure packet.

The EAP protocol can support multiple authentication mechanisms without having to pre-negotiate a particular one during LCP Phase. Certain devices (e.g. an NAS) do not necessarily have to understand each request type and may be able to simply act as a passthrough agent for a “back-end” server on a host. The device only need look for the success/failure code to terminate the authentication phase.

However, EAP does require the addition of a new authentication type to LCP and thus PPP implementations will need to be modified to use it. It also strays from the previous PPP authentication model of negotiating a specific authentication mechanism during LCP.

Protocol Structure

The Authentication-Protocol Configuration Option format to negotiate the EAP Authentication Protocol is shown below:

8	16	32bit	Variable
Type	Length	Authentication-Protocol	Data

- Type - 3
- Length - 4
- Authentication-Protocol - C227 (Hex) for PPP Extensible Authentication Protocol (EAP)

One PPP EAP packet is encapsulated in the Information field of a PPP Data Link Layer frame where the protocol field indicates type hex C227 (PPP EAP). The EAP packet format is shown below:

8	16	32bit	Variable
Code	Identifier	Length	Data

- Code - The Code field identifies the type of EAP packet.
- EAP Codes are assigned as follows: 1 Request; 2 Response; 3 Success; 4 Failure.
- Identifier - The Identifier field aids in matching responses with requests.
- Length - The Length field indicates the length of the EAP packet including the Code, Identifier, Length and Data fields.
- Data - The format of the Data field is determined by the Code field.

Related protocols

PPP, CHAP

Sponsor Source

EAP is defined by IETF (http://www.ietf.org) .

Reference

http://www.javvin.com/protocol/rfc2284.pdf
PPP Extensible Authentication Protocol (EAP)

Protocol Name

CHAP: Challenge Handshake Authentication Protocol

Protocol Description

Challenge Handshake Authentication Protocol (CHAP) is used to periodically verify the identity of the peer using a 3-way handshake. This is done upon initial link establishment and may be repeated any time after the link has been established.

- After the Link Establishment phase is complete, the authenticator sends a “challenge” message to the peer.
- The peer responds with a value calculated using a “one-way hash” function.
- The authenticator checks the response against its own calculation of the expected hash value. If the values match, the authentication is acknowledged; otherwise the connection SHOULD be terminated.
- At random intervals, the authenticator sends a new challenge to the peer and the three steps above are repeated.

CHAP provides protection against playback attack by the peer through the use of an incrementally changing identifier and a variable challenge value. The use of repeated challenges is intended to limit the time of exposure to any single attack. The authenticator is in control of the frequency and timing of the challenges.

This authentication method depends upon a “secret” known only to the authenticator and that peer. The secret is not sent over the link.

Although the authentication is only one-way, by negotiating CHAP in both directions the same secret set may easily be used for mutual authentication.

Since CHAP may be used to authenticate many different systems, name fields may be used as an index to locate the proper secret in a large table of secrets. This also makes it possible to support more than one name/secret pair per system, and to change the secret in use at any time during the session.

CHAP requires that the secret be available in plaintext form. Irreversably encrypted password databases commonly available cannot be used. It is not as useful for large installations, since every possible secret is maintained at both ends of the link.

Protocol Structure

Configuration Option format for CHAP:

8	16	32	40bit
Type	Length	Authentication-Protocol	Algorithm

- Type - 3
- Length - 5
- Authentication-Protocol – C223 (Hex) for CHAP
- Algorithm The Algorithm field is one octet and indicates the authentication method to be used.

The structure of the CHAP packet is shown in the following illustration.

8	16	32bit	Variable
Code	Identifier	Length	Data ...

- Code - Identifies the type of CHAP packet. CHAP codes are assigned as follows:
  1 Challenge
  2 Response
  3 Success
  4 Failure
- Identifier - Aids in matching challenges, responses and replies.
- Length - Length of the CHAP packet including the Code, Identifier, Length and Data fields.
- Data - Zero or more octets, the format of which is determined by the Code field. For Success and Failure, the data field contains a variable message field which is implementation dependent.

Related protocols

PPP, PPPoE, PPPoA, LCP, NCP, PAP

Sponsor Source

CHAP is defined by IETF (http://www.ietf.org) .

Reference

http://www.javvin.com/protocol/rfc1994.pdf
PPP Challenge Handshake Authentication Protocol (CHAP)

Protocol Name

LCP: PPP Link Control Protocol

Protocol Description

The Link Control Protocol (LCP) is used to automatically agree upon the encapsulation format options, handle varying limits on sizes of packets, detect a looped-back link and other common misconfiguration errors, and terminate the link. Other optional facilities provided are authentication of the identity of its peer on the link, and determination when a link is functioning properly and when it is failing. The Link Control Protocol LCP in PPP is versatile and portable to a wide variety of environment.

There are three classes of LCP packets:

1. Link Configuration packets used to establish and configure a link (Configure-Request, Configure-Ack, Configure-Nak and Configure-Reject).
2. Link Termination packets used to terminate a link (Terminate-Request and Terminate-Ack).
3. Link Maintenance packets used to manage and debug a link (Code-Reject, Protocol-Reject, Echo-Request, Echo-Reply, and Discard-Request).

In the interest of simplicity, there is no version field in the LCP packet. A correctly functioning LCP implementation will always respond to unknown Protocols and Codes with an easily recognizable LCP packet, thus providing a deterministic fallback mechanism for implementations of other versions.

Regardless of which Configuration Options are enabled, all LCP Link Configuration, Link Termination, and Code-Reject packets (codes 1 through 7) are always sent as if no Configuration Options were negotiated. In particular, each Configuration Option specifies a default value. This ensures that such LCP packets are always recognizable, even when one end of the link mistakenly believes the link to be open.

Exactly one LCP packet is encapsulated in the PPP Information field, where the PPP Protocol field indicates type hex c021 (Link Control Protocol).

Protocol Structure

8	16	32bit	Variable
Code	Identifier	Length	Data

- Code - Decimal value which indicates the type of LCP packet:
  1 Configure-Request.
  2 Configure-Ack.
  3 Configure-Nak.
  4 Configure-Reject.
  5 Terminate-Request.
  6 Terminate-Ack.
  7 Code-Reject.
  8 Protocol-Reject.
  9 Echo-Request.
  10 Echo-Reply.
  11 Discard-Request.
  12 Link-Quality Report.
- Identifier - Decimal value which aids in matching requests and replies.
- Length - Length of the LCP packet, including the Code, Identifier, Length and Data fields.
- Data - Variable length field which may contain one or more configuration options.

Related protocols

PPP, PPPoE, PPPoA, SLIP, CHAP, HDLC, NCP

Sponsor Source

LCP is defined by IETF (http://www.ietf.org) .

Reference

http://www.javvin.com/protocol/rfc1570.pdf
PPP LCP Extensions.
http://www.javvin.com/protocol/rfc1661.pdf
The Point-to-Point Protocol (PPP)

Protocol Name

MPPP: MultiLink Point to Point Protocol (MultiPPP)

Protocol Description

MultiLink Point to Point Protocol (MultiPPP or MPPP) is a method for splitting, recombining and sequencing datagrams across multiple logical data links. PPP MultiLink (MP) protocol is based on an LCP option negotiation that permits a system to indicate to its peer that it is capable of combining multiple physical links into a “bundle”. Multilink is negotiated during the initial LCP option negotiation. A system indicates to its peer that it is willing to do multilink by sending the multilink option as part of the initial LCP option negotiation.

Once multilink has been successfully negotiated, the sending system is free to send PDUs encapsulated and/or fragmented with the multilink header. To establish communications over a point-to-point link, each end of the PPP link must first send LCP packets to configure the data link during Link Establishment phase. After the link has been established, there is an Authentication phase in which the Authentication protocols can be used to determine identifiers associated with each system connected by the link.

Multilink coordinates multiple independent links between a fixed pair of systems, providing a virtual link with greater bandwidth than any of the constituent members. The aggregate link, or bundle, is named by the pair of identifiers for two systems connected by the multiple links. A system identifier may include information provided by PPP Authentication and information provided by LCP negotiation. The bundled links can be different physical links, as in multiple async lines, but may also be instances of multiplexed links, such as ISDN, X.25 or Frame Relay. The links can be of different kinds, such as pairing dialup async links with leased synchronous links.

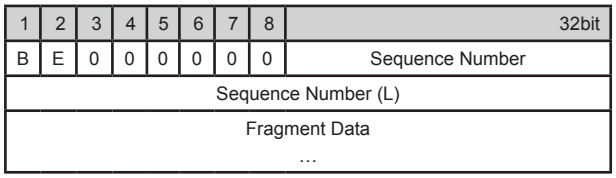
Multilink operation is moduled as a virtual PPP link-layer entity where packets received over different physical link-layer entities are identified as belonging to a separate PPP network protocol (the Multilink Protocol, or MP) and recombined and sequenced according to information present in a multilink fragmentation header. All packets received over links identified as belonging to the multilink arrangement are presented to the same network-layer protocol processing machine, whether they have multilink headers or not.

Network protocol packets to be transmitted using the multilink procedure are first encapsulated (but not framed) according to normal PPP procedures, and large packets are broken up into multiple segments sized appropriately for the multiple physical links. A new PPP header consisting of the Multilink Protocol Identifier, and the Multilink header is inserted before each section.

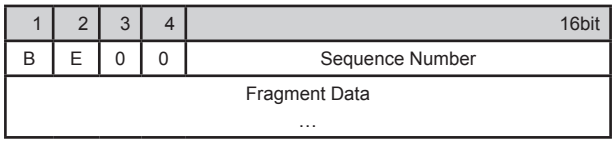
Protocol Structure

The header of MP can either be in Long Sequence Number Fragment format or Short Sequence Number Fragment Format.

Long Sequence Number Fragment Format



Short Sequence Number Fragment Format



- Address, Control – Compressed fields for the Address, the control field and the Protocol ID.
- PID (L), PID (H) - Protocol Identifier for PPP MultiLink; this is 0x00-0x3d
- B(egining) Bit - One bit field set to 1 on the first fragment derived from a PPP packet and set to 0 for all other fragments from the same PPP packet.
- E(nd) Bit - A one bit field set to 1 on the last fragment and set to 0 for all other fragments.
- 00 - Reserved field with the value of 0
- Sequence Number - The sequence field is a 24 bit or 12 bit number that is incremented for every fragment transmitted.
- Fragment Data - The data itself.

Related protocols

PPP, PPPoE, PPPoA, SLIP, CHAP, HDLC, LCP, NCP

Sponsor Source

MP (MultiPPP) is defined by IETF (http://www.ietf.org) .

Reference

http://www.javvin.com/protocol/rfc1990.pdf
The PPP Multilink Protocol (MP).
http://www.javvin.com/protocol/rfc1661.pdf
The Point-to-Point Protocol (PPP)

Protocol Name

PPP NCP: Point to Point Protocol Network Control Protocols

Protocol Description

The Network Control Protocol (NCP) phase in the PPP link connection process is used for establishing and configuring different network-layer protocols such as IP, IPX or AppleTalk.

After a NCP has reached the Opened state, PPP will carry the corresponding network-layer protocol packets. Any supported network-layer protocol packets received when the corresponding NCP is not in the Opened state MUST be silently discarded.

During this phase, link traffic consists of any possible combination of LCP, NCP, and network-layer protocol packets.

The most common layer 3 protocol negotiated is IP. The routers exchange IP Control Protocol (IPCP) messages negotiating options specific to the protocol. The corresponding network control protocol for IPv6 is IPv6CP.

IPCP negotiates two options: compression and IP address assignments. However, IPCP is also used to pass network related information such as primary and backup Windows Name Service (WINS) and Domain Name System (DNS) servers.

Protocol Structure

Network Control Protocols such as IPCP and IPv6CP use the same packet format as the Link Control Protocols.

Configuration Option format:

8	16	32bit
Type	Length	Configuration Option

Packet format:

8	16	32bit	Variable
Code	Identifier	Length	Data

- Code - The Code field is one octet and identifies the type of the packet.
- Identifier - The Identifier field is one octet and aids in matching requests and replies.
- Length - The Length field is two octets and indicates the length of the packet.
- Data - The Data field is zero or more octets. The format of the Data field is determined by the Code field.

Related protocols

PPP, NCP, IPCP, IPv6CP, LCP, IP, IPX, DECnet, AppleTalk

Sponsor Source

PPP Network Control Protocols are defined by IETF (http://www.ietf.org) .

Reference

http://www.javvin.com/protocol/rfc1661.pdf
The Point-to-Point Protocol (PPP)
http://www.javvin.com/protocol/rfc1332.pdf
The PPP Internet Protocol Control Protocol (IPCP).
http://www.javvin.com/protocol/rfc2472.pdf
IP Version 6 over PPP
http://www.javvin.com/protocol/rfc3241.pdf
Robust Header Compression (ROHC) over PPP.
http://www.javvin.com/protocol/rfc3544.pdf
IP Header Compression over PPP.

Protocol Name

PAP: Password Authentication Protocol

Protocol Description

The Password Authentication Protocol (PAP), a Link Control Protocol in the PPP suite, provides a simple method for the peer to establish its identity using a 2-way handshake. This is done only upon initial link establishment.

After the Link Establishment phase is complete, an ID/Password pair is repeatedly sent by the peer to the authenticator until authentication is acknowledged or the connection is terminated.

PAP is not a strong authentication method. Passwords are sent over the circuit in text format, and there is no protection from sniffing, playback or repeated trial and error attacks. The peer is in control of the frequency and timing of the attempts. Any implementations which include a stronger authentication method (such as CHAP) MUST offer to negotiate that method prior to PAP.

This authentication method is most appropriately used where a plaintext password must be available to simulate a login at a remote host. In such a use, this method provides a similar level of security to the usual user login at the remote host.

Protocol Structure

Configuration Option format for Password Authentication Protocol:

8	16	32bit
Type	Length	Authentication-Protocol

- Type - 3
- Length - 4
- Authentication-Protocol – C023 (Hex) for Password Authentication Protocol

Password Authentication Protocol (PAP) packet format:

8	16	32bits	Variable
Code	Identifier	Length	Data

- Code - The Code field is one octet and identifies the type of PAP packet. PAP Codes are assigned as follows:
  1 Authenticate-Request
  2 Authenticate-Ack
  3 Authenticate-Nak
- Identifier - The Identifier field is one octet and aids in matching requests and replies.
- Length - The Length field is two octets and indicates the length of the PAP packet including the Code, Identifier, Length and Data fields. Octets outside the range of the Length field should be treated as Data Link Layer padding and should be ignored on reception.
- Data - The Data field is zero or more octets. The format of the Data field is determined by the Code field.

Related protocols

PPP, CHAP, LCP, NCP

Sponsor Source

PAP is defined by IETF (http://www.ietf.org) RFC 1334; now replaced by RFC 1994.

Reference

http://www.javvin.com/protocol/rfc1334.pdf
PPP Authentication Protocols
http://www.javvin.com/protocol/rfc1994.pdf
PPP Challenge Handshake Authentication Protocol (CHAP)

Protocol Name

PPPoA: PPP over ATM AAL5

Protocol Description

PPPoA describes the use of ATM Adaptation Layer 5 (AAL5) for framing PPP encapsulated packets.

The Point-to-Point Protocol (PPP) provides a standard method for transporting multi-protocol datagrams over point-to-point links.

The ATM AAL5 protocol is designed to provide virtual connections between end stations attached to the same network. These connections offer a packet delivery service that includes error detection, but does not do error correction.

Most existing implementations of PPP use ISO 3309 HDLC as a basis for their framing.

When an ATM network is configured with point-to-point connections, PPP can use AAL5 as a framing mechanism.

The PPP layer treats the underlying ATM AAL5 layer service as a bit-synchronous point-to-point link. In this context, the PPP link corresponds to an ATM AAL5 virtual connection. The virtual connection MUST be full-duplex, point to point, and it MAY be either dedicated (i.e. permanent, set up by provisioning) or switched (set up on demand). LLC encapsulated PPP over AAL5 is the alternative technique to VC-multiplexed PPP over AAL5.

When transporting a PPP payload over AAL5, an implementation:

1. MUST support virtual circuit multiplexed PPP payloads as described in section 5 below by mutual configuration or negotiation of both end points. This technique is referred to as “VC-multiplexed PPP”.
2. MUST support LLC encapsulated PPP payloads on PVCs, as described in section 6 below by mutual configuration or negotiation of both end points. This technique is referred to as “LLC encapsulated PPP”.
3. For SVC set up, an implementation MUST negotiate using the Q.2931 [9] Annex C procedure, encoding the Broadband Lower Layer Interface (B-LLI) information element to signal either VC- multiplexed PPP or LLC encapsulated PPP.

Protocol Structure

Virtual Circuit Multiplexed PPP Over AAL5. The AAL5 PDU format is shown below:

AAL5 CPCS-PDU Format

1 byte	0-47bytes	1 byte	1 byte	2 bytes	4 bytes
CPCS-PDU	PAD	CPCS-UU	CPI	Length	CRC
			CPCS-PDU Trailer		

The AAL5 CPCS-PDU payload field is encoded as shown below:

1. LLC header: 2 bytes encoded to specify a source SAP and destination SAP of routed OSI PDU (values 0xFE 0xFE), followed by an Un-numbered Information (UI) frame type (value 0x03).
2. Network Layer Protocol IDentifier (NLPID) representing PPP, (value 0xCF).
3. The PPP protocol identifier field, which can be either 1 or 2 octets long.
4. Followed by the PPP information field.

Destination SAP	Source SAP	Frame type	LLC Header
NLPID = PPP			
Protocol ID	PPP Info	Padding	PPP Payload
PAD (0 – 47 bytes)			
CPCS-UU	CPI	Length	CRC	CPCS-PDU Trailer

Related protocols

PPP, 802.3, ATM

Sponsor Source

PPPoA is defined by IETF (http://www.ietf.org) RFC 2364.

Reference

http://www.javvin.com/protocol/rfc2364.pdf
PPP Over AAL5

Protocol Name

PPPoE: PPP over Ethernet

Protocol Description

PPP over Ethernet (PPPoE) provides the ability to connect a network of hosts over a simple bridging access device to a remote Access Concentrator. With this model, each host utilizes its own PPP stack and the user is presented with a familiar user interface. Access control, billing and type of service can be done on a per-user, rather than per-site, basis.

To provide a point-to-point connection over Ethernet, each PPP session must learn the Ethernet address of the remote peer, as well as establish a unique session identifier. PPPoE includes a discovery protocol that provides this.

PPPoE has two distinct stages. There is a Discovery stage and a PPP Session stage. When a Host wishes to initiate a PPPoE session, it must first perform Discovery to identify the Ethernet MAC address of the peer and establish a PPPoE SESSION_ID. While PPP defines a peer-to-peer relationship, Discovery is inherently a client-server relationship. In the Discovery process, a Host (the client) discovers an Access Concentrator (the server). Based on the network topology, there may be more than one Access Concentrator that the Host can communicate with. The Discovery stage allows the Host to discover all Access Concentrators and then select one. When Discovery completes successfully, both the Host and the selected Access Concentrator have the information they will use to build their point-to-point connection over Ethernet.

The Discovery stage remains stateless until a PPP session is established. Once a PPP session is established, both the Host and the Access Concentrator MUST allocate the resources for a PPP virtual interface.

Protocol Structure

The Ethernet payload for PPPoE is as follows:

4	8	16	32bit
Ver	Type	Code	Session-ID
Length			Payload

- VER – version of PPPOE MUST be set to 0x1.
- TYPE - MUST be set to 0x1.
- CODE - is defined below for the Discovery and PPP Session stages.
- SESSION_ID - is an unsigned value in network byte order. It's value is defined below for Discovery packets. The value is fixed for a given PPP session and, in fact, defines a PPP session along with the Ethernet SOURCE_ADDR and DESTINATION_ADDR. A value of 0xffff is reserved for future use and MUST NOT be used
- LENGTH - The value, in network byte order, indicates the length of the PPPoE payload. It does not include the length of the Ethernet or PPPoE headers.

Related protocols

PPP, 802.3

Sponsor Source

PPPoE is defined by IETF (http://www.ietf.org) RFC 2516.

Reference

http://www.javvin.com/protocol/rfc2516.pdf
A Method for Transmitting PPP Over Ethernet (PPPoE)

Other WAN Protocols

Protocol Name

Frame Relay: WAN Protocol for Internetworking

Protocol Description

Frame Relay is a WAN protocol for LAN internetworking which operates at the physical and data link layer to provide a fast and efficient method of transmitting information from a user device to another across multiple switches and routers.

Frame Relay is based on packet-switched technologies similar to x.25, which enables end stations to dynamically share the network medium and the available bandwidth. It employs the following two packet techniques: a) Variable-length packets and b) Statistical multiplexing. It does not guarantee data integrity and discard packets when there is network congestion. In reality, it still delivers data with high reliability.

The Frame Relay frame is transmitted to its destination through virtual circuits, which are logical paths from an originating point in the network to a destination point. Virtual circuits provide bidirectional communication paths from one terminal device to another and are uniquely identified by a data-link connection identifier (DLCI). A number of virtual circuits can be multiplexed into a single physical circuit for transmission across the network. This capability often can reduce the equipment and network complexity required to connect multiple terminal devices. A virtual circuit can pass through any number of intermediate switches located within the Frame Relay packet switched network.

There are permanent virtual circuits (PVCs) and switched virtual circuits (SVCs). PVCs are set up administratively by the network manager for a dedicated point-to-point connection; SVCs are set up on a call-by-call basis using the same signaling as for ISDN set up.

Due to its bandwidth efficiency and high reliability, Frame Relay offers an attractive alternative to both dedicated lines and X.25 networks for the inter-connecting of LANs through switches and routers.

Protocol Structure

The Frame Relay (LAPF Q.922 based) frame structure is as follows:

1byte	2 bytes	Variable	2 bytes	1byte
Flags	Address	Data	FCS	Flags

- Flags—Delimits the beginning and end of the frame. The value of this field is always the same and is represented either as the hexadecimal number 7E or as the binary number 01111110.
- Address—Contains the following information:

6	7	8	12	13	14	15	16bit
DLCI	C/R	E	DLCI	FECN	BECN	DE	EA

  - DLCI – Datalink Connection Identifier field represents the address of the frame and corresponds to a PVC.
  - C/R - Designates whether the frame is a command or response.
  - EA - Extended Address field signifies up to two additional bytes in the Frame Relay header, thus greatly expanding the number of possible addresses.
  - FECN - Forward Explicit Congestion Notification (see ECN below).
  - BECN - Backward Explicit Congestion Notification (see ECN below).
  - DE - Discard Eligibility.

- Data—Contains encapsulated upper-layer data. Each frame in this variable-length field includes a user data or payload field that will vary in length up to 16,000 octets. This field serves to transport the higher-layer protocol packet (PDU) through a Frame Relay network.
- Frame Check Sequence—Ensures the integrity of transmitted data. This value is computed by the source device and verified by the receiver to ensure integrity of transmission.

Frame Relay frames that conform to the LMI specifications consist of the fields as follows:

1byte	2 bytes	1byte	1byte	1byte	1byte
Flags	LMI DLCI	I-Indicator	Protocol Dis	Call Ref	M-Type
Information Elements (Variable)			FCS		Flags

- Flags—Delimits the beginning and end of the frame.
- LMI DLCI—Identifies the frame as an LMI frame instead of a basic Frame Relay frame. The LMI-specific DLCI value defined in the LMI consortium specification is DLCI = 1023.
- Unnumbered Information Indicator—Sets the poll/final bit to zero.
- Protocol Discriminator—Always contains a value indicating that the frame is an LMI frame.
- Call Reference—Always contains zeros. This field currently is not used for any purpose.
- Message Type—Labels the frame as one of the following message types:
  - Status-inquiry message—Allows a user device to inquire about the status of the network.
  - Status message—Responds to status-inquiry

messages. Status messages include keepalives and PVC status messages.

- Information Elements—Contains a variable number of individual information elements (IEs). IEs consist of the following fields:
  - IE Identifier—Uniquely identifies the IE.
  - IE Length—Indicates the length of the IE.
  - Data—Consists of 1 or more bytes containing encapsulated upper-layer data.
- Frame Check Sequence (FCS)—Ensures the integrity of transmitted data.

Related protocols

LAPD, ISDN, X.25, LAPF

Sponsor Source

Frame Relay is defined by ITU-T (http://www.itu.org), ANSI (http://www.ansi.org) in the ANSI T1.618 and ANSI t1.617.

Reference

http://www.cisco.com/univercd/cc/td/doc/cisintwk/ito_doc/frame.htm
Understand Frame Relay

Protocol Name

LAPF: Link Access Procedure for Frame Mode Services

Protocol Description

Link Access Procedure/Protocol for Frame Mode Services (LAPF) as defined in ITU Q.922, is an enhanced LAPD (Q.921) with congestion control capabilities for Frame Mode Services in the Frame Relay network. The core functions of LAPF are:

- Frame delimiting, alignment, and flag transparency
- Virtual circuit multiplexing and de-multiplexing
- Octet alignment: Integer number of octets before zero-bit insertion
- Checking minimum and maximum frame sizes
- Error detection, Sequence and non-duplication
- Congestion control

The purpose of LAPF is to convey data link service data units between DL-service users in the User Plane for frame mode bearer services across the ISDN user-network interface on B-, D- or H-channels. Frame mode bearer connections are established using either procedures specified in Recommendation Q.933 or (for permanent virtual circuits) by subscription. LAPF uses a physical layer service, and allows for statistical multiplexing of one or more frame mode bearer connections over a single ISDN B-, D- or H-channel by use of LAPF and compatible HDLC procedures.

LAPF is used in the Frame Relay network for end-to-end signaling.

Protocol Structure

LAPF is similar to LAPD and its address format is as follows:

6	7	8	12	13	14	15	16bit
DLCI	C/R	E	DLCI	FECN	BECN	DE	EA

- DLCI – Datalink Connection Identifier field represents the address of the frame and corresponds to a PVC.
- C/R - Designates whether the frame is a command or response.
- EA - Extended Address field signifies up to two additional bytes in the Frame Relay header, thus greatly expanding the number of possible addresses.
- FECN - Forward Explicit Congestion Notification (see ECN below).
- BECN - Backward Explicit Congestion Notification (see ECN below).
- DE - Discard Eligibility.

LAPF control field format:

Control field bits (Modulo 128)	8	7	6	5	4	3	2	1
I Format	N(S)							0
	N(R)							P/F
S Format	X	X	X	X	Su	Su	0	1
	N(R)							P/F
U Format	M	M	M	P/F	M	M	1	1

- N(S) - Transmitter send sequence number.
- N(R) - Transmitter receive sequence number.
- P/F - Poll bit when used as a command, final bit when used as a response.
- X - Reserved and set to 0.
- Su - Supervisory function bit.
- M - Modifier function bit.

Related protocols

LAPD, ISDN, X.25, Frame Relay

Sponsor Source

LAPF is defined by ITU-T (http://www.itu.org) Q.922: ISDN Data Link Layer Specification for Frame Mode Bearer Services

Reference

http://www.cisco.com/univercd/cc/td/doc/cisintwk/ito_doc/frame.htm
Understand Frame Relay

Protocol Name

HDLC: High Level Data Link Control

Protocol Description

The High Level Data Link Control (HDLC) protocol, an ISO data link layer protocol based on the IBM SDLC, ensures that data passed up to the next layer has been received exactly as transmitted (i.e. error free, without loss and in the correct order). Another important function of HDLC is flow control, which ensures that data is transmitted only as fast as the receiver can receive it. There are two distinct HDLC implementations: HDLC NRM (also known as SDLC) and HDLC Link Access Procedure Balanced (LAPB). The later is the more popular implementation. HDLC is part of the X.25 stack.

LAPB is a bit-oriented synchronous protocol that provides complete data transparency in a full-duplex point-to-point operation. It supports a peer-to-peer link in that neither end of the link plays the role of the permanent master station. HDLC NRM, on the other hand, has a permanent primary station with one or more secondary stations.

HDLC LAPB is a very efficient protocol, which requires a minimum of overhead to ensure flow control, error detection and recovery. If data is flowing in both directions (full duplex), the data frames themselves carry all the information required to ensure data integrity.

The concept of a frame window is used to send multiple frames before receiving confirmation that the first frame has been correctly received. This means that data can continue to flow in situations where there may be long “turn-around” time lags without stopping to wait for an acknowledgement. This kind of situation occurs, for instance in satellite communication.

There are three categories of frames:

- Information frames transport data across the link and may encapsulate the higher layers of the OSI architecture.
- Supervisory frames perform the flow control and error recovery functions.
- Unnumbered frames provide the link initialization and termination.

Protocol Structure

1 byte	1-2 bytes	1 byte	variable	2 bytes	1 byte
Flag	Address field	Control field	Information	FCS	Flag

- Flag - The value of the flag is always (0x7E).
- Address field - Defines the address of the secondary station which is sending the frame or the destination of the frame sent by the primary station. It contains Service Access Point (6bits), a Command/Response bit to indicate whether the frame relates to information frames (I-frames) being sent from the node or received by the node, and an address extension bit which is usually set to true to indicate that the address is of length one byte. When set to false it indicates an additional byte follows.
- Extended address - HDLC provides another type of extension to the basic format. The address field may be extended to more than one byte by agreement between the involved parties.
- Control field - Serves to identify the type of the frame. In addition, it includes sequence numbers, control features and error tracking according to the frame type.
- FCS - The Frame Check Sequence (FCS) enables a high level of physical error control by allowing the integrity of the transmitted frame data to be checked.

Related protocols

LAPB, X.25, Frame Relay, SDLC

Sponsor Source

HDLC is defined by ISO (http://www.iso.org).

Reference

http://www2.rad.com/networks/1994/hdlc/hdlc.htm
High Level Data Link Control

Protocol Name

LAPB: Link Access Procedure, Balanced

Protocol Description

Link Access Procedure, Balanced (LAPB) is a data link layer protocol used to manage communication and packet framing between data terminal equipment (DTE) and the data circuit-terminating equipment (DCE) devices in the X.25 protocol stack. LAPB, a bit-oriented protocol derived from HDLC, is actually the HDLC in BAC (Balanced Asynchronous Class) mode. LAPB makes sure that frames are error free and properly sequenced.

LAPB shares the same frame format, frame types, and field functions as SDLC and HDLC. Unlike either of these, however, LAPB is restricted to the Asynchronous Balanced Mode (ABM) transfer mode and is appropriate only for combined stations. Also, LAPB circuits can be established by either the DTE or DCE. The station initiating the call is determined to be the primary, and the responding station the secondary. Finally, LAPB use of the P/F bit is somewhat different from that of the other protocols.

In LAPB, since there is no master/slave relationship, the sender uses the Poll bit to insist on an immediate response. In the response frame this same bit becomes the receivers Final bit. The receiver always turns on the Final bit in its response to a command from the sender with the Poll bit set. The P/F bit is generally used when either end becomes unsure about proper frame sequencing because of a possible missing acknowledgement, and it is necessary to re-establish a point of reference.

LAPB's Frame Types:

- I-Frames (Information frames): Carry upper-layer information and some control information. I-frame functions include sequencing, flow control, and error detection and recovery. I-frames carry send and receive sequence numbers.
- S-Frames (Supervisory Frames): Carry control information. S-frame functions include requesting and suspending transmissions, reporting on status, and acknowledging the receipt of I-frames. S-frames carry only receive sequence numbers.
- U-Frames (Unnumbered Frames): Carry control information. U-frame functions include link setup and disconnection, as well as error reporting. U-frames carry no sequence numbers.

Protocol Structure

The format of an LAPB frame is as follows:

1 byte	1 byte	1-2 bytes	variable	2 byte	1 byte
Flag	Address field	Control field	Information	FCS	Flag

- Flag - The value of the flag is always (0x7E). In order to ensure that the bit pattern of the frame delimiter flag does not appear in the data field of the frame (and therefore cause frame misalignment), a technique known as Bit Stuffing is used by both the transmitter and the receiver.
- Address field - In LAPB, the address field has no meaning since the protocol works in a point to point mode and the DTE network address is represented in the layer 3 packets.
- Control field – Serves to identify the type of the frame. In addition, it includes sequence numbers, control features and error tracking according to the frame type.
- Modes of operation - LAPB works in the Asynchronous Balanced Mode (ABM). This mode is totally balanced (i.e., no master/slave relationship) and is signified by the SABM(E) frame. Each station may initialize, supervise, recover from errors, and send frames at any time. The DTE and DCE are treated as equals.
- FCS - The Frame Check Sequence enables a high level of physical error control by allowing the integrity of the transmitted frame data to be checked.
- Window size - LAPB supports an extended window size (modulo 128) where the number of possible outstanding frames for acknowledgement is raised from 8 to 128.

Related protocols

LAPD, ISDN, X.25, Frame Relay, HDLC, SDLC

Sponsor Source

LAPB is defined by ISO (http://www.iso.org).

Reference

http://www.cisco.com/univercd/cc/td/doc/cisintwk/ito_doc/x25.htm
X.25
http://www2.rad.com/networks/1994/hdlc/hdlc.htm
High Level Data Link Control

Protocol Name

X.25: ISO/ITU-T Protocol for WAN Communications

Protocol Description

X.25, an ISO and ITU-T protocol for WAN communications, is a packet switched data network protocol which defines the exchange of data as well as control information between a user device, called Data Terminal Equipment (DTE) and a network node, called Data Circuit Terminating Equipment (DCE).

X.25 is designed to operate effectively regardless of the type of systems connected to the network. X.25 is typically used in the packet-switched networks (PSNs) of common carriers, such as the telephone companies. Subscribers are charged based on their use of the network. X.25 utilizes a Connection-Oriented service which insures that packets are transmitted in order.

X.25 sessions are established when one DTE device contacts another to request a communication session. The DTE device that receives the request can either accept or refuse the connection. If the request is accepted, the two systems begin full-duplex information transfer. Either DTE device can terminate the connection. After the session is terminated, any further communication requires the establishment of a new session. X.25 uses virtual circuits for packets communications. Both switched and permanent virtual circuits are used.

The X.25 protocol suite comes with three levels based on the first three layers of the OSI seven layers architecture.

The Physical Level: describes the interface with the physical environment. There are three protocols in this group: 1) X.21 interface operates over eight interchange circuits; 2) X.21bis defines the analogue interface to allow access to the digital circuit switched network using an analogue circuit; 3) V.24 provides procedures which enable the DTE to operate over a leased analogue circuit connecting it to a packet switching node or concentrator.

The Link Level is responsible for reliable communication between the DTE and the DCE. There are four protocols in this group: 1) LAPB, derived from HDLC and the most commonly used, has all the characteristics of HDLC and also enables the formatition of a logical link connection; 2) Link Access Protocol (LAP) is an earlier version of LAPB and is seldom used today; and 3) LAPD, derived from LAPB and used for ISDN, enables data transmission between DTEs through D channel, especially between a DTE and an ISDN node; 4) Logical Link Control (LLC), an IEEE 802 LAN protocol, enables X.25 packets to be transmitted through a LAN channel.

The Packet Layer Protocol (PLP): describes the data transfer protocol in the packet switched network at the network layer (layer 3). PLP manages packet exchanges between DTE devices across virtual circuits. PLPs also can run over Logical Link Control 2 implementations on LANs as well as over ISDN interfaces running LAPD. The PLP operates in five distinct modes: call setup, data transfer, idle, call clearing, and restarting.

- Call setup mode is used to establish SVCs between DTE devices.
- Data transfer mode is used for transferring data between two DTE devices across a virtual circuit.
- Idle mode is used when a virtual circuit is established but data transfer is not occurring.
- Call clearing mode is used to end communication sessions between DTE devices and to terminate SVCs.
- Restarting mode is used to synchronize transmission between a DTE device and a locally connected DCE device.

X.75 is the signaling protocol for X.25, which defines the signaling system between two PDNs. X.75 is essentially a Network to Network Interface (NNI).

We focus on the X.25 PLP; other protocols will be discussed separately.

Protocol Structure

X.25 PLP has many control messages. The control packet as well as all X.25 packets begins with a 3-byte header. Bytes 1,2 contain the Group and the Channel fields that together form a 12 bit virtual circuit number. The additional information for each message is different.

1. Control Packet

1	2	3	4	8	16	23	24bit
0	0	0	1	Group	Channel	Type	C
Additional Information (Variable)							

2. The additional information of the Call Request Packet is as follows:

4bits	4bits	Variable	2bits	6bits	Variable
Length Calling address	Length Called address	Calling & Called address	00	Facility length	Facilities
Data (Variable)					

Other Control Packets are:

- The CALL ACCEPTED packet is sent by the callee DTE if it accepts the call.
- The CLEAR REQUEST is sent for various reasons. The fourth byte of the packet tells why the connection is being cleared. It is acknowledged by a CLEAR REQUEST CONFIRMATION packet.
- The INTERRUPT packet allows a short (32 bytes) signal to be sent out of sequence. It is acknowledged by an INTERRUPT CONFIRMATION packet.

- The RECEIVE READY (RR) packet is used to send separate acknowledgments where there is no reverse traffic. The ppp field (three first bits of the type field) tells which packet is expected next.
- The RECEIVE NOT READY (RNR) packet allows a DTE to tell the other side to stop sending packets to it for a while.
- The REJECT packet allows a DTE to request retransmission of a series of packets. The ppp field gives the first sequence number desired.
- The RESET and RESTART packets are used to recover from varying degrees of trouble. Both are acknowledged by RESET CONFIRMATION and RESTART CONFIRMATION respectively.
- The DIAGNOSTIC packet is also provided, to allow the network to inform the user of problems.

3. The format of the data packet is as follows:

1	2	4	8	16	23	24	31	32bit
Q	D	Mod-ulo	Group	Channel	Piggy-back	M	sequence	C
Data (Variable)								

Related protocols

LAPB, X.25, Frame Relay, HDLC, ISDN, LLC, LAPD

Sponsor Source

X.25 protocol stack is defined by ISO (http://www.iso.org) and ITU-T (http://www.itu.org)

Reference

http://www.cisco.com/univercd/cc/td/doc/cisintwk/ito_doc/x25.htm
X.25 Overview
http://www2.rad.com/networks/1996/x25/x25.htm
X.25 Protocols

Local Area Network and LAN Protocols

Description

Local Area Network (LAN) is a data communications network connecting terminals, computers and printers within a building or other geographically limited areas. These devices may be connected through wired cables or wireless links. Ethernet, Token Ring and Wireless LAN using IEEE 802.11 are examples of standard LAN technologies.

Ethernet is by far the most commonly used LAN technology. Token Ring technology is still used by some companies. FDDI is sometimes used as a backbone LAN interconnecting Ethernet or Token Ring LANs. WLAN using IEEE 802.11 technologies is rapidly becoming the new leading LAN technology because of its mobility and easy to use features.

Local Area Networks can be interconnected using Wide Area Network (WAN) or Metropolitan Area Network (MAN) technologies. The common WAN technologies include TCP/IP, ATM, Frame Relay etc. The common MAN technologies include SMDS and 10 Gigabit Ethernet.

LANs are traditionally used to connect a group of people who are in the same local area. However, working groups are becoming more geographically distributed in today's working environment. In these cases, virtual LAN (VLAN) technologies are defined for people in different places to share the same networking resource.

Local Area Network protocols are mostly at the data link layer (layer 2). IEEE is the leading organization defining LAN standards. There are some vendor specific LAN protocols such as Novell Netware, AppleTalk etc. which are discussed in seperate sections.

Key Protocols

Ethernet	Ethernet LAN protocols as defined in IEEE 802.3 suite
	Fast Ethernet: Ethernet LAN at data rate 100Mbps (IEEE 802.3u)
	Gigabit Ethernet: Ethernet at data rate 1000Mbps (IEEE 802.3z, 802.3ab)
	10Gigabit Ethernet: Ethernet at data rate 10 Gbps (IEEE 802.3ae)
VLAN	802.1Q: Virtual LAN Bridging Switching Protocol
	GARP: Generic Attribute Registration Protocol (802.1P)
	GMRP: GARP Multicast Registration Protocol (802.1P)
	GVRP: GARP VLAN Registration Protocol (802.1P, 802.1Q)
	IEEE 802.1P: LAN Layer 2 QoS/CoS Protocol
WLAN	Wireless LAN in IEEE 802.11, 802,11a, 802.11b, 802.11g
	IEEE 802.1X: WLAN Authentication & Key Management
	IEEE 802.15: Bluetooth for Wireless Personal Area Network (WPAN)
Token Ring	Token Ring: IEEE 802.5 LAN protocol
FDDI	FDDI: Fiber Distributed Data Interface
Others	LLC: Logic Link Control (IEEE 802.2)
	SNAP: SubNetwork Access Protocol
	STP: Spanning Tree Protocol (IEEE 802.1D)

Ethernet Protocols

Protocol Name

Ethernet: IEEE 802.3 Local Area Network protocols

Protocol Description

Ethernet protocols refer to the family of local-area networks (LAN) covered by a group of IEEE 802.3 standards. In the Ethernet standard, there are two modes of operation: half-duplex and full-duplex. In the half-duplex mode, data are transmitted using the popular Carrier-Sense Multiple Access/Collision Detection (CSMA/CD) protocol on a shared medium. The main disadvantages of the half-duplex are the efficiency and distance limitation, in which the link distance is limited by the minimum MAC frame size. This restriction reduces the efficiency drastically for high-rate transmission. Therefore, the carrier extension technique is used to ensure the minimum frame size of 512 bytes in Gigabit Ethernet to achieve a reasonable link distance.

Four data rates are currently defined for operation over optical fiber and twisted-pair cables:

- 10 Mbps—10Base-T Ethernet (802.3)
- 100 Mbps—Fast Ethernet (802.3u)
- 1000 Mbps—Gigabit Ethernet (802.3z)
- 10-Gigabit Ethernet - IEEE 802.3ae

In this document, we discuss the general aspects of the Ethernet. The specific issues on fast Ethernet, Gigabit and 10 Gigabit Ethernet will be discussed in separate documents.

The Ethernet system consists of three basic elements: 1) the physical medium used to carry Ethernet signals between computers, 2) a set of medium access control rules embedded in each Ethernet interface that allows multiple computers to fairly arbitrate access to the shared Ethernet channel, and 3) an Ethernet frame that consists of a standardized set of bits used to carry data over the system.

As with all IEEE 802 protocols, the ISO data link layer is divided into two IEEE 802 sublayers, the Media Access Control (MAC) sub-layer and the MAC-client sublayer. The IEEE 802.3 physical layer corresponds to the ISO physical layer.

The MAC sublayer has two primary responsibilities:

- Data encapsulation, including frame assembly before transmission, and frame parsing/error detection during and after reception
- Media access control, including initiation of frame transmission and recovery from transmission failure

The MAC-client sublayer may be one of the following:

- Logical Link Control (LLC), which provides the interface between the Ethernet MAC and the upper layers in the protocol stack of the end station. The LLC sublayer is defined by IEEE 802.2 standards.
- Bridge entity, which provides LAN-to-LAN interfaces between LANs that use the same protocol (for example, Ethernet to Ethernet) and also between different protocols (for example, Ethernet to Token Ring). Bridge entities are defined by IEEE 802.1 standards.

Each Ethernet-equipped computer operates independently of all other stations on the network: there is no central controller. All stations attached to an Ethernet are connected to a shared signaling system, also called the medium. To send data a station first listens to the channel and, when the channel is idle then transmits its data in the form of an Ethernet frame, or packet.

After each frame transmission, all stations on the network must contend equally for the next frame transmission opportunity. Access to the shared channel is determined by the medium access control (MAC) mechanism embedded in the Ethernet interface located in each station. The medium access control mechanism is based on a system called Carrier Sense Multiple Access with Collision Detection (CSMA/CD).

As each Ethernet frame is sent onto the shared signal channel, all Ethernet interfaces look at the destination address. If the destination address of the frame matches with the interface address, the frame will be read entirely and be delivered to the networking software running on that computer. All other network interfaces will stop reading the frame when they discover that the destination address does not match their own address.

When it comes to how signals flow over the set of media segments that make up an Ethernet system, it helps to understand the topology of the system. The signal topology of the Ethernet is also known as the logical topology, to distinguish it from the actual physical layout of the media cables. The logical topology of an Ethernet provides a single channel (or bus) that carries Ethernet signals to all stations.

Multiple Ethernet segments can be linked together to form a larger Ethernet LAN using a signal amplifying and retiming device called a repeater. Through the use of repeaters, a given Ethernet system of multiple segments can grow as a “non-rooted branching tree.” “Non-rooted” means that the resulting system of linked segments may grow in any direction, and does not have a specific root segment. Most importantly, segments must never be connected in a loop. Every segment in the system must have two ends, since the Ethernet system will not operate correctly in the presence of loop paths.

Even though the media segments may be physically connected in a star pattern, with multiple segments attached to a repeater, the logical topology is still that of a single Ethernet channel that carries signals to all stations.

Protocol Structure

The basic IEEE 802.3 MAC Data Frame for 10/100Mbps Ethernet:

7	1	6	6	2	46-1500bytes	4bytes
Pre	SFD	DA	SA	Length Type	Data unit + pad	FCS

- Preamble (Pre)— 7 bytes. The PRE is an alternating pattern of ones and zeros that tells receiving stations that a frame is coming, and that provides a means to synchronize the frame-reception portions of receiving physical layers with the incoming bit stream.
- Start-of-frame delimiter (SFD)—1 byte. The SOF is an alternating pattern of ones and zeros, ending with two consecutive 1-bits indicating that the next bit is the left-most bit in the left-most byte of the destination address.
- Destination address (DA)— 6 bytes. The DA field identifies which station(s) should receive the frame.
- Source addresses (SA)— 6 bytes. The SA field identifies the sending station.
- Length/Type— 2 bytes. This field indicates either the number of MAC-client data bytes that are contained in the data field of the frame, or the frame type ID if the frame is assembled using an optional format.
- Data—Is a sequence of n bytes (46=< n =<1500) of any value. The total frame minimum is 64bytes.
- Frame check sequence (FCS)— 4 bytes. This sequence contains a 32-bit cyclic redundancy check (CRC) value, which is created by the sending MAC and is recalculated by the receiving MAC to check for damaged frames.

MAC Frame with Gigabit Carrier Extension:

1000Base-X has a minimum frame size of 416bytes, and 1000Base-T has a minimum frame size of 520bytes. An extension field is used to fill the frames that are shorter than the minimum length.

7	1	6	6	2	46=< n =<1500	4bytes	Variable
Pre	SFD	DA	SA	Length Type	Data unit + pad	FCS	Ext

Related protocols

IEEE 802.3, 802.3u, 802.3z, 802.3ab, 802.2, 802.1, 802.3ae, 802.1D, 802.1G, 802.1Q, 802.1p

Sponsor Source

Ethernet standards are defined by IEEE (http://www.ieee.org) in 802.3 specifications.

Reference

http://standards.ieee.org/getieee802/download/802.3-2002.pdf
Carrier sense multiple access with collision detection (CSMA/CD) access method and physical layer specification.
http://www.cisco.com/univercd/cc/td/doc/cisintwk/ito_doc/ethernet.htm
Ethernet Technologies
http://www.cisco.com/warp/public/cc/techno/media/lan/gig/tech/gigbt_tc.htm
Introduction to gigabit Ethernet

Protocol Name

Fast Ethernet: 100Mbps Ethernet (IEEE 802.3u)

Protocol Description

Fast Ethernet (100BASE-T) offers a speed increase ten times that of the 10BaseT Ethernet specification, while preserving such qualities as frame format, MAC mechanisms, and MTU. Such similarities allow the use of existing 10BaseT applications and network management tools on Fast Ethernet networks. Officially, the 100BASE-T standard is IEEE 802.3u.

Like Ethernet, 100BASE-T is based on the CSMA/CD LAN access method. There are several different cabling schemes that can be used with 100BASE-T, including:

100BASE-TX: two pairs of high-quality twisted-pair wires
100BASE-T4: four pairs of normal-quality twisted-pair wires
100BASE-FX: fiber optic cables

The Fast Ethernet specifications include mechanisms for Auto-Negotiation of the media speed. This makes it possible for vendors to provide dual-speed Ethernet interfaces that can be installed and run at either 10-Mbps or 100-Mbps automatically.

The IEEE identifiers include three pieces of information. The first item, “100”, stands for the media speed of 100-Mbps. The “BASE” stands for “baseband,” which is a type of signaling. Baseband signaling simply means that Ethernet signals are the only signals carried over the media system.

The third part of the identifier provides an indication of the segment type. The “T4” segment type is a twisted-pair segment that uses four pairs of telephone-grade twisted-pair wires. The “TX” segment type is a twisted-pair segment that uses two pairs of wires and is based on the data grade twisted-pair physical medium standard developed by ANSI. The “FX” segment type is a fiber optic link segment based on the fiber optic physical medium standard developed by ANSI and uses two strands of fiber cable. The TX and FX medium standards are collectively known as 100BASE-X.

The 100BASE-TX and 100BASE-FX media standards used in Fast Ethernet are both adopted from physical media standards first developed by ANSI, the American National Standards Institute. The ANSI physical media standards were originally developed for the Fiber Distributed Data Interface (FDDI) LAN standard (ANSI standard X3T9.5), and are widely used in FDDI LANs.

Protocol Structure

Fast Ethernet has a minimum frame of 64 bytes and maximum up to 1518bytes, just as for Ethernet 802.3.

7	1	6	6	2	46=< n =<1500	4bytes
Pre	SFD	DA	SA	Length Type	Data unit + pad	FCS

- Preamble (Pre)— 7 bytes. The PRE is an alternating pattern of ones and zeros that tells receiving stations that a frame is coming, and that provides a means to synchronize the frame-reception portions of receiving physical layers with the incoming bit stream.
- Start-of-frame delimiter (SFD)—1 byte. The SOF is an alternating pattern of ones and zeros, ending with two consecutive 1-bits indicating that the next bit is the left-most bit in the left-most byte of the destination address.
- Destination address (DA)— 6 bytes. The DA field identifies which station(s) should receive the frame..
- Source addresses (SA)— 6 bytes. The SA field identifies the sending station.
- Length/Type— 2 bytes. This field indicates either the number of MAC-client data bytes that are contained in the data field of the frame, or the frame type ID if the frame is assembled using an optional format.
- Data—Is a sequence of n bytes (46=< n =<1500) of any value. The total frame minimum is 64bytes.
- Frame check sequence (FCS)— 4 bytes. This sequence contains a 32-bit cyclic redundancy check (CRC) value, which is created by the sending MAC and is recalculated by the receiving MAC to check for damaged frames.

Related protocols

IEEE 802.3, 802.3u, 802.3z, 802.3ab, 802.2, 802.1, 802.3ae, 802.1D, 802.1G, 802.1Q, 802.1p, 802.1w

Sponsor Source

Fast Ethernet standard is defined by IEEE (http://www.ieee.org) in 802.3u.

Reference

http://www.ethermanage.com/ethernet/descript-100quickref.html
Fast Ethernet Quick Guide

Protocol Name

Gigabit (1000 Mbps) Ethernet: IEEE 802.3z (1000Base-X) and 802.3ab (1000Base-T) and GBIC

Protocol Description

Ethernet protocols refer to the family of local-area network (LAN) covered by the IEEE 802.3 standard. The Gigabit Ethernet protocol is based on the Ethernet protocol but has tenfold speed increase over Fast Ethernet, using shorter frames with carrier Extension. It is published as the IEEE 802.3z and 802.3ab supplements to the IEEE 802.3 base standards.

Carrier Extension is a simple solution, but it wastes bandwidth. Packet Bursting is “Carrier Extension plus a burst of packets”. Burst mode is a feature that allows a MAC to send a short sequence (a burst) of frames equal to approximately 5.4 maximum-length frames without having to relinquish control of the medium.

The Gigabit Ethernet standards are fully compatible with Ethernet and Fast Ethernet installations. They retain Carrier Sense Multiple Access/ Collision Detection (CSMA/CD) as the access method. Full-duplex as well as half duplex modes of operation are supported, as are single-mode and multi mode fiber and short-haul coaxial cable, and twisted pair cables. The Gigabit Ethernet architecture is displayed in the following figure:

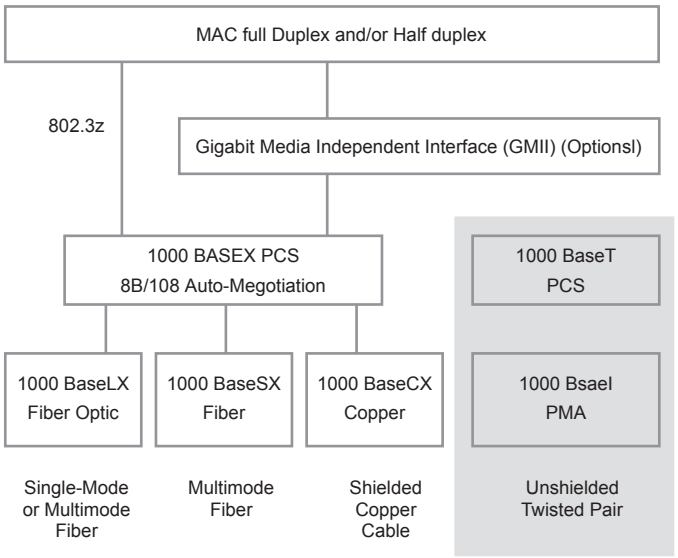


Figure 2-11: Gigabit Ethernet Protocol Stack

IEEE 802.3z defines the Gigabit Ethernet over fiber and cable, and has the physical media standard 1000Base-X (1000BaseSX – short wave covering up to 500m, and 1000BaseLX – long wave covers up to 5km). The IEEE 802.3ab defines the Gigabit Ethernet over the unshielded twisted pair wire (1000Base-T covers up to 75m).

The Gigabit interface converter (GBIC) allows network managers to configure each gigabit port on a port-by-port basis for short-wave (SX), long-wave (LX), long-haul (LH), and copper physical interfaces (CX). LH GBICs extend the single-mode fiber distance from the standard 5 km to 10 km.

Protocol Structure

1000Base-X has a minimum frame size of 416bytes, and 1000Base-T has a minimum frame size of 520bytes. An extension field is used to fill the frames that are shorter than the minimum length.

7	1	6	6	2	46=< n =<1500	4bytes	Variable
Pre	SFD	DA	SA	Length Type	Data unit + pad	FCS	Ext

- Preamble (Pre)— 7 bytes. The Pre is an alternating pattern of ones and zeros that tells receiving stations that a frame is coming, and that provides a means to synchronize the frame-reception portions of receiving physical layers with the incoming bit stream.
- Start-of-frame delimiter (SFD)—1 byte. The SOF is an alternating pattern of ones and zeros, ending with two consecutive 1-bits indicating that the next bit is the left-most bit in the left-most byte of the destination address.
- Destination address (DA)— 6 bytes. The DA field identifies which station(s) should receive the frame..
- Source addresses (SA)— 6 bytes. The SA field identifies the sending station.
- Length/Type— 2 bytes. This field indicates either the number of MAC-client data bytes that are contained in the data field of the frame, or the frame type ID if the frame is assembled using an optional format.
- Data—Is a sequence of n bytes (46=< n =<1500) of any value. The total frame minimum is 64bytes.
- Frame check sequence (FCS)— 4 bytes. This sequence contains a 32-bit cyclic redundancy check (CRC) value, which is created by the sending MAC and is recalculated by the receiving MAC to check for damaged frames.
- Ext – Extension, which is a non-data variable extension field for frames that are shorter than the minimum length.

Packet Bursting Mode:

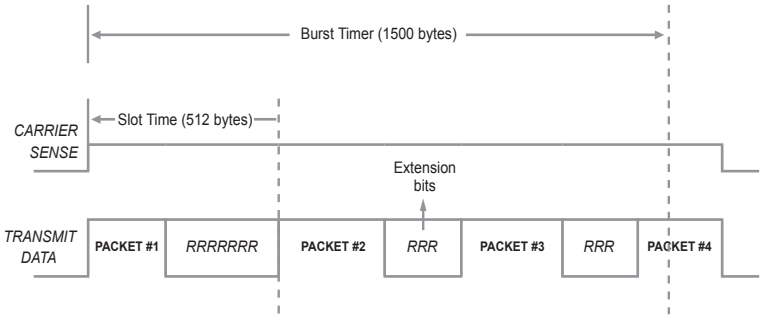


Figure 2-12: Packet Bursting Mode in Gigabit Ethernet

Related protocols

IEEE 802.3, 802.3u, 802.3z, 802.3ab, 802.2, 802.1, 802.3ae, 802.1D, 802.1G, 802.1Q, 802.1p, 802.1w

Sponsor Source

Gigabit Ethernet standards are defined by IEEE (http://www.ieee.org) 802.3z (1000BaseX0 and 802.3ab(1000BaseT).

Reference

http://www.cisco.com/univercd/cc/td/doc/cisintwk/ito_doc/ethernet.htm
Ethernet Technologies
http://www.cis.ohio-state.edu/~jain/cis788-97/ftp/gigabit_ethernet/index.htm
Gigabit Ethernet
http://www.cis.ohio-state.edu/~jain/refs/gbe_refs.htm
Links and reference regarding Ethernet

Protocol Name

10 Gigabit Ethernet: The Ethernet Protocol IEEE 802.3ae for LAN, WAN and MAN

Protocol Description

10-Gigabit Ethernet, standardized in IEEE 802.3ae, offers data speeds up to 10 billion bits per second. Built on the Ethernet technology used in most of today's local area networks (LANs), it offers similar benefits to those of the preceding Ethernet standard. 10-Gigabit Ethernet is used to interconnect local area networks (LANs), wide area networks (WANs), and metropolitan area networks (MANs). 10-Gigabit Ethernet uses the familiar IEEE 802.3 Ethernet media access control (MAC) protocol and its frame format and size. However, it supports full duplex but not half-duplex mode and only functions over optical fiber. Therefore, it does not need the carrier-sensing multiple-access with Collision Detection (CSMA/CD) protocol used in other Ethernet standards. The 10 Gigabit Ethernet architecture is displayed as follows:

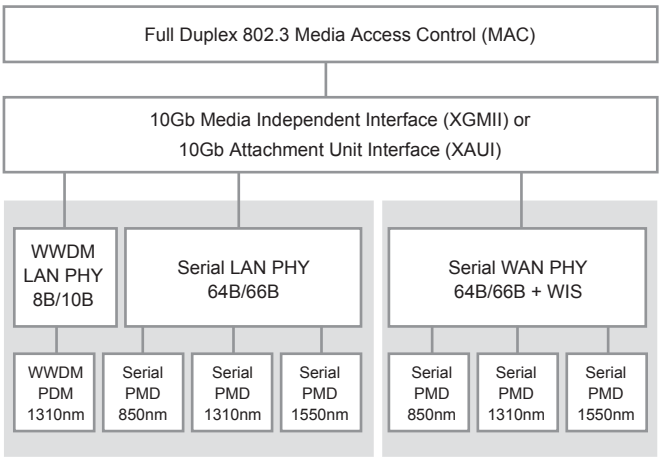


Figure 2-13: 10 Gigabit Ethernet Architecture

The 10 Gigabit specifications, contained in the IEEE 802.3ae supplement to the 802.3 standard, provides support to extend the 802.3 protocol and MAC specification to an operating speed of 10 Gb/s. In addition to the data rate of 10 Gb/s, 10-Gigabit Ethernet is able to accommodate slower date rates such as 9.584640 Gb/s (OC-192), through its "WAN interface sublayer" (WIS) which allows 10 Gigabit Ethernet equipment to be compatible with the Synchronous Optical Network (SONET) STS-192c transmission format.

The 10GBASE-SR and 10GBASE-SW media types are designed for use over short wavelength (850 nm) multimode fiber (MMF), which covers a fiber distance from 2 meters to 300 meters.. The 10GBASE-SR media type is designed for use over dark fiber, meaning a fiber optic cable that is not in use and that is not connected to any other equipment. The 10GBASE-SW media type is designed to connect to SONET equipment, which is typically used to provide long distance data communications.

The 10GBASE-LR and 10GBASE-LW media types are designed for use over long wavelength (1310 nm) single-mode fiber (SMF), which covers a fiber distance from 2 meters to 10 kilometers (32,808 feet). The 10GBASE-LR media type is designed for use over dark fiber, while the 10GBASE-LW media type is designed to connect to SONET equipment.

The 10GBASE-ER and 10GBASE-EW media types are designed for use over extra long wavelength (1550 nm) single-mode fiber (SMF), which covers a fiber distance from 2 meters up to 40 kilometers (131,233 feet). The 10GBASE-ER media type is designed for use over dark fiber, while the 10GBASE-EW media type is designed to connect to SONET equipment.

Finally, there is a 10GBASE-LX4 media type, which uses wave division multiplexing technology to send signals over four wavelengths of light carried over a single pair of fiber optic cables. The 10GBASE-LX4 system is designed to operate at 1310 nm over multi-mode or single-mode dark fiber. The design goal for this media system is from 2 meters up to 300 meters over multimode fiber or from 2 meters up to 10 kilometers over single-mode fiber.

Protocol Structure

10 gigabit Ethernet has a minimum frame of 64 bytes and maximum up to 1518bytes, as for the Ethernet 802.3.

7	1	6	6	2	46=< n =<1500	4bytes
Pre	SFD	DA	SA	Length Type	Data unit + pad	FCS

- Preamble (Pre)— 7 bytes. The Pre is an alternating pattern of ones and zeros that tells receiving stations that a frame is coming, and that provides a means to synchronize the frame-reception portions of receiving physical layers with the incoming bit stream.
- Start-of-frame delimiter (SFD)—1 byte. The SFD is an alternating pattern of ones and zeros, ending with two consecutive 1-bits indicating that the next bit is the left-most bit in the left-most byte of the destination address.
- Destination address (DA)— 6 bytes. The DA field identifies which station(s) should receive the frame..
- Source address (SA)— 6 bytes. The SA field identifies the sending station.
- Length/Type— 2 bytes. This field indicates either the number of MAC-client data bytes that are contained in the data field of the frame, or the frame type ID if the frame is assembled using an optional format.

- Data—Is a sequence of n bytes (46=< n =<1500) of any value. The total frame minimum is 64bytes.
- Frame check sequence (FCS)— 4 bytes. This sequence contains a 32-bit cyclic redundancy check (CRC) value, which is created by the sending MAC and is recalculated by the receiving MAC to check for damaged frames.

Related protocols

IEEE 802.3, 802.3u, 802.3z, 802.3ab, 802.2, 802.1, 802.3ae, 802.1D, 802.1G, 802.1Q, 802.1p, 802.1w

Sponsor Source

10 Gigabit Ethernet standard is defined by IEEE (http://www.ieee.org) 802.3ae.

Reference

http://www.10gea.org/10GEA%20White%20Paper_0502.pdf
10 Gigabit Ethernet Technology White Paper
http://www.intel.com/network/connectivity/resources/doc_library/white_papers/pro10gbe_lr_sa_wp.pdf
10 Gigabit Ethernet technology Overview

Virtual LAN Protocols

Protocol Name

VLAN: Virtual Local Area Network and the IEEE 802.1Q

Protocol Description

Virtual LAN (VLAN) is a group of devices on one or more LANs that are configured so that they can communicate as if they were attached to the same wire, when in fact they are located on a number of different LAN segments. Because VLANs are based on logical instead of physical connections, they are very flexible for user/host management, bandwidth allocation and resource optimization.

There are the following types of Virtual LANs:

1. Port-Based VLAN: Each physical switch port is configured with an access list specifying membership in a set of VLANs.
2. MAC-based VLAN: A switch is configured with an access list mapping individual MAC addresses to VLAN membership.
3. Protocol-based VLAN: A switch is configured with a list of mapping layer 3 protocol types to VLAN membership – thereby filtering IP traffic from nearby end-stations using a particular protocol such as IPX.
4. ATM VLAN – uses LAN Emulation (LANE) protocol to map Ethernet packets into ATM cells and deliver them to their destination by converting an Ethernet MAC address into an ATM address.

The IEEE 802.1Q specification establishes a standard method for tagging Ethernet frames with VLAN membership information. The IEEE 802.1Q standard defines the operation of VLAN Bridges that permit the definition, operation and administration of Virtual LAN topologies within a Bridged LAN infrastructure. The 802.1Q standard is intended to address the problem of how to break large networks into smaller parts so broadcast and multicast traffic will not grab more bandwidth than necessary. The standard also helps provide a higher level of security between segments of internal networks.

The key for the IEEE 802.1Q to perform the above functions is in its tags. 802.1Q-compliant switch ports can be configured to transmit tagged or untagged frames. A tag field containing VLAN (and/or 802.1p priority) information can be inserted into an Ethernet frame. If a port has an 802.1Q-compliant device attached (such as another switch), these tagged frames can carry VLAN membership information between switches, thus letting a VLAN span multiple switches. However, it is important to ensure ports with non-802.1Q-compliant devices attached are configured to transmit untagged frames. Many NICs for PCs and printers are not 802.1Q-compliant. If they receive a tagged frame, they will not understand the VLAN tag and will drop the frame. Also, the maximum legal Ethernet frame size for tagged frames was increased in 802.1Q (and its companion, 802.3ac) from 1,518 to 1,522 bytes. This could cause network interface cards and older switches to drop tagged frames as “oversized.”

Protocol Structure

IEEE 802.1Q Tagged Frame for Ethernet:

7	1	6	6	2	2	2	42-1496	4bytes
Pre-amble	SFD	DA	SA	TPID	TCI	Length Type	Data	CRC

- Preamble (Pre)— 7 bytes. The Pre is an alternating pattern of ones and zeros that tells receiving stations that a frame is coming, and that provides a means to synchronize the frame-reception portions of receiving physical layers with the incoming bit stream.
- Start-of-frame delimiter (SFD)—1 byte. The SOF is an alternating pattern of ones and zeros, ending with two consecutive 1-bits indicating that the next bit is the left-most bit in the left-most byte of the destination address.
- Destination address (DA)— 6 bytes. The DA field identifies which station(s) should receive the frame.
- Source addresses (SA)— 6 bytes. The SA field identifies the sending station.
- TPID - defined value of 8100 in hex. When a frame has the EtherType equal to 8100, this frame carries the tag IEEE 802.1Q / 802.1P.
- TCI – Tag Control Information field including user priority, Canonical format indicator and VLAN ID.

3bits	1bit	12bits
User Priority	CFI	Bits of VLAN ID (VIDI) to identify possible VLANs

- User Priority : Defines user priority, giving eight (2^3) priority levels. IEEE 802.1P defines the operation for these 3 user priority bits.
- CFI : Canonical Format Indicator is always set to zero for Ethernet switches. CFI is used for compatibility reason between Ethernet type network and Token Ring type network. If a frame received at an Ethernet port has a CFI set to 1, then that frame should not be forwarded as it is to an untagged port.
- VID : VLAN ID is the identification of the VLAN, which is basically used by the standard 802.1Q. It has 12 bits and allow the identification of 4096 (2^12) VLANs. Of the 4096 possible VIDs, a VID of 0 is used to identify priority frames and value 4095 (FFF) is reserved, so the maximum possible VLAN configurations are 4,094.

- Length/Type— 2 bytes. This field indicates either the number of MAC-client data bytes that are contained in the data field of the frame, or the frame type ID if the frame is assembled using an optional format.
- Data—Is a sequence of n bytes (42=< n =<1496) of any value. The total frame minimum is 64bytes.
- Frame check sequence (FCS)— 4 bytes. This sequence contains a 32-bit cyclic redundancy check (CRC) value, which is created by the sending MAC and is recalculated by the receiving MAC to check for damaged frames.

Related protocols

IEEE 802.3, 802.2, 802.1, 802.1D, 802.1G, 802.1Q, 802.1p

Sponsor Source

VLAN standard is defined by IEEE (http://www.ieee.org) 802.1Q.

Reference

http://standards.ieee.org/getieee802/download/802.1Q-1998.pdf
IEEE 802.1Q Standard

Protocol Name

IEEE 802.1P: LAN Layer 2 QoS/CoS Protocol for Traffic Prioritization

Protocol Description

IEEE 802.1p specification enables Layer 2 switches to prioritize traffic and perform dynamic multicast filtering. The prioritization specification works at the media access control (MAC) framing layer (OSI model layer 2). The 802.1p standard also offers provisions to filter multicast traffic to ensure it does not proliferate over layer 2-switched networks.

The 802.1p header includes a three-bit field for prioritization, which allows packets to be grouped into various traffic classes. The IEEE has made broad recommendations concerning how network managers can implement these traffic classes, but it stops short of mandating the use of its recommended traffic class definitions. It can also be defined as best-effort QoS (Quality of Service) or CoS (Class of Service) at Layer 2 and is implemented in network adapters and switches without involving any reservation setup. 802.1P traffic is simply classified and sent to the destination; no bandwidth reservations are established.

The IEEE 802.1p is an extension of the IEEE 802.1Q (VLANs tagging) standard and the two standards work in tandem. The 802.1Q standard specifies a tag that appends to an Ethernet MAC frame. The VLAN tag has two parts: the VLAN ID (12-bit) and Prioritization (3-bit). The prioritization field was not defined and used in the 802.1Q VLAN standard. The 802.1P defines this prioritization field.

IEEE 802.1p establishes eight levels of priority. Although network managers must determine actual mappings, IEEE has made broad recommendations. The highest priority is seven, which might go to network-critical traffic such as Routing Information Protocol (RIP) and Open Shortest Path First (OSPF) table updates. Values five and six might be for delay-sensitive applications such as interactive video and voice. Data classes four through one range from controlled-load applications such as streaming multimedia and business-critical traffic - SAP data, for instance - down to “loss eligible” traffic. The zero value is used as a best-effort default, invoked automatically when no other value has been set.

Protocol Structure

IEEE 802.1Q Tagged Frame for Ethernet – modified format from Ethernet (802.3) frame:

7	1	6	6	2	2	2	42-1496	4bytes
Pre-amble	SFD	DA	SA	TPID	TCI	Length Type	Data	CRC

- Preamble (Pre)— 7 bytes. The Pre is an alternating pattern of ones and zeros that tells receiving stations that a frame is coming, and that provides a means to synchronize the frame-reception portions of receiving physical layers with the incoming bit stream.
- Start-of-frame delimiter (SFD)—1 byte. The SFD is an alternating pattern of ones and zeros, ending with two consecutive 1-bits indicating that the next bit is the left-most bit in the left-most byte of the destination address.
- Destination address (DA)— 6 bytes. The DA field identifies which station(s) should receive the frame.
- Source address (SA)— 6 bytes. The SA field identifies the sending station.
- TPID - defined value of 8100 in hex. When a frame has the EtherType equal to 8100, this frame carries the tag IEEE 802.1Q / 802.1P.
- TCI – Tag Control Information field including user priority, Canonical format indicator and VLAN ID.

3	1	12bits
User Priority	CFI	Bits of VLAN ID (VIDI) to identify possible VLANs

- User Priority : Defines user priority, giving eight priority levels. IEEE 802.1P defines the operation for these 3 user priority bits.
- CFI : Canonical Format Indicator is always set to zero for Ethernet switches. CFI is used for compatibility reasons between an Ethernet type network and a Token Ring type network. If a frame received at an Ethernet port has a CFI set to 1, then that frame should not be forwarded as it is to an untagged port.
- VID : VLAN ID is the identification of the VLAN, which is basically used by the standard 802.1Q. It has 12 bits and allows the identification of 4096 (2^12) VLANs. Of the 4096 possible VIDs, a VID of 0 is used to identify priority frames and value 4095 (FFF) is reserved, so the maximum possible VLAN configurations are 4,094.
- Length/Type— 2 bytes. This field indicates either the number of MAC-client data bytes that are contained in the data field of the frame, or the frame type ID if the frame is assembled using an optional format.
- Data—Is a sequence of n bytes (42=< n =<1496) of any value. The total frame minimum is 64bytes.
- Frame check sequence (FCS)— 4 bytes. This sequence contains a 32-bit cyclic redundancy check (CRC) value, which is created by the sending MAC and is recalculated by the receiving MAC to check for

damaged frames.

Related protocols

IEEE 802.2, 802.3, 802.1D, 802.1Q

Sponsor Source

802.1P is an IEEE (http://www.ieee.org) protocol.

Reference

http://standards.ieee.org/getieee802/download/802.1Q-1998.pdf
IEEE 802.1Q Standard

Protocol Name

GARP: Generic Attribute Registration Protocol

Protocol Description

The Generic Attribute Registration Protocol (GARP) provides a generic framework whereby devices in a bridged LAN, e.g. end stations and switches, can register and de-register attribute values, such as VLAN Identifiers, with each other. In doing so, the attributes are propagated to devices in the bridged LAN, and these devices form a “reachability” tree that is a subset of an active topology. GARP defines the architecture, rules of operation, state machines and variables for the registration and de-registration of attribute values.

A GARP participation in a switch or an end station consists of a GARP application component, and a GARP Information Declaration (GID) component associated with each port or switch. The propagation of information between GARP participants for the same application in a bridge is carried out by the GARP Information Propagation (GIP) component. Protocol exchanges take place between GARP participants by means of LLC Type 1 services, using the group MAC address and PDU format defined for the GARP application concerned.

GARP is part of the IEEE 802.1p extension to the 802.1d (spanning tree) specification. It includes:

- GARP Information Declaration (GID): The part of GARP that generates data.
- GARP Information Propagation (GIP): The part of GARP that distributes data.
- GARP Multicast Registration Protocol (GMRP): Provides a mechanism that allows participants to dynamically register and de-register information with the Media Access Control (MAC) bridges attached to the same local-area network (LAN) segment.

Protocol Structure

GARP PDU structure

2 bytes	
Protocol ID	Message

GARP message structure

1 byte				
Attribute type	Attribute 1	...	Attribute n	End mark

GARP attribute structure

1 byte	1 byte	1 byte
Attribute length	Attribute event	Attribute value

- Protocol ID - Identifies the GARP protocol.
- Identifier - Decimal value which aids in matching requests and replies.
- Attribute type - Defines the attribute. Values may be: 1 Group attribute; 2 Service Requirement attribute.
- Attribute length - Length of the Attribute.
- Attribute event - The values of the attribute event can be:
  - 0 Leave_all
  - 1 Join_Empty operator
  - 2 Join_In operator
  - 3 Leave_Empty operator
  - 4 Leave_In operator
  - 5 Empty operator
- Attribute value - This is encoded in accordance with the specification for the Attribute Type.
- End mark - Coded as 0.

Related protocols

IEEE 802.1D, 802.1G, 802.1Q, 802.3ac, 802.1P, VTP, GVRP, GMRP

Sponsor Source

GARP standard is defined by IEEE (http://www.ieee.org) 802.1P.

Reference

http://www.alliedtelesyn.co.nz/documentation/at8700/261/pdf/garp.pdf
Overview of Generic Attribute Registration Protocol

Protocol Name

GMRP: GARP Multicast Registration Protocol

Protocol Description

GARP Multicast Registration Protocol (GMRP) is a Generic Attribute Registration Protocol (GARP) application that provides a constrained multicast flooding facility similar to IGMP snooping. GMRP and GARP are industry-standard protocols defined by the IEEE 802.1P.

GMRP provides a mechanism that allows bridges and end stations to dynamically register group membership information with the MAC bridges attached to the same LAN segment and for that information to be disseminated across all bridges in a Bridged LAN that supports extended filtering services. The operation of GMRP relies upon the services provided by the GARP.

GMRP software components run on both the switch and on the host. On the host, GMRP is typically used with IGMP: The host GMRP software spawns Layer 2 GMRP versions of the host's Layer 3 IGMP control packets. The switch receives both the Layer 2 GMRP and the Layer 3 IGMP traffic from the host. The switch uses the received GMRP traffic to constrain multicasts at Layer 2 in the host's VLAN. In all cases, you can use IGMP snooping to constrain multicasts at Layer 2 without the need to install or configure software on hosts.

When a host wants to join an IP multicast group, it sends an IGMP join message, which spawns a GMRP join message. Upon receipt of the GMRP join message, the switch adds the port through which the join message was received to the appropriate multicast group. The switch propagates the GMRP join message to all other hosts in the VLAN, one of which is typically the multicast source. When the source is multicasting to the group, the switch forwards the multicast only to the ports from which it has received join messages for the group. The switch sends periodic GMRP queries. If a host wants to remain in a multicast group, it responds to the query. In this case, the switch does nothing. If a host does not want to remain in the multicast group, it can either send a leave message or not respond to the periodic queries from the switch. If the switch receives a leave message or receives no response from the host for the duration of the leaveall timer, it removes the host from the multicast group.

Protocol Structure

GMRP messages have the same structure as GARP with the attribute type specific to GMRP: This can be as follows: 1 Group Attribute Type; 2 Service Requirement Attribute Type.

GARP PDU Format:

2 bytes	
Protocol ID	Message

GARP message structure

1 byte				
Attribute type	Attribute 1	...	Attribute n	End mark

GARP attribute structure

1 byte	1 byte	1 byte
Attribute length	Attribute event	Attribute value

- Protocol ID - Identifies the GARP protocol.
- Identifier - Decimal value which aids in matching requests and replies.
- Attribute type - Defines the attribute. Values may be: 1 Group attribute; 2 Service Requirement attribute.
- Attribute length - Length of the Attribute.
- Attribute event - The values of the attribute event can be:
  - 0 Leave_all
  - 1 Join_Empty operator
  - 2 Join_In operator
  - 3 Leave_Empty operator
  - 4 Leave_In operator
  - 5 Empty operator
- Attribute value - This is encoded in accordance with the specification for the Attribute Type.
- End mark - Coded as 0.

Related protocols

IEEE 802.1D, 802.1G, 802.1Q, 802.1P, 802.1ac, VTP, GVRP, GARP

Sponsor Source

GMRP standard is defined by IEEE (http://www.ieee.org) 802.1P.

Reference

http://www.alliedtelesyn.co.nz/documentation/at8700/261/pdf/garp.pdf
Overview of Generic Attribute Registration Protocol

Protocol Name

GVRP: GARP VLAN Registration Protocol

Protocol Description

The GARP (Generic Attribute Registration Protocol) VLAN Registration Protocol (GVRP) defines a GARP application that provides 802.1Q-compliant VLAN pruning and dynamic VLAN creation on 802.1Q trunk ports. GVRP is an application, defined in the IEEE 802.1P standard, which allows for the control of 802.1Q VLANs.

With GVRP, the switch can exchange VLAN configuration information with other GVRP switches, prune unnecessary broadcast and unknown unicast traffic, and dynamically create and manage VLANs on switches connected through 802.1Q trunk ports.

GVRP makes use of GID and GIP, which provide the common state machine descriptions and the common information propagation mechanisms defined for use in GARP-based applications. GVRP runs only on 802.1Q trunk links. GVRP prunes trunk links so that only active VLANs will be sent across trunk connections. GVRP expects to hear join messages from the switches before it will add a VLAN to the trunk. GVRP updates and hold timers can be altered. GVRP ports run in various modes to control how they will prune VLANs. GVRP can be configured to dynamically add and manage VLANS to the VLAN database for trunking purposes.

In other words, GVRP allows the propagation of VLAN information from device to device. With GVRP, a single switch is manually configured with all the desired VLANs for the network, and all other switches on the network learn those VLANs dynamically. An endnode can be plugged into any switch and be connected to that endnode's desired VLAN. For endnodes to make use of GVRP, they need GVRP-aware Network Interface Cards (NICs). The GVRP-aware NIC is configured with the desired VLAN or VLANs, then connected to a GVRP-enabled switch. The NIC communicates with the switch, and VLAN connectivity is established between the NIC and switch.

Protocol Structure

GVRP messages have the same structure as GARP with the attribute type pecific to GVRP: 1 VID Group Attribute Type.

GARP PDU Format:

2 bytes	
Protocol ID	Message

GARP message structure

1 byte				
Attribute type	Attribute 1	...	Attribute n	End mark

GARP attribute structure

1 byte	1 byte	1 byte
Attribute length	Attribute event	Attribute value

- Protocol ID - Identifies the GARP protocol.
- Identifier - Decimal value which aids in matching requests and replies.
- Attribute type - Defines the attribute. Values may be: 1 Group attribute; 2 Service Requirement attribute.
- Attribute length - Length of the Attribute.
- Attribute event - The values of the attribute event can be:
  - 0 Leave_all
  - 1 Join_Empty operator
  - 2 Join_In operator
  - 3 Leave_Empty operator
  - 4 Leave_In operator
  - 5 Empty operator
- Attribute value - This is encoded in accordance with the specification for the Attribute Type.
- End mark - Coded as 0.

Related protocols

IEEE 802.1D, 802.1Q, 802.1P, GMRP, GARP

Sponsor Source

GVRP standard is defined by IEEE (http://www.ieee.org) 802.1Q and 802.1P.

Reference

http://standards.ieee.org/getieee802/download/802.1Q-1998.pdf
IEEE 802.1Q Standard
http://www.alliedtelesyn.co.nz/documentation/at8700/261/pdf/garp.pdf
Overview of Generic Attribute Registration Protocol

Wilress LAN Protocols

Protocol Name

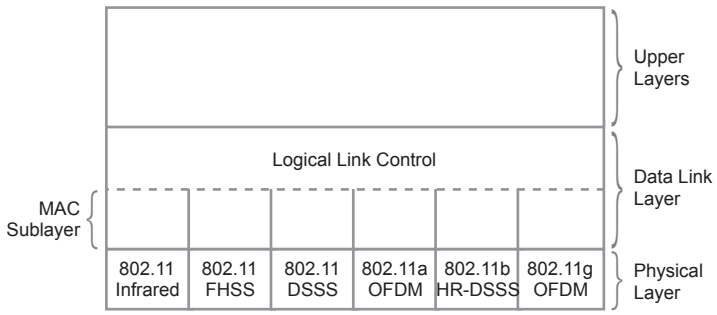
WLAN: Wireless LAN by IEEE 802.11 protocols

Protocol Description

The Wireless Local Area Network (WLAN) technology is defined by the IEEE 802.11 family of specifications. There are currently four specifications in the family: 802.11, 802.11a, 802.11b, and 802.11g. All four use the Ethernet protocol and CSMA/CA (carrier sense multiple access with collision avoidance instead of CSMA/CD) for path sharing.

- 802.11 -- applies to wireless LANs and provides 1 or 2 Mbps transmission in the 2.4 GHz band using either frequency hopping spread spectrum (FHSS) or direct sequence spread spectrum (DSSS).
- 802.11a -- an extension to 802.11 that applies to wireless LANs and provides up to 54 Mbps in the 5GHz band. 802.11a uses an orthogonal frequency division multiplexing (OFDM) encoding scheme rather than FHSS or DSSS. The 802.11a specification applies to wireless ATM systems and is used in access hubs.
- 802.11b (also referred to as 802.11 High Rate or Wi-Fi) -- an extension to 802.11 that applies to wireless LANS and provides 11 Mbps transmission (with a fallback to 5.5, 2 and 1 Mbps) in the 2.4 GHz band. 802.11b uses only DSSS. 802.11b is a modification of the original 802.11 standard, allowing wireless functionality comparable to Ethernet.
- 802.11g -- offers wireless transmission over relatively short distances at 20 – 54 Mbps in the 2.4 GHz band. 802.11g also uses the OFDM encoding scheme.

The modulation used in 802.11 has historically been phase-shift keying (PSK). The modulation method selected for 802.11b is known as complementary code keying (CCK), which allows higher data speeds and is less susceptible to multipath-propagation interference. 802.11a uses a modulation scheme known as orthogonal frequency-division multiplexing (OFDM) that makes possible data speeds as high as 54 Mbps, but most commonly, communications takes place at 6 Mbps, 12 Mbps, or 24 Mbps. The 802.11 stack structure is as follows:



For short range and low power wireless (less than 10 meters) communications among personal devices such as PDA, Bluetooth and subsequent IEEE standards (802.15) are taking effects. For long range wireless communications in the metropolitan areas, WiMax and IEEE 802.16 are the standards.

Protocol Structure

801.11 protocol family MAC frame structure:

2	2	6	6	6	2	6	0-2312	4bytes
Frame Control	Duration	Address 1	Address 2	Address 3	Seq	Address 4	Data	Check sum

- Frame Control Structure:

2	2	4	1	1	1	1	1	1	1	1
Version	Type	Subtype	To DS	From DS	MF	Retry	Pwr	More	W	O

  - Protocol Version - indicates the version of IEEE 802.11 standard.
  - Type – Frame type: Management, Control and Data.
  - Subtype - Frame subtype: Authentication frame, Deauthentication frame; Association request frame; Association response frame; Reassociation request frame; Reassociation response frame; Disassociation frame; Beacon frame; Probe frame; Probe request frame or Probe response frame.
  - To DS - is set to 1 when the frame is sent to Distribution System (DS)
  - From DS - is set to 1 when the frame is received from the Distribution System (DS)
  - MF- More Fragment is set to 1 when there are more fragments belonging to the same frame following the current fragment
  - Retry indicates that this fragment is a retransmission of a previously transmitted fragment. (For receiver to recognize duplicate transmissions of frames)
  - Pwr - Power Management indicates the power management mode that the station will be in after the transmission of the frame.
  - More - More Data indicates that there are more frames buffered to this station.
  - W - WEP indicates that the frame body is encrypted according to the WEP (wired equivalent privacy) algorithm.
  - O - Order indicates that the frame is being sent using the Strictly-Ordered service class.

- Duration/ID (ID) -
  - Station ID is used for Power-Save poll message

frame type.
    - The duration value is used for the Network Allocation Vector (NAV) calculation.
- Address fields (1-4) - contain up to 4 addresses (source, destination, transmitter and receiver addresses) depending on the frame control field (the ToDS and FromDS bits).
- Sequence Control - consists of fragment number and sequence number. It is used to represent the order of different fragments belonging to the same frame and to recognize packet duplications.
- Data - is information that is transmitted or received.
- CRC - contains a 32-bit Cyclic Redundancy Check (CRC).

Related protocols

IEEE 802.2, 802.3, 802.11, 802.11a, 802.11b, 802.11g, Bluetooth, 802.15, WiMax, 802.16

Sponsor Source

WLAN protocols are defined by IEEE (http://www.ieee.org) 802.11 specifications.

Reference

http://standards.ieee.org/getieee802/download/802.11-1999.pdf
Wireless LAN Media Access Control (MAC) and Physical Control Specifications
http://standards.ieee.org/getieee802/download/802.11a-1999.pdf
Wireless LAN MAC: High-speed physical layer in the 5 GHz Band.
http://standards.ieee.org/getieee802/download/802.11b-1999_Cor1-2001.pdf
Wireless LAN MAC: Higher-speed physical layer extension in the 2.4 GHz band.
http://www.cis.ohio-state.edu/~durresi/presentations/802.11.pdf
“Wireless Data Networking IEEE 802.11 & Overview of IEEE 802.11b”

Protocol Name

IEEE 802.1X: EAP over LAN (EAPOL) for LAN/WLAN Authentication and Key Management

Protocol Description

The IEEE 802.1X offers an effective framework for authenticating and controlling user traffic to a protected network, as well as dynamically varying encryption keys. 802.1X ties a protocol called EAP (Extensible Authentication Protocol) to both the wired and wireless LAN media and supports multiple authentication methods, such as token cards, Kerberos, one-time passwords, certificates, and public key authentication.

In the 802.1x architecture, there are three key components: 1) Supplicant: the user or client that wants to be authenticated; 2) the authentication server, typically a RADIUS server; and 3) the authenticator: the device in between, such as a wireless access point, which can be simple and dumb.

The key protocol in 802.1x is called EAP encapsulation over LANs (EAPOL). It is currently defined for Ethernet-like LANs including 802.11 wireless, as well as token ring LANs (including FDDI). The operation process in 802.1X is as follows:

1. The supplicant (such as a client wireless card) sends an “EAP-Response/Identity” packet to the authenticator (such as an 802.11 access point), which is then passed on to the authentication server (RADIUS server which is located at the wired side of the access point).
2. The authentication server sends back a challenge to the authenticator. The authenticator unpacks this from IP and repackages it into EAPOL and sends it to the supplicant.
3. The supplicant responds to the challenge via the authenticator and passes the response onto the authentication server. The authentication server uses a specific authentication algorithm to verify the client’s identity. This could be through the use of digital certificates or other EAP authentication type.
4. If the supplicant provides proper identity, the authentication server responds with a success message, which is then passed onto the supplicant. The authenticator now opens port for the supplicant to access the LAN based on attributes that came back from the authentication server.

The 802.1X (EAPOL) protocol provides effective authentication regardless of whether 802.11 WEP keys are implemented or there is no encryption at all. If configured to implement dynamic key exchange, the 802.1X authentication server can return session keys to the access point along with the accept message. The access point uses the session keys to build, sign and encrypt an EAP key message that is sent to the client immediately after sending the success message. The client can then use contents of the key message to define applicable encryption keys.

802.1X (EAPOL) is a delivery mechanism and does not provide the actual authentication mechanisms. When utilizing 802.1X, an EAP type, such as Transport Layer Security (EAP-TLS) or EAP Tunneled Transport Layer Security (EAP-TTLS), which defines how the authentication takes place, must be chosen. The specific EAP type resides on the authentication server and within the operating system or application software on the client devices. The access point acts as a “pass through” for 802.1X messages, which means that any EAP type can be specifie without an 802.1X-compliant access point needing to be upgraded.

Protocol Structure

EAPOL Frame Format for 802.3/Ethernet:

2 bytes	1 byte	1 byte	2 bytes	Variable
PAE Ethernet Type	Protocol version	Packet type	Packet Body length	Packet Body

- PAE Ethernet type – PAE (Port Access Entity) Ethernet type contains the Ethernet Type value assigned for use by the PAE.
- Protocol version –an unsigned binary number, the value of which is the version of the EAPOL protocol.
- Packet type –an unsigned binary number, the value of which determines the type of the packet as follows: a) EAP-packet; b) EAPOL-Start; c) EAPOL-Logoff; d) EAPOL-Key; e) EAPOL-Encapsulated-ASF-Alert
- Packet body length – an unsigned binary, the value of which defines the length in octets of the packet body field.
- Packet Body – This field is presented if the packet type contains the value EAP-Packet, EAPOL-Key, or EAP-Encapsulated-ASF-Alert, otherwise, it is not presented.

EAPOL Frame Format for Token Ring /FDDI:

8 bytes	1 byte	1 byte	2 bytes	Variable
SNAP Ethernet Type	Protocol version	Packet type	Packet Body length	Packet Body

- SNAP Ethernet Type – contains the SNAP-encoded Ethernet type encoded in the SNAP format as follows: 1-3 bytes carry the standard SNAP header; 4-6 bytes carry the SNAP PID; 7-8 bytes carry the PAE

Ethernet Type value.

Related protocols

Ethernet, EAP, RADIUS, Token Ring

Sponsor Source

EAPOL (802.1X) is defined by IEEE (http://www.ieee.org).

Reference

http://standards.ieee.org/getieee802/download/802.1X-2001.pdf
Port based Network Access Control

Protocol Name

IEEE 802.15 and Bluetooth: WPAN Communications

Protocol Description

IEEE 802.15, a standardization of Bluetooth wireless specification defined by IEEE, is for wireless personal area networks (WPANs). IEEE 802.15 has characters such as short-range, low power, low cost, small networks and communication of devices within a Personal Operating Space.

The initial version, 802.15.1, was adapted from the Bluetooth specification and is fully compatible with Bluetooth 1.1. Bluetooth becomes widely used specification for wireless communications among portable digital devices including notebook computers, peripherals, cellular telephones, beepers, and consumer electronic devices. The specification also allows for connection to the Internet. 802.15.1/Bluetooth specify standards in on the Physical layer and Data link layer of the OSI model with the following four sub-layers:

RF layer: The air interface is based on antenna power range starting from 0 dBm up to 20 dBm. Bluetooth operates in the 2.4 GHz band and the link range is anywhere from 10 centimeters to 10 meters.

Baseband layer: establishes the Bluetooth physical link between devices forming a piconet -- a network of devices connected in an ad hoc fashion using Bluetooth technology.

Link manager: sets up the link between Bluetooth devices. Other functions of the link manager include security, negotiation of Baseband packet sizes, power mode and duty cycle control of the Bluetooth device, and the connection states of a Bluetooth device in a piconet.

Logical Link Control and Adaptation Protocol (L2CAP): provides the upper layer protocols with connectionless and connection-oriented services.

The IEEE 802.15 Working Groups are making progress to improve the Bluetooth standards. They proposed two general categories of 802.15: the low rate 802.15.4 (TG4) and high rate 802.15.3 (TG3). The TG4 version provides data speeds of 20 Kbps or 250 Kbps, low power and low cost solutions. The TG3 version supports data speeds ranging 20 Mbps or greater, for multi-media applications.

IEEE 802.15/Bluetooth, IEEE 802.11/WLAN and IEEE 802.16/WiMAX technologies are complementary to each other and each play a unique role in today's wireless communications. The following table outlines the three technologies:

Parameters	802.16a (WiMAX)	802.11 (WLAN)	802.15 (Bluetooth)
Frequency Band:	2-11GHz	2.4GHz	Varies
Range	~31 miles	~100 meters	~10meters
Data transfer rate:	70 Mbps	11 Mbps – 55 Mbps	20Kbps – 55 Mbps
Number of users:	Thousands	Dozens	Dozens

Related Terms:

WLAN, WMAN, WiMAX, IEEE 802.11, IEEE 802.16

Reference Links:

http://standards.ieee.org/getieee802/802.15.html
IEEE 802.15 Specification Download Page

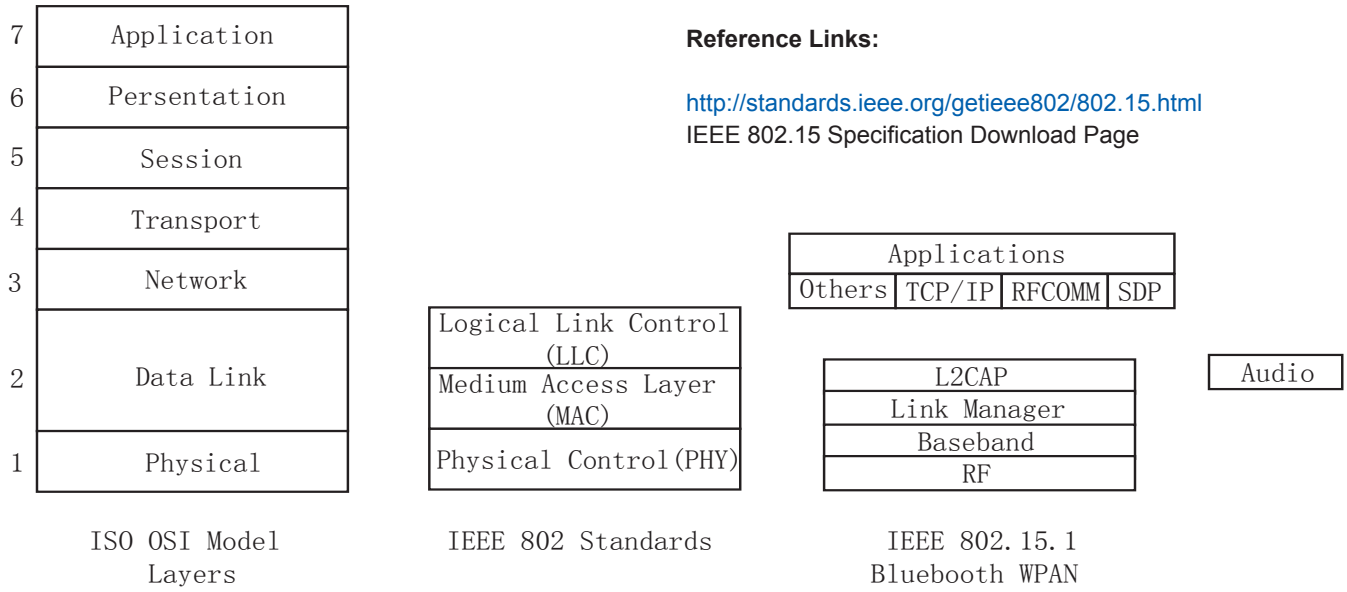


Figure 2-14: IEEE 802.15 (Bluetooth) Protocol Stack

Other Protocols

Protocol Name

FDDI: Fiber Distributed Data Interface

Protocol Description

Fiber Distributed Data Interface (FDDI) is a set of ANSI protocols for sending digital data over fiber optic cable. FDDI networks are token-passing (similar to IEEE 802.5 Token Ring protocol) and dual-ring networks, and support data rates of up to 100 Mbps. FDDI networks are typically used as backbone technology because of the protocol supports a high bandwidth and a great distance. A related copper specification similar to FDDI protocols, called Copper Distributed Data Interface (CDDI), has also been defined to provide 100-Mbps service over twisted-pair copper.

An extension to FDDI, called FDDI-2, supports the transmission of voice and video information as well as data. Another variation of FDDI, called FDDI Full Duplex Technology (FFDT) uses the same network infrastructure but can potentially support data rates up to 200 Mbps.

FDDI uses dual-ring architecture with traffic on each ring flowing in opposite directions (called counter-rotating). The dual rings consist of a primary and a secondary ring. During normal operation, the primary ring is used for data transmission, and the secondary ring remains idle. As will be discussed in detail later in this chapter, the primary purpose of the dual rings is to provide superior reliability and robustness.

FDDI specifies the physical and media access portions of the OSI reference model. FDDI is not actually a single specification but is a collection of four separate specifications, each with a specific function. Combined, these specifications have the capability to provide high-speed connectivity between upper-layer protocols such as TCP/IP and IPX, and media such as fiber-optic cabling.

FDDI's four specifications are the Media Access Control (MAC), Physical Layer Protocol (PHY), Physical-Medium Dependent (PMD), and Station Management (SMT) specifications. The MAC specification defines how the medium is accessed, including frame format, token handling, addressing, algorithms for calculating cyclic redundancy check (CRC) value, and error-recovery mechanisms. The PHY specification defines data encoding/decoding procedures, clocking requirements, and framing, among other functions. The PMD specification defines the characteristics of the transmission medium, including fiber-optic links, power levels, bit-error rates, optical components, and connectors. The SMT specification defines FDDI station configuration, ring configuration, and ring control features, including station insertion and removal, initialization, fault isolation and recovery, scheduling, and statistics collection.

Protocol Structure

2	6	6	0-30	Variable	4bytes
Frame control	Destination address	Source address	Route information	Information	FCS

- Frame control - The frame control structure is as follows:

C	L	F	F	Z	Z	Z	Z

  C Class bit: 0 Asynchronous frame; 1 Synchronous frame/
  L Address length bit: 0 16 bits (never); 1 48 bits (always).
  FF Format bits.
  ZZZZ Control bits.
- Destination address - The address structure is as follows:

I/G	U/L	Address bits

- Source address - The address structure is as follows:

I/G	RII	Address bits

  I/G Individual/group address: 0 Group address; 1 Individual address.
  RII Routing information indicator: 0 RI absent; 1 RI present.
- Route Information - The structure of the route information is as follows:

3	5	1	6	1	16	16		16
RT	LTH	D	LF	r	RD1	RD2	...	RDn

  RC Routing control (16 bits).
  RDn Route descriptor (16 bits).
  RT Routing type (3 bits).
  LTH Length (5 bits).
  D Direction bit (1 bit).
  LF Largest frame (6 bits).
  r Reserved (1 bit).
- Information - The Information field may be LLC, MAC or SMT protocol.
- FCS - Frame check sequence.

Related protocols

IEEE 802.5

Sponsor Source

FDDI is defined by ANSI (http://www.ansi.org) X3T9.5.

Reference

http://www.cisco.com/univercd/cc/td/doc/cisintwk/ito_doc/fddi.htm
Fiber Distributed Data Interface

Protocol Name

Token Ring: IEEE 802.5 LAN Protocol

Protocol Description

Token Ring is a LAN protocol, defined in IEEE 802.5 where all stations are connected in a ring and each station can directly hear transmissions only from its immediate neighbor. Permission to transmit is granted by a message (token) that circulates around the ring.

Token Ring as defined in IEEE 802.5 is originated from the IBM Token Ring LAN technologies. Both are based on the Token Passing technologies. While they differ in minor ways; they are generally compatible with each other.

Token-passing networks move a small frame, called a token, around the network. Possession of the token grants the right to transmit. If a node receiving the token has information to send, it seizes the token, alters 1 bit of the token (which turns the token into a start-of-frame sequence), appends the information that it wants to transmit, and sends this information to the next station on the ring. While the information frame is circling the ring, no token is on the network, which means that other stations wanting to transmit must wait. Therefore, collisions cannot occur in Token Ring networks.

The information frame circulates the ring until it reaches the intended destination station, which copies the information for further processing. The information frame continues to circle the ring and is finally removed when it reaches the sending station. The sending station can check the returning frame to see whether the frame was seen and subsequently copied by the destination.

Unlike Ethernet CSMA/CD networks, token-passing networks are deterministic, which means that it is possible to calculate the maximum time that will pass before any end station will be capable of transmitting. This feature and several reliability features make Token Ring networks ideal for applications in which delay must be predictable and robust network operation is important.

The Fiber Distributed-Data Interface (FDDI) also uses the Token Passing protocol.

Protocol Structure

<table>
<tr><td>1</td><td>2</td><td>3</td><td colspan="2">9</td><td>15bytes</td></tr>
<tr><td>SDEL</td><td>AC</td><td>FC</td><td colspan="2">Destination address</td><td>Source address</td></tr>
<tr><td colspan="6">Route information 0-30 bytes</td></tr>
<tr><td colspan="6">Information (LLC or MAC) variable</td></tr>
<tr><td colspan="3">FCS (4 bytes)</td><td>EDEL</td><td>FS</td><td></td></tr>
</table>

- SDEL / EDEL - Starting Delimiter / Ending Delimiter. Both the SDEL and EDEL have intentional Manchester code violations in certain bit positions so that the start and end of a frame can never be accidentally recognized in the middle of other data.
- AC - Access Control field contains the priority fields.
- FC - Frame Control field indicates whether the frame contains data or control information
- Destination address – Destination station address.
- Source address –Source station address.
- Route information – The field with routing control, route descriptor and routing type information.
- Information - The Information field may be LLC or MAC.
- FCS - Frame check sequence.
- Frame status - Contains bits that may be set on by the recipient of the frame to signal recognition of the address and whether the frame was successfully copied.

Related protocols

IEEE 802.2, 802.3, 802.4, 802.5

Sponsor Source

Token Ring is defined by IEEE (http://www.ieee.org) 802.5.

Reference

http://standards.ieee.org/getieee802/download/802.5-1998.pdf
Token Ring Access Method and Physical Layer Specification
http://www.cisco.com/univercd/cc/td/doc/cisintwk/ito_doc/tokenrng.htm
Token Ring and IEEE 802.5

Protocol Name

LLC: Logic Link Control (IEEE 802.2)

Protocol Description

Logic Link Control (LLC) is the IEEE 802.2 LAN protocol that specifies an implementation of the LLC sublayer of the data link layer. IEEE 802.2 LLC is used in IEEE802.3 (Ethernet) and IEEE802.5 (Token Ring) LANs to perform these functions:

a. Managing the data-link communication
b. Link Addressing
c. Defining Service Access Points (SAPs)
d. Sequencing

The LLC provides a way for the upper layers to deal with any type of MAC layer (e.g. Ethernet - IEEE 802.3 CSMA/CD or Token Ring IEEE 802.5 Token Passing).

LLC originated from the High-Level Data-Link Control (HDLC) and uses a subclass of the HDLC specification. LLC defines three types of operation for data communication:

Type 1: Connectionless. The connectionless operation is basically sending with no guarantee of receiving;

Type 2: Connection Oriented. The Type 2 Connection-Oriented operation for the LLC layer provides these 4 services: Connection establishment, Confirmation and acknowledgement that data has been received, Error recovery by requesting received bad data to be resent, Sliding Windows (Modulus: 128), which is a method of increasing the rate of data transfer.

Type 3: Acknowledgement with connectionless service.

The Type 1 connectionless service of LLC specifies a static-frame format and allows network protocols to run on it. Network protocols that fully implement a transport layer will generally use Type 1 service.

The Type 2 connection-oriented service of LLC provides reliable data transfer. It is used in LAN environments that do not invoke network and transport layer protocols.

Protocol Structure

Logic Link Control Layer (LLC) Header:

8	16	24 or 32bit	Variable
DSAP	SSAP	Control	LLC information

- DSAP - The destination service access point structure is as follows:

1	8bit
I/G	Address bits

I/G Individual/group address may be: 0 Individual DSAP; 1 Group DSAP.

- SSAP - The source service access point structure is as follows:

1	8bit
C/R	Address bits

C/R Command/response: 0 Command; 1 Response.

- Control - The structure of the control field is as follows:

	1	8				9	16bit
Information	0	N(S)				P/F	N(R)
Supervisory	1	0	SS	XXXX		P/F	N(R)
Unnumbered	1	1	MM	P/F	MMM		

N(S) Transmitter send sequence number.
N(R) Transmitter receive sequence number.
P/F Poll/final bit. Command LLC PDU transmission/response LLC PDU transmission.
S Supervisory function bits:
00 RR (receive ready).
01 REJ (reject).
10 RNR (receive not ready).
X Reserved and set to zero.
M Modifier function bits.

- LLC information - LLC data or higher layer protocols.

Related protocols

IEEE 802.3, 802.5

Sponsor Source

LLC is defined by IEEE (http://www.ieee.org) in the 802.2 specifications.

Reference

http://standards.ieee.org/getieee802/download/802.2-1998.pdf
IEEE 802.2 specification.
https://secure.linuxports.com/howto/intro_to_networking/c5048.htm
IEEE 802.2: Logic Link Control Layer

Protocol Name

SNAP: SubNetwork Access Protocol

Protocol Description

The SubNetwork Access Protocol (SNAP) is a standard for the transmission of IP datagrams over IEEE 802 networks. In other words, IP datagrams can be sent on IEEE 802 networks encapsulated within the 802.2 LLC and SNAP data link layers and the 802.3, 802.4 or 802.5 physical network layers.

SNAP is included in an extension of the Logic Link Control (LLC IEEE 802.2) header and is used for encapsulating IP datagrams and ARP requests and replies on IEEE 802 networks. The SNAP header follows the LLC header and contains an organization code indicating that the following 16 bits specify the EtherType code. Normally, all communication is performed using 802.2 type 1 communication. Consenting systems on the same IEEE 802 network may use 802.2 type 2 communication after verifying that it is supported by both nodes. This is accomplished using the 802.2 XID mechanism. However, type 1 communication is the recommended method at this time and must be supported by all implementations.

The mapping of 32-bit Internet addresses to 16-bit or 48-bit IEEE 802 addresses is done via the dynamic discovery procedure of the Address Resolution Protocol (ARP). The IEEE 802 networks may have 16-bit or 48-bit physical addresses. SNAP allows the use of either size of address within a given IEEE 802 network.

With SNAP, the transmission of IP datagrams does not depend on the transmission rate of the under layer LAN technologies (various types of Ethernet and Token Ring), which may have very different transmission rates (from 1 to 20 Mbps).

Protocol Structure

LLC Header:

8	16	24 or 32bit
DSAP	SSAP	Control

For details of the LLC header, please see the LLC page.

SNAP header:

24	40bit
Organization code	EtherType

When SNAP is present the DSAP and SSAP fields within the LLC header contain the value 170 (decimal) each and the Control field is set to 3 (unnumbered information).

- Organization code - Set to 0.
- EtherType - Specifies which protocol is encapsulated within the IEEE 802 network: IP = 2048, ARP = 2054.

Related protocols

IEEE 802.2, 802.3, 802.4, 802.5, IP, ARP

Sponsor Source

SNAP is defined by IEEE (http://www.ieee.org) and IETF (http://www.ietf.org).

Reference

http://standards.ieee.org/getieee802/download/802.2-1998.pdf
IEEE 802.2 specification.
http://www.javvin.com/protocol/rfc1042.pdf
A Standard for the Transmission of IP Datagrams over IEEE 802 Networks

Protocol Name

STP: Spanning Tree Protocol (IEEE 802.1D)

Protocol Description

Spanning-Tree Protocol (STP) as defined in IEEE 802.1D is a link management protocol that provides path redundancy while preventing undesirable loops in the network. For an Ethernet network to function properly, only one active path can exist between two stations. Loops occur in networks for a variety of reasons. The most common reason for loops in networks is a deliberate attempt to provide redundancy—in case one link or switch fails, another link or switch can take over.

STP is a technology that allows bridges to communicate with each other to discover physical loops in the network. The protocol then specifies an algorithm that bridges can use to create a loop-free logical topology. In other words, STP creates a tree structure of loop-free leaves and branches that spans the entire Layer 2 network.

Spanning-Tree Protocol operation is transparent to end stations, which are unaware whether they are connected to a single LAN segment or a switched LAN of multiple segments. Where two bridges are used to interconnect the same two computer network segments, Spanning Tree is a protocol that allows the bridges to exchange information so that only one of them will handle a given message that is being sent between two computers within the network.

Bridge Protocol Data Units (BPDUs) are used by bridges in a network to exchange information regarding their status. The Spanning-Tree Protocol uses the BPDU information to elect the root switch and root port for the switched network, as well as the root port and designated port for each switched segment.

The program in each bridge that allows it to determine how to use the protocol is known as the spanning tree algorithm, which is specifically constructed to avoid bridge loops. The algorithm ensures that a bridge uses only the most efficient path when faced with multiple paths. If the best path fails, the algorithm recalculates the network and finds the next best route.

The spanning tree algorithm determines the network (which computer hosts are in which segment) and this data is exchanged using Bridge Protocol Data Units (BPDUs). It is broken down into two steps:

Step 1: The algorithm determines the best message a bridge can send by evaluating the configuration messages it has received and choosing the best option.

Step 2: Once it selects the top message for a particular bridge to send, it compares its choice with possible configuration messages from the non-root-connections it has. If the best option from step 1 isn't better than what it receives from the non-root-connections, it will prune that port.

Protocol Structure

The Bridge Protocol Data Units (BPDUs).

Protocol ID (2)	Version (1)	Type (1)	Flags (1)	Rood ID (8)	Root Path (4)
Sender BID (8)	Port ID (2)	M-Age (2)	Max Age (2)	Hello (2)	FD (2 Bytes)

- Protocol ID—Always 0.
- Version—Always 0.
- Type—Determines which of the two BPDU formats this frame contains (Configuration BPDU or TCN BPDU).
- Flags—Used to handle changes in the active topology covered in the next section on Topology Change Notifications.
- Root BID —Contains the Bridge ID of the Root Bridge. After convergence, all Configuration BPDUs in the bridged network should contain the same value for this field (for a single VLAN). NetXRay breaks out the two BID subfields: Bridge Priority and bridge MAC address.
- Root Path Cost—The cumulative cost of all links leading to the Root Bridge.
- Sender BID —The BID of the bridge that created the current BPDU. This field is the same for all BPDUs sent by a single switch (for a single VLAN), but it differs between switches.
- Port ID—Contains a unique value for every port. Port 1/1 contains the value 0×8001, whereas Port 1/2 contains 0×8002.
- Message Age—Records the time since the Root Bridge originally generated the information that the current BPDU is derived from.
- Max Age—Maximum time that a BPDU is saved. Also influences the bridge table aging timer during the Topology Change Notification process (discussed later).
- Hello Time—Time between periodic Configuration BPDUs.
- Forward Delay—The time spent in the Listening and Learning states. Also influences timers during the Topology Change Notification process (discussed later).

Related protocols

IEEE 802.2, 802.3, 802.1P, 802.1Q

Sponsor Source

STP is defined by IEEE (http://www.ieee.org) in 802.1D.

Reference

http://standards.ieee.org/getieee802/download/802.1D-1998.pdf
ANSI/IEEE Std 802.1D 1998 Edition
http://www.cisco.com/univercd/cc/td/doc/product/rtrmgmt/sw_ntman/cwsimain/cwsi2/cwsiug2/vlan2/stpapp.htm
Understanding Spanning Tree Protocol

Metropolitan Area Network and MAN Protocols

Description

A Metropolitan Area Network (MAN) is a computer network usually spanning a campus or a city, which typically connect a few local area networks using high speed backbone technologies. A MAN often provides efficient connections to a wide area network (WAN). There are three important features which discriminate MANs from LANs or WANs:

1. The network size falls intermediate between LANs and WANs. A MAN typically covers an area of between 5 and 50 km range. Many MANs cover an area the size of a city, although in some cases MANs may be as small as a group of buildings.
2. A MAN (like a WAN) is not generally owned by a single organization. The MAN, its communications links and equipment are generally owned by either a consortium of users or by a network service provider who sells the service to the users.
3. A MAN often acts as a high speed network to allow sharing of regional resources. It is also frequently used to provide a shared connection to other networks using a link to a WAN.

MAN adopted technologies from both LAN and WAN to serve its purpose. Some legacy technologies used for MAN are ATM, FDDI, DQDB and SMDS. These older technologies are in the process of being displaced by Gigabit Ethernet and 10 Gigabit Ethernet. At the physical level, MAN links between LANs have been built on fibre optical cables or using wireless technologies such as microwave or radio.

The Metropolitan Area Network (MAN) protocols are mostly at the data link level (layer 2 in the OSI model), which are defined by IEEE, ITU-T, etc.

Key Protocols

The key MAN protocols are listed as follows:

ATM: Asynchronous Transfer Mode
DQDB: Distributed Queue Dual Bus Defined in IEEE 802.6
Ethernet at data rate 10 Gbps (IEEE 802.3ae)
FDDI: Fiber Distributed Data Interface
Gigabit Ethernet: Ethernet at data rate 1000Mbps (IEEE 802.3z, 802.3ab)
IEEE 802.16: Wireless MAN (WiMAX)
SMDS: Switched Multimegabit Data Service

Related protocols

LAN, WAN, TCP/IP

Sponsor Source

MAN protocols are mostly defined by IEEE.

Protocol Name

DQDB: Distributed Queue Dual Bus (Defined in IEEE 802.6)

Protocol Description

Distributed Queue Dual Bus (DQDB) is a Data-link layer communication protocol for Metropolitan Area Networks (MANs), specified in the IEEE 802.6 standard and designed for use in MANs. DQDB is designed for data as well as voice and video transmission and is based on cell switching technology (similar to ATM). DQDB, which permits multiple systems to interconnect using two unidirectional logical buses, is an open standard that is designed for compatibility with carrier transmission standards such as SMDS.

For a MAN to be effective it requires a system that can function across long, “city-wide” distances of several miles, have a low susceptibility to error, adapt to the number of nodes attached and have variable bandwidth distribution. Using DQDB, networks can be thirty miles long and function in the range of 34 Mbps to 155 Mbps. The data rate fluctuates due to many hosts sharing a dual bus, as well as to the location of a single host in relation to the frame generator, but there are schemes to compensate for this problem making DQDB function reliably and fairly for all hosts.

The DQDB is composed of two bus lines with stations attached to both and a frame generator at the end of each bus. The buses run in parallel in such a fashion as to allow the frames generated to travel across the stations in opposite directions. Below is a picture of the basic DQDB architecture.

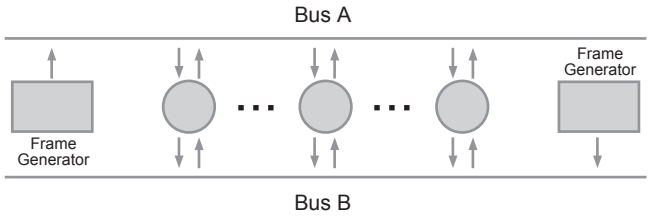
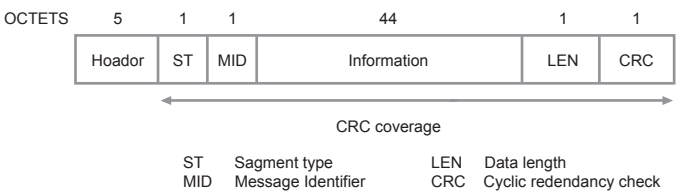


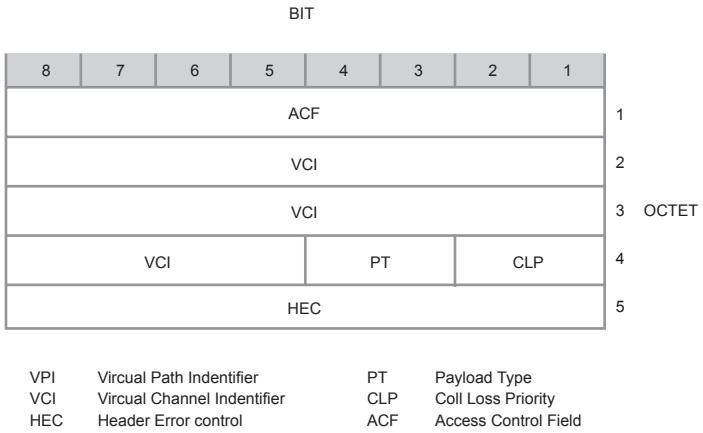
Figure 2-15: DQDB Architecture

Protocol Structure

The DQDB cell has a similar format as to the ATM:



DQDB cell header:



Related protocols

IEEE 802.6, ATM, SMDS

Sponsor Source

DQDB is defined by IEEE (http://www.ieee.org) 802.6.

Reference

http://standards.ieee.org/getieee802/download/802.6-1994.pdf
Distributed Queue Dual Bus (DQDB) access Method and Physical layer specifications
http://www.cisco.com/univercd/cc/td/doc/cisintwk/ito_doc/smds.htm
Switched Multimegabit Data Service

Protocol Name

SMDS: Switched Multimegabit Data Service

Protocol Description

Switched Multimegabit Data Service (SMDS) is a broadband networking technology developed by Bellcore based on the IEEE 802.6 DQDB (Distributed Queue Dual Bus) MAN technology.

SMDS can use fiber- or copper-based media. It supports speeds of 1.544 Mbps over DS-1, or 44.736 Mbps over DS-3. In addition, SMDS data units are large enough to encapsulate entire IEEE 802.3, IEEE 802.5, and Fiber Distributed Data Interface (FDDI) frames. SMDS operates by accepting high-speed customer data in increments of up to 9,188 octets, and dividing it into 53-octet cells for transmission through the service provider's network. These cells are reassembled, at the receiving end, into the customer data.

The SMDS Interface Protocol (SIP) is a three-level protocol, based on IEEE 802.6 DQDB, which controls the customer's access to the network. SIP Levels 3 and 2 operate at the Media Access Control (MAC) sublayer of the data link layer of the OSI reference model. SIP Level 1 operates at the physical layer of the OSI reference model. SIP Level 3 receives and transports frames of the upper layer protocol information. SIP Level 2 controls access to the physical medium. SIP Level 1 includes the PLCP and the transmission system.

Protocol Structure

The SIP Level 3 PDU is shown in the following diagram.

36	<= 9188	0-3	0,4	4bytes
Header	Information	PAD	X + CRC32	Trailer

The format of the level 3 header is as follows:

Rsv (1)	BE-tag (1)	BA-size (2)	Destination address (8bytes)					
Source address (8bytes)			X + HLPI (6 bits)	PL (2)	X+ QoS (4 bits)	CIB (1)	HEL (3 bits)	X+Bridging (2bytes)
HE (12 bytes)								

And the format of the level 3 trailer is as follows:

1 byte	1 byte	2 bytes
Reserved	Betag	BAsize

The Level 3 PDU fields are described as follows:

- Rsv - Reserved. A 1-octet field that the CPE and the SS fill with zeros.
- BE tag - A 1-octet field that contains a beginning/end tag.
- BA size - A 2-octet field containing the length in octets of the Level 3 PDU from the beginning of the Destination Address field and including the CRC32 field, if present.
- Destination address - An 8-octet field containing the address of the intended recipient of the PDU.
- Source address - An 8-octet field containing the address of the sender of this PDU. This field contains Address Type and Address subfields as described for Destination Address.
- HLPI - Higher Layer Protocol Identifier. A 6-bit field that aligns the SIP and DQDB protocol formats.
- PL - PAD Length. A 2-bit field that indicates the number of octets in the PAD field, which aligns the Level 3 PDU on a 32-bit boundary.
- QoS - Quality of Service. A 4-bit field that aligns the SIP and DQDB protocol formats.
- CIB - CRC32 Indication Bit. A 1-bit field that indicates the presence (1) or absence (0) of the CRC32 field.
- HEL - Header Extension Length. A 3-bit field that indicates the number of 32-bit words in the Header Extension field.
- Bridging - A 2-octet field that aligns the SIP and DQDB Bridging protocol formats.
- HE - Header Extension. A 12-octet field that contains the version and carrier-selection information .
- Information field - Variable-length field, up to 9,188 octets in length, which contains user information.
- PAD - Variable-length field, 1-3 octets in length, filled with zeros aligning the entire PDU on a 32-bit boundary.
- CRC32 - 2-octet field that performs error detection on the PDU, beginning with the DA field, up to and including the CRC32 field.

The SIP Level 2 PDU contains a 5-octet header, a 44-octet Segmentation Unit (payload) and a 2-octet trailer as shown below.

Access control (8 bits)	Network control info (32 bits)	Segment type (2bits)	Sequence number (4 bits)	Message ID (10 bits)
Segmentation unit (352 bits or 44 bytes)				
Payload length (6 bits)	Payload CRC (10bits)			

The Level 2 PDU fields are described as follows:

- Access control - 8-bit field that indicates whether the Level 2 PDU Access Control contains information (1) or is empty (0).
- Network control info - 4-octet field that determines whether Network Control Information of the Level 2 PDU contains information (FFFFF022H) or is empty

(0).

- Segment type - 2-bit field that indicates how the receiver should process non-empty Level 2 PDUs. Sequence number - 4-bit number that verifies that all the Level 2 PDUs belonging to a single Level 3 PDU have been received in the correct order.
- Sequence number - 4-bit number that verifies that all the Level 2 PDUs belonging to a single Level 3 PDU have been received in the correct order.
- Message identifier - 10-bit number that allows the various segments to be associated with a single Level 3 PDU.
- Segmentation unit - 44-octet field that contains a portion of the Level 3 PDU.
- Payload length - 6-bit field that indicates which of the 44 octets in the Segmentation Unit contain actual data. BOM and COM segments always indicate 44 octets. EOM segments indicate between 4 and 44 octets, in multiples of 4 octets. SSM segments indicate between 28 and 44 octets, in multiples of 4 octets.
- Payload CRC - 10-bit field that performs error detection on the Segment Type, Sequence Number, Message Identifier, Segmentation Unit, Payload Length and Payload CRC fields.
- Once assembled, SIP Level 2 PDUs are passed to the PLCP and physical functions within SIP Level 1 for transmission.

Related protocols

IEEE 802.6 (DQDB), ATM, SMDS

Sponsor Source

SMDS is defined by Bellcore (Telcordia) (http://www.telcordia.com)

Reference

http://www.cisco.com/univercd/cc/td/doc/cisintwk/ito_doc/smds.htm
Switched Multimegabit Data Service

Protocol Name

IEEE 802.16: Broadband Wireless MAN Standard (WiMAX)

Protocol Description

The IEEE 802.16 defines wireless service that provides a communications path between a subscriber site and a core network such as the public telephone network and the Internet. The Wireless MAN technology is also branded as WiMAX, This wireless broadband access standard provides the missing link for the “last mile” connection in metropolitan area networks where DSL, Cable and other broadband access methods are not available or too expensive.

IEEE 802.16 standards are concerned with the air interface between a subscriber’s transceiver station and a base transceiver station. IEEE 802.16 is approved by th IEEE in June 2004. Three working groups have been chartered to produce standards: Task Group 1 of IEEE 802.16 developed a point-to-multipoint broadband wireless access standard for systems in the frequency range 10-66 GHz. The standard covers both the Media Access Control (MAC) and the physical (PHY) layers. Task groups a and b are jointly producing an amendment to extend the specification to cover both the licensed and unlicensed bands in the 2-11 GHz range.

IEEE 802.16 and WiMAX are designed as a complimentary technology to Wi-Fi and Bluetooth. The following table provides a quick comparison of 802.16a with to 802.11b:

Parameters	802.16a (WiMAX)	802.11 (WLAN)	802.15 (Bluetooth)
Frequency Band:	2-11GHz	2.4GHz	Varies
Range	~31 miles	~100 meters	~10meters
Data transfer rate:	70 Mbps	11 Mbps – 55 Mbps	20Kbps – 55 Mbps
Number of users:	Thousands	Dozens	Dozens

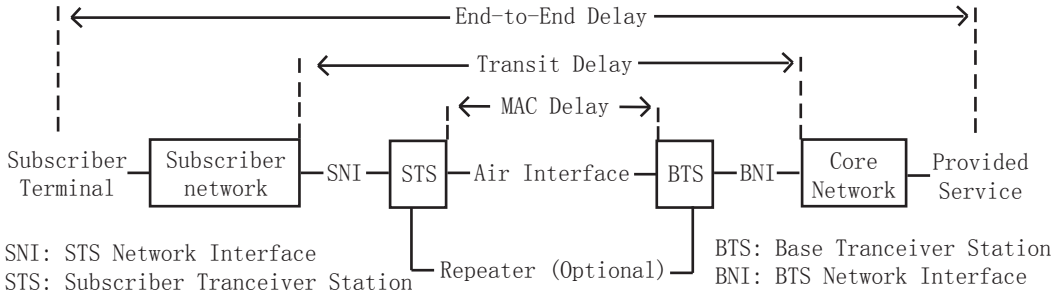


Figure 2-16: IEEE 802.16 (WiMax) Functional Flow Chart

Protocol Structure

IEEE 802.16 Protocol Architecture has 4 layers: Convergence, MAC, Trnamission and physical, which can be map to two OSI lowest layers: phusical and data link.

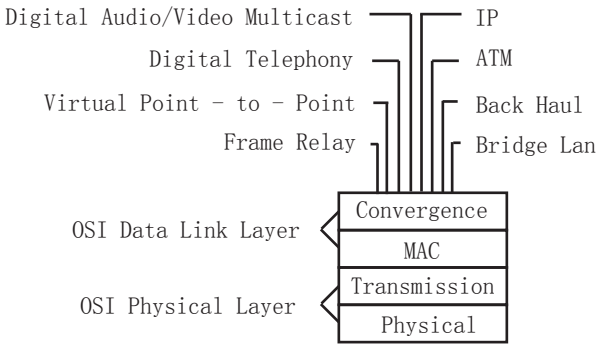


Figure 2-17: IEEE 802.16 (WiMax) Protocol Stack

Related protocols

IEEE 802.11, WLAN, Bluetooth, IEEE 802.15

Sponsor Source

Wireless MAN is defined by the IEEE 802.16 working group.

Reference

http://grouper.ieee.org/groups/802/16/published.html
Published 802.16 standards

http://www.intel.com/ebusiness/pdf/wireless/intel/80216_wimax.pdf
IEEE 802.16 and WiMAX

Storage Area Network and SAN Protocols

Description

Storage Area Network (SAN) is a high-speed network or subnetwork whose primary purpose is to transfer data between computer and storage systems. A storage device is a machine that contains nothing but a disk or disks for storing data. A SAN consists of a communication infrastructure, which provides physical connections; and a management layer, which organizes the connections, storage elements, and computer systems so that data transfer is secure and robust.

Typically, a storage area network is part of the overall network of computing resources for an enterprise. A storage area network is usually clustered in close proximity to other computing resources but may also extend to remote locations for backup and archival storage. SANs support disk mirroring, backup and restore, archival and retrieval of archived data, data migration from one storage device to another, and the sharing of data among different servers in a network. SANs can incorporate subnetworks with network-attached storage (NAS) systems.

There are a few SAN technologies available in today's implementations, such as IBM's optical fiber ESCON which is enhanced by FICON architecture, or the newer Fibre Channel technology. High speed Ethernet is also used in the storage Area Network for connection. SCSI and iSCSI are popular technologies used in the Storage Area Network.

A typical SAN architecture is displayed as follows:

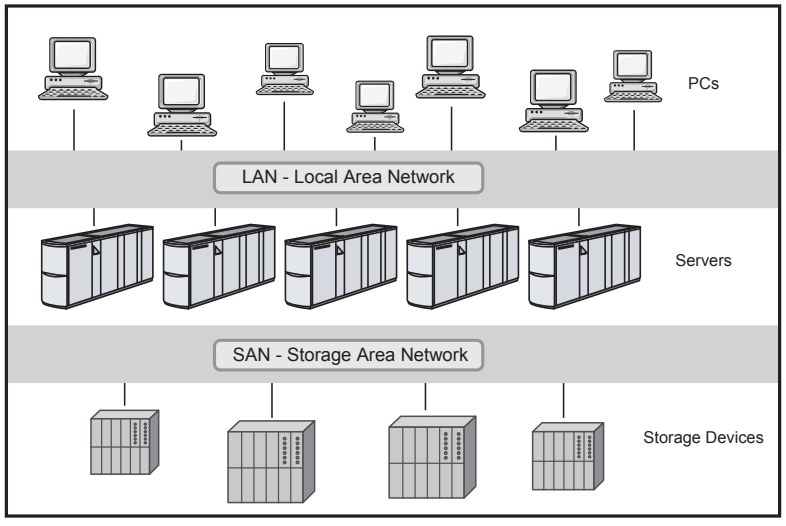


Figure 2-18: Storage Area Network Architecture

Key Protocols

The key SAN protocols are listed as follows:

Storage Area Network Protocols
FCIP: Entire Fibre Channel Frame Over IP
iFCP: Internet Fibre Channel Protocol
iSCSI: Internet Small Computer System Interface
iSNS: Internet Storage Name Service
NDMP: Network Data Management Protocol
SAS: Serial Attached SCSI
SCSI: Small Computer System Interface

Related protocols

LAN, WAN, TCP/IP

Sponsor Source

Storage Area Network protocols are defined by IETF, ANSI and ISO.

Reference

http://www.redbooks.ibm.com/redbooks/SG245470.html
Storage Area Network

Protocol Name

FC & FCP: Fibre Channel and Fibre Channel Protocol

Protocol Description

The Fibre Channel Standards (FCS) define a high-speed data transfer mechanism that can be used to connect workstations, mainframes, supercomputers, storage devices and displays. FCS addresses the need for very fast transfers of large volumes of information and could relieve system manufacturers of the burden of supporting the variety of channels and networks currently in place, as it provides one standard for networking, storage and data transfer. Fibre Channel Protocol (FCP) is the interface protocol of SCSI on the Fibre Channel. The key Fibre Channel characteristics are as follows:

- Performance from 266 megabits/second to over four gigabits/second
- Supports both optical and electrical media, working from 133 Megabits/sec up to 1062 Megabits/sec with distances up to 10 km.
- Small connectors
- High-bandwidth utilization with distance insensitivity
- Support for multiple cost/performance levels, from small systems to supercomputers
- Ability to carry multiple existing interface command sets, including Internet Protocol (IP), SCSI, IPI, HIPPI-FP, and audio/video.

Fibre Channel consists of the following layers:

FC-0 The interface to the physical media

FC-1 The encoding and decoding of data and out-of-band physical link control information for transmission over the physical media

FC-2 The transfer of frames, sequences and Exchanges comprising protocol information units.

FC-3 Common Services required for advanced features such as striping, hunt group and multicast.

FC-4 Application interfaces that can execute over fibre channel such as the fibre channel protocol for SCSI (FCP).

The fundamental entity in fibre channel is the fibre channel network. Unlike a layered network architecture, a fibre channel network is largely specified by functional elements and the interfaces between them. These consist, in part, of the following:

a) N_PORTs -- The end points for fibre channel traffic.
b) FC Devices -- The fibre channel devices to which the N_PORTs provide access.
c) Fabric Ports -- The interfaces within a fibre channel network that provide attachment for an N_PORT.
d) The network infrastructure for carrying frame traffic between N_PORTs.
e) Within a switched or mixed fabric, a set of auxiliary servers, including a name server for device discovery and network address resolution.

The principal fibre channel network topologies consist of the following:

a) Arbitrated Loop -- A series of N_PORTs connected together in daisy-chain fashion.
b) Switched Fabric -- A network consisting of switching elements.
c) Mixed Fabric -- A network consisting of switches and “fabric-attached” loops. A loop-attached N_PORT (NL_PORT) is connected to the loop through an L_PORT and accesses the fabric by way of an FL_PORT.

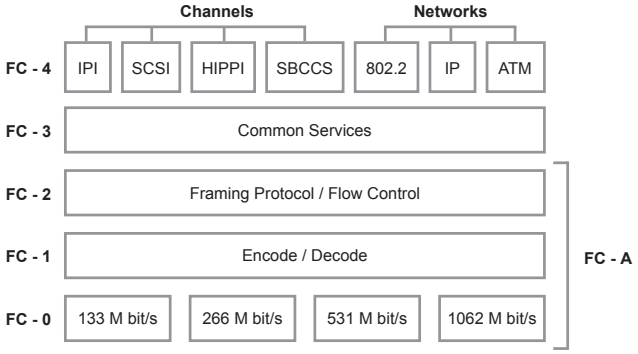


Figure 2-19: Fibre Channel Protocol

Protocol Structure

Fibre Channel frames are a maximum 2148 bytes long. Fibre Channel Frame Header Structure:

8	16	24	32bit
Routine Control	Destination Address		
Reserve	Source Address		
Upper Level Protocol Type	Frame Control		
Seq_ ID	Data Field Control	Sequence Count	
Parameter			

Related protocols

SCSI, iFCP, FCP, FCIP, mFCP

Sponsor Source

Fibre Channel (FC) and Fibre Channel Protocol (FCP) are defined by ANSI (www.ansi.org).

Reference

http://www.javvin.com/protocol/rfc3643.pdf
Fibre Channel (FC) Frame Encapsulation
http://hsi.web.cern.ch/HSI/fcs/spec/overview.htm
Fibre Channel Overview

Protocol Name

FCIP: Fibre Channel over TCP/IP

Protocol Description

Fibre Channel Over TCP/IP (FCIP) describes mechanisms that allow the interconnection of islands of Fibre Channel storage area networks over IP-based networks to form a unified storage area network in a single Fibre Channel fabric. FCIP relies on IP-based network services to provide the connectivity between the storage area network islands over local area networks, metropolitan area networks, or wide area networks.

The primary function of an FCIP Entity is forwarding FC Frames, employing FC Frame Encapsulation. Viewed from the IP Network perspective, FCIP Entities are peers and communicate using TCP/IP. Each FCIP Entity contains one or more TCP endpoints in the IP-based network. Viewed from the FC Fabric perspective, pairs of FCIP Entities, in combination with their associated FC Entities, forward FC Frames between FC Fabric elements. The FC End Nodes are unaware of the existence of the FCIP Link.

FC Primitive Signals, Primitive Sequences, and Class 1 FC Frames are not transmitted across an FCIP Link because they cannot be encoded using FC Frame Encapsulation. The path (route) taken by an encapsulated FC Frame follows the normal routing procedures of the IP Network.

An FCIP Entity MAY contain multiple FCIP Link Endpoints, but each FCIP Link Endpoint (FCIP_LEP) communicates with exactly one other FCIP_LEP. FCIP Entities do not actively participate in FC Frame routing. The FCIP Control & Services module MAY use TCP/IP quality of service features.

It is necessary to statically or dynamically configure each FCIP entity with the IP addresses and TCP port numbers corresponding to FCIP Entities with which it is expected to initiate communication. If dynamic discovery of participating FCIP Entities is supported, the function SHALL be performed using the Service Location Protocol (SLPv2). Before creating a TCP Connection to a peer FCIP Entity, the FCIP Entity attempting to create the TCP connection SHALL statically or dynamically determine the IP address, TCP port, expected FC Fabric Entity World Wide Name, TCP Connection Parameters, and Quality of Service Information.

FCIP Entities do not actively participate in the discovery of FC source and destination identifiers. Discovery of FC addresses (accessible via the FCIP Entity) is provided by techniques and protocols within the FC architecture.

To support IP Network security, FCIP Entities MUST: 1)implement cryptographically protected authentication and cryptographic data integrity keyed to the authentication process, and 2) implement data confidentiality security features.

On an individual TCP Connection, this specification relies on TCP/IP to deliver a byte stream in the same order that it was sent.

Protocol Structure

Fibre Channel Frame Encapsulation Header Structure – FCIP specific

8	16	24	32bit
Protocol #	Version	-Protocol #	-Version
replication of encapsulation word 0			
pFlags	Reserved	-pFlags	-Reserved
Flags	Frame Length	-Flags	-Frame Length
Time Stamp (integer)			
Time Stamp (fraction)			
CRC			

Common fields:

- Protocol# - IANA-assigned protocol number identifying the protocol using the encapsulation.
- Version - Encapsulation version as specified in [ENCAP]
- -Protocol# - One's complement of the protocol#
- -Version – One's complement of the version
- Flags - Encapsulation flags
- Frame Length - Contains the length of the entire FC Encapsulated frame including the FC Encapsulation Header and the FC frame (including SOF and EOF words) in units of 32-bit words.
- -Flags – One's complement of the Flags field.
- -Frame Length – One's complement of the Frame Length field.
- Time Stamp [integer] - Integer component of the frame time stamp as specified in [ENCAP].
- Time Stamp - Fractional component of the time stamp [fraction] as specified in [ENCAP].
- CRC - Header CRC. MUST be valid for iFCP.

FCIP specific fields:

Word 1 of the Protocol Specific field SHALL contain an exact copy of word 0 in FC Frame Encapsulation.

The pFlags (protocol specific flags) field provides information about the protocol specific usage of the FC Encapsulation Header.

Ch	Reserved	SF

The SF (Special Frame) bit indicates whether the FCIP Frame is an encapsulated FC Frame or an FSF (FCIP Special Frame).

The Ch (Changed) bit indicates whether an echoed FSF has been intentionally altered. The Ch bit SHALL be 0 unless the FSF bit is 1.

Related protocols

SCSI, iFCP, FCP, FCIP, mFCP, iSCSI, TCP

Sponsor Source

FCIP is defined by IETF (www.ietf.org).

Reference

http://www.javvin.com/protocol/rfc3643.pdf
Fibre Channel (FC) Frame Encapsulation
http://www.javvin.com/protocol/draft-ietf-ips-fcovertcpip-12.pdf
Fibre Channel Over TCP/IP (FCIP)

Protocol Name

iFCP: Internet Fibre Channel Protocol

Protocol Description

Internet Fibre Channel Protocol (iFCP) is a gateway-to-gateway protocol, which provides fibre channel fabric services to fibre channel devices over a TCP/IP network. iFCP uses TCP to provide congestion control, error detection and recovery. iFCP's primary objective is to allow interconnection and networking of existing fibre channel devices at wire speeds over an IP network. The protocol and method of frame address translation defined permit the attachment of fibre channel storage devices to an IP-based fabric by means of transparent gateways.

The fundamental entity in Fibre Channel is the fibre channel network. Unlike a layered network architecture, a fibre channel network is largely specified by functional elements and the interfaces between them. These consist, in part, of the following:

a) N_PORTs -- The end points for fibre channel traffic.
b) FC Devices -- The fibre channel devices to which the N_PORTs provide access.
c) Fabric Ports -- The interfaces within a fibre channel network that provide attachment for an N_PORT.
d) The network infrastructure for carrying frame traffic between N_PORTs.
e) Within a switched or mixed fabric, a set of auxiliary servers, including a name server for device discovery and network address resolution.

The iFCP protocol enables the implementation of fibre channel fabric functionality on an IP network in which IP components and technology replace the fibre channel switching and routing infrastructure.

The main function of the iFCP protocol layer is to transport fibre channel frame images between locally and remotely attached N_PORTs. When transporting frames to a remote N_PORT, the iFCP layer encapsulates and routes the fibre channel frames comprising each fibre channel Information Unit via a predetermined TCP connection for transport across the IP network.

When receiving fibre channel frame images from the IP network, the iFCP layer de-encapsulates and delivers each frame to the appropriate N_PORT. The iFCP layer processes the following types of traffic:

a) FC-4 frame images associated with a fibre channel application protocol.
b) FC-2 frames comprising fibre channel link service requests and responses
c) Fibre channel broadcast frames
d) iFCP control messages required to setup, manage or terminate an iFCP session.

Protocol Structure

Fibre Channel Frame Encapsulation Header Structure – iFCP specific:

8	16	24	32bit
Protocol #	Version	-Protocol #	-Version
Reserved (must be zero)			
LS_Command_ACC	IFCP Flags	SOF	EOF
Flags	Frame Length	-Flags	-Frame Length
Time Stamp (integer)			
Time Stamp (fraction)			
CRC			

Common fields:

- Protocol# - IANA-assigned protocol number identifying the protocol using the encapsulation. For iFCP, the value assigned by [ENCAP] is 2.
- Version - Encapsulation version as specified in [ENCAP]
- -Protocol# - One's complement of the protocol#
- -Version – One's complement of the version
- Flags - Encapsulation flags
- Frame Length - Contains the length of the entire FC Encapsulated frame including the FC Encapsulation Header and the FC frame (including SOF and EOF words) in units of 32-bit words.
- -Flags – One's complement of the Flags field.
- -Frame Length – One's complement of the Frame Length field.
- Time Stamp [integer] - Integer component of the frame time stamp as specified in [ENCAP].
- Time Stamp - Fractional component of the time stamp [fraction] as specified in [ENCAP].
- CRC - Header CRC. MUST be valid for iFCP.

iFCP specific fields:

LS_COMMAND_ACC - For a special link service ACC response to be processed by iFCP, the LS_COMMAND_ACC field SHALL contain a copy of bits 0 through 7 of the LS_COMMAND to which the ACC applies. Otherwise the LS_COMMAND_ACC field SHALL be set to zero.

iFCP Flags - iFCP-specific flags:

Reserved	SES	TRP	SPC

SES: Session control frame
TRP: Transparent mode Flag
SPC: Special processing flag

- SOF - Copy of the SOF delimiter encoding
- EOF - Copy of the EOF delimiter encoding

Related protocols

SCSI, iFCP, FCP, FCIP, mFCP, iSCSI, TCP

Sponsor Source

iFCP is defined by IETF (www.ietf.org).

Reference

http://www.javvin.com/protocol/rfc3643.pdf
Fibre Channel (FC) Frame Encapsulation
http://www.javvin.com/protocol/draft-ietf-ips-ifcp-14.pdf
iFCP - A Protocol for Internet Fibre Channel Storage Networking

Protocol Name

iSCSI: Internet Small Computer System Interface (SCSI)

Protocol Description

Internet Small Computer System Interface (iSCSI) is a TCP/IP-based protocol for establishing and managing connections between IP-based storage devices, hosts and clients, creating a Storage Area Network (SAN). The SAN makes possible to use the SCSI protocol in network infrastructures for high-speed data transfer at the block level between multiple elements of data storage networks.

The architecture of the SCSI is based on the client/server model, which is mostly implemented in an environment where devices are very close to each other and connected with SCSI buses. Encapsulation and reliable delivery of bulk data transactions between initiators and targets through the TCP/IP network is the main function of the iSCSI. iSCSI provides a mechanism for encapsulating SCSI commands on an IP network and operates on top of TCP.

For today's SAN (Storage Area Network), the key requirements for data communication are: 1) Consolidation of data storage systems, 2) Data backup, 3) Server clusterization, 4) Replication, and 5) Data recovery in emergency conditions. In addition, a SAN is likely to have a geographic distribution over multiple LANs and WANs with various technologies. All operations must be conducted in a secure environment and with QoS. iSCSI is designed to perform the above functions in the TCP/IP network safely and with proper QoS.

The iSCSI has four components:

- iSCSI Address and Naming Conventions: An iSCSI node is an identifier of SCSI devices (in a network entity) available through the network. Each iSCSI node has a unique iSCSI name (up to 255 bytes) which is formed according to the rules adopted for Internet nodes.
- iSCSI Session Management: The iSCSI session consists of a Login Phase and a Full Feature Phase which is completed with a special command.
- iSCSI Error Handling: Because of a high probability of errors in data delivery in some IP networks, especially WAN, where the iSCSI can work, the protocol provides a number of measures for handling errors.
- iSCSI Security: As the iSCSI can be used in networks where data can be accessed illegally, the protocol allows different security methods.

Protocol Structure

iSCSI PDU structure:

8	16	24	32bit
Basic Header Structure (BHS)			
Additional Header Structure 1 (AHS) (optional)			
…			
Additional Header Structure n (AHS) (optional)			
Header Digest (optional)			
Data Segment (optional)			
Data Digest (optional)			

iSCSI BHS Format:

8			16	24	32bit
.	I	Opcode	F	Opcode-specific fields	
Total AHS Length			Data Segment length		
Opcode-specific fields or Logic Unit Number (LUN) (8 bytes)					
Initiator Task Tag (4 bytes)					
Opcode-specific fields (28 bytes)					

- I - For request PDUs, the I bit set to 1 is an immediate delivery marker.
- Opcode - The Opcode indicates the type of iSCSI PDU the header encapsulates. The Opcodes are divided into two categories: initiator opcodes and target opcodes. Initiator opcodes are in PDUs sent by the initiator (request PDUs). Target opcodes are in PDUs sent by the target (response PDUs).
- Final (F) bit - When set to 1 it indicates the final (or only) PDU of a sequence.
- Opcode-specific Fields - These fields have different meanings for different opcode types.
- TotalAHSLength - Total length of all AHS header segments in units of four byte words including padding, if any.
- DataSegmentLength - This is the data segment payload length in bytes (excluding padding). The DataSegmentLength MUST be 0 whenever the PDU has no data segment.
- LUN - Some opcodes operate on a specific Logical Unit. The Logical Unit Number (LUN) field identifies which Logical Unit. If the opcode does not relate to a Logical Unit, this field is either ignored or may be used in an opcode specific way.
- Initiator Task Tag - The initiator assigns a Task Tag to each iSCSI task it issues. While a task exists, this tag MUST uniquely identify the task session-wide.

Related protocols

SCSI, iFCP, FCP, FCIP, mFCP, TCP

Sponsor Source

ISCSI is defined by IETF (www.ietf.org).

Reference

http://www.javvin.com/protocol/rfc3347.pdf
Small Computer Systems Interface protocol over the Internet (iSCSI) Requirements and Design Considerations
http://www.javvin.com/protocol/draft-ietf-ips-iscsi-20.pdf
iSCSI (iSCSI protocol specification draft)

Protocol Name

iSNS and iSNSP: Internet Storage Name Service and iSNS Protocol

Protocol Description

iSNS facilitates scalable configuration and management of iSCSI and Fibre Channel (FCP) storage devices in an IP network, by providing a set of services comparable to that available in Fibre Channel networks. iSNS thus allows a commodity IP network to function at a comparable level of intelligence to a Fibre Channel fabric. iSNS allows the administrator to go beyond a simple device-by-device management model, where each storage device is manually and individually configured with its own list of known initiators and targets. Using the iSNS, each storage device subordinates its discovery and management responsibilities to the iSNS server. The iSNS server thereby serves as the consolidated configuration point through which management stations can configure and manage the entire storage network, including both iSCSI and Fibre Channel devices.

iSNS can be implemented to support iSCSI and/or iFCP protocols as needed; an iSNS implementation MAY provide support for one or both of these protocols as desired by the implementer. Implementation requirements within each of these protocols are further discussed in section 5. Use of iSNS is OPTIONAL for iSCSI, and REQUIRED for iFCP.

There are four main functions of the iSNS:

1) A Name Service Providing Storage Resource Discovery
2) Discovery Domain (DD) and Login Control Service
3) State Change Notification Service
4) Open Mapping of Fibre Channel and iSCSI Devices

iSNS has the following key Architectural Components:

iSNS Protocol (iSNSP) - iSNSP is a flexible and lightweight protocol that specifies how iSNS clients and servers communicate. It is suitable for various platforms, including switches and targets as well as server hosts.

iSNS Client - iSNS clients initiate transactions with iSNS servers using the iSNSP. iSNS clients are processes that are coresident in the storage device, and can register device attribute information, download information about other registered clients in a common Discovery Domain (DD), and receive asynchronous notification of events that occur in their DD(s). Management stations are a special type of iSNS client that have access to all DDs stored in the iSNS.

iSNS Server - iSNS servers respond to iSNS protocol queries and requests, and initiate iSNS protocol State Change Notifications. Properly authenticated information submitted by a registration request is stored in an iSNS database.

iSNS Database - The iSNS database is the information repository for the iSNS server(s). It maintains information about iSNS client attributes. A directory-enabled implementation of iSNS may store client attributes in an LDAP directory infrastructure.

Protocol Structure

iSNSP message structure:

16	32bit
iSNSP version	Function ID
PDU Length	Flags
Transaction ID	Sequence ID
PDU Payload (variable bytes)	
Authentication Block (variable bytes)	

- iSNSP Version – the current version is 0x0001. All other values are RESERVED.
- iSNSP Function ID - defines the type of iSNS message and the operation to be executed. iSNSP PDU Length - specifies the length of the PDU PAYLOAD field in bytes. The PDU Payload contains TLV attributes for the operation.
- iSNSP Flags - indicates additional information about the message and the type of Network Entity that generated the message.
- iSNSP Transaction ID - MUST be set to a unique value for each concurrently outstanding request message. Replies MUST use the same TRANSACTION ID value as the associated iSNS request message.
- iSNSP Sequence ID - The SEQUENCE ID has a unique value for each PDU within a single transaction.
- iSNSP PDU Payload - The iSNSP PDU PAYLOAD is of variable length and contains attributes used for registration and query operations.
- Authentication Block - For iSNS multicast and broadcast messages, the iSNSP provides authentication capability. The iSNS Authentication Block is identical in format to the SLP authentication block.

Related protocols

iFCP, iSCSI, TCP, UDP, NAT, SNMP, SLP, DHCP, DNS, BOOTP

Sponsor Source

iSNS and iSNSP are defined by IET (www.ietf.org).

Reference

http://www.javvin.com/protocol/draft-ietf-ips-isns-22.pdf
Internet Storage Name Service (iSNS)

Protocol Name

NDMP: Network Data Management Protocol

Protocol Description

The Network Data Management Protocol (NDMP) is an open protocol for enterprise-wide network based data management. NDMP defines a network-based mechanism and protocol for controlling backup, recovery, and other transfers of data between primary and secondary storage.

The NDMP (version 5) architecture is based on the client server model. The backup management software is considered the client, namely NDMP data management application (DMA). There is one and only one DMA in an NDMP session. Each additional process participating in the data management session is an NDMP service.

There are three types of NDMP services: Data service, Tape service and Translate service. The NDMP architecture separates the network attached Data Management Application (DMA), Data Servers and Tape Servers participating in archival or recovery operations. NDMP also provides low-level control of tape devices and SCSI media changers.

The DMA is the application that creates and controls the NDMP session. The Client reads, stores and manages all session state: server topology, tape sets and numbering, synchronization points, etc., everything needed to continue the session, or reverse the session, for instance to restore fully or partially a file system. There is one and only one connection between the DMA and each NDMP service. This is the NDMP Control Connection. The control connection is a bi-directional TCP/IP connection.

If the Client is distributed in such a way that two or more client processes need to communicate to one NDMP service, the client side commands needs to be merged into one command stream and synchronized by the DMA. This command stream is sent to the service over one connection.

The NDMP protocol is based on XDR encoded messages transmitted over a TCP/IP connection.

Protocol Structure

An NDMP message consists of a message header optionally followed by a message body. Each message is identified by a message number that is sent as part of the message header. Each message will be XDR encoded and sent within a single XDR record.

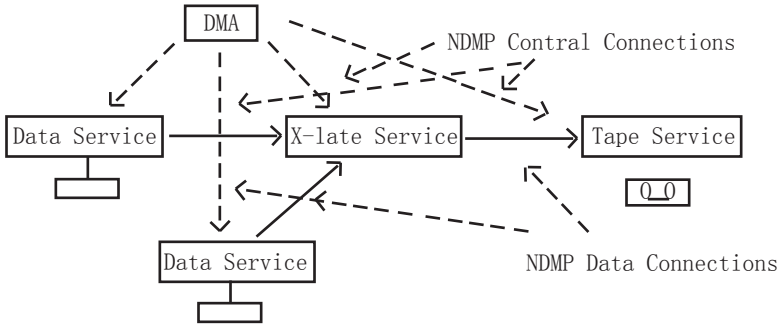


Figure 2-20: NDMP Functional Components

NDMP_header	message_request
NDMP_header	message_reply

The message headers are defined by the following XDR block

```
enum ndmp_header_message_type
{
NDMP_MESSAGE_REQUEST,
 NDMP_MESSAGE_REPLY
};
struct ndmp_header
{
      u_long                  sequence;
      u_long                  time_stamp;
      ndmp_header_message_type    message_type;
      enum ndmp_message          message;
      u_long                  reply_sequence;
      ndmp_error                error;
};
```

Message header data definitions:

sequence - The sequence number is a connection local counter that starts at one and increases by one for every message sent.

time_stamp - The time_stamp identifies the time, in seconds.

message_type - The message_type enum identifies the message as a request or a reply message.

message - The message field identifies the message.

reply_sequence - The reply_sequence field is 0 in a request message. In reply messages, the reply_sequence is the sequence number from the request message to which the reply is associated.

error - The error field is 0 in request messages. In reply messages, the error field identifies any problem that occurred receiving or decoding the message.

Related protocols

iSCSI, iFCP, FCP, FCIP, mFCP

Sponsor Source

NDMP standards are defined by Storage Networking Industry Association (www.snia.org)

Reference

http://www.ndmp.org
Network Data Management Protocol specifications

Protocol Name

SCSI: Small Computer System Interface

Protocol Description

Small Computer System Interface (SCSI), an ANSI standard, is a parallel interface standard used by Apple Macintosh computers, PCs, and many UNIX systems for attaching peripheral devices to computers. SCSI interfaces provide for faster data transmission rates than standard serial and parallel ports. In addition, you can attach many devices to a single SCSI port. There are many variations of SCSI: SCSI-1, SCSI-2, SCSI-3 and the recently approved standard Serial Attached SCSI (SAS).

SCSI-1

SCSI-1 is the original SCSI and is now obsolete. Basically, SCSI-1 used an 8-bit bus, and supported data rates of 4 MBps.

SCSI-2

SCSI-2, an improved version of SCSI-1 based on CCS, is a minimum set of 18 basic commands which work on all manufacture's hardware. SCSI-2 also provides extra speed with options called Fast SCSI and a 16-bit version called Wide SCSI. A feature called command queuing gives the SCSI device the ability to execute commands in the most eficient order. Fast SCSI delivers a 10 MB/sec transfer rate. When combined with a 16-bit bus, this doubles to 20 MB/sec (Fast-Wide SCSI).

SCSI-3

SCSI-3 has many advances over SCSI-2 such as Serial SCSI. This feature will allow data transfer up to 100MB/sec through a six-conductor coaxial cable. SCSI-3 solves many of the termination and delay problems of older SCSI versions. It also eases SCSI installation woes by being more plug-and-play in nature, as by automatic SCSI ID assigning and termination. SCSI-3 also supports 32 devices while SCSI-2 supports only 8.

SCSI-3 changed the document structure. It is not one document dealing with all the different layers and electrical interfaces, but a collection of documents covering the physical layer, the basic protocol specific to that electrical interface, the primary command set layer (SPC) and the specific protocol layer. For example, the specific protocol layer document contains the Hard Disk interface Commands in the Block Commands (SBC), Steam Commands for tape drives (SSC), Controller Commands for RAID arrays (SCC), Multimedia Commands (MMC), Media Changer Commands (MCC) and enclosure services commands (SES). There is an overall architectural model (SAM).

Elements of SCSI-3 are in use today in the forms of Ultra-Wide and Ultra SCSI drives. Ultra SCSI delivers 20MB/sec over the 8-bit bus. Ultra-Wide SCSI incorporates the 16-bit bus, and the speed rises to 40MB/sec.

SAS - Serial Attached SCSI

Serial Attached SCSI (SAS) is an evolutionary replacement for the Parallel SCSI physical storage interface. Serial Attached SCSI offers much faster communication and easier configuration. In addition, Serial Attached SCSI provides device compatibility with Serial ATA and uses similar cabling.

Serial Attached SCSI (SAS) is a point-to-point connection and allows multiple ports to be aggregated into a single controller, either built onto the mother board or as an add-on. Its technology is built upon the robust and tested parallel SCSI communication technology. Serial Attached SCSI (SAS) uses Serial ATA (SATA) cables which are a thin point-to-point connection allowing easy cable routing within a computer system, without the need for daisy-chaining. The first implementation of Serial Attached SCSI provides 1.5 Gb/sec (150MB/sec) of performance for each drive within an array.

Protocol Structure

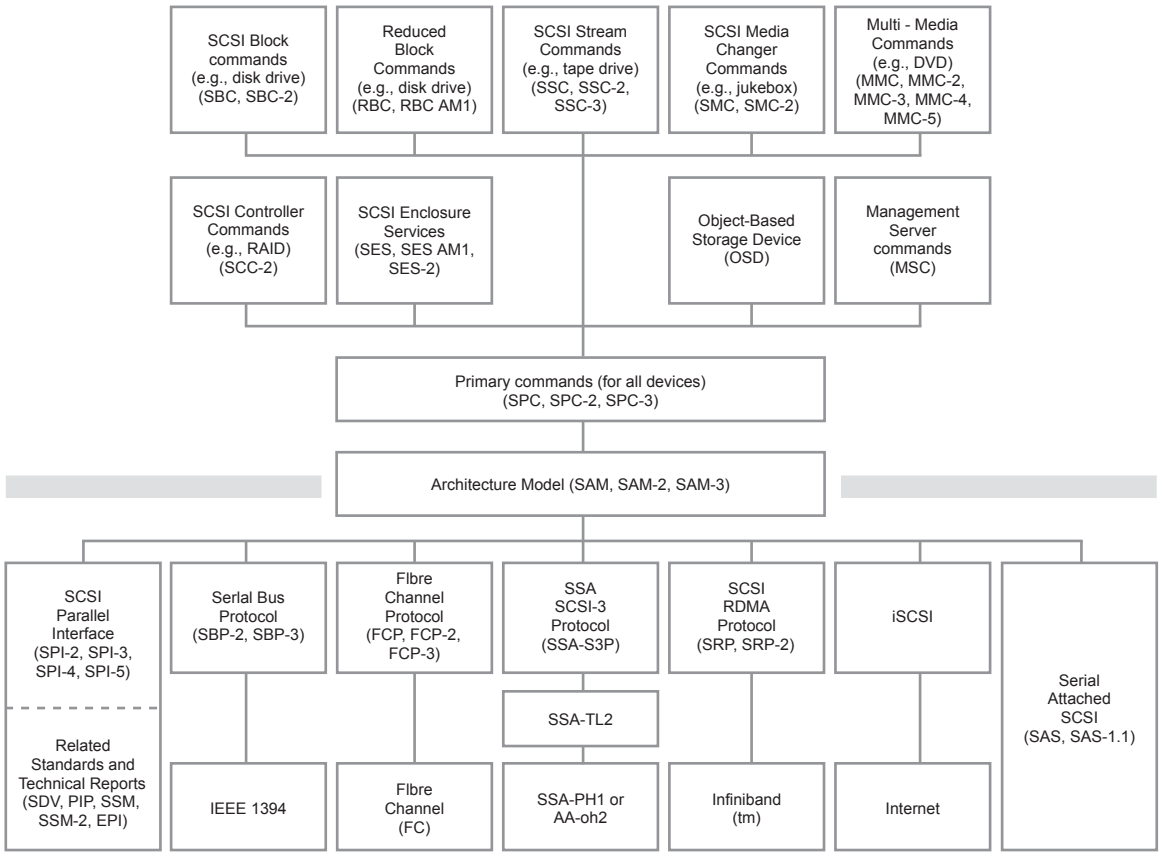


Figure 2-21: SCSI Protocol Stack Structure

Related protocols

iSCSI, iFCP, FCP, FCIP, mFCP

Sponsor Source

SCSI standards are defined by ANSI (www.ansi.org).

Reference

http://www.danbbs.dk/~dino/SCSI/
SCSI-2 Specification

ISO Protocols in OSI 7 Layers Reference Model

Description

The Open Systems Interconnection (OSI) model is a reference model developed by ISO (International Organization for Standardization) in 1984, as a conceptual framework of standards for communication in the network across different equipment and applications by different vendors. It is now considered the primary architectural model for inter-computing and internetworking communications. Most of the network communication protocols used today have a structure based on the OSI model. The OSI model defines the communications process into 7 layers, dividing the tasks involved in moving information between networked computers into seven smaller, more manageable task groups. A task or group of tasks is then assigned to each of the seven OSI layers. Each layer is reasonably self-contained so that the tasks assigned to each layer can be implemented independently. This enables the solutions offered by one layer to be updated without adversely affecting the other layers.

ISO defined a group of protocols for internetworking communications based on the OSI model, which are mostly deployed in European countries. ISO protocols are in the layers 3 to 7 and support almost any layer one and two protocols by various standard organizations and major vendors.

Key Protocols

The key ISO protocols are listed as follows:

Application	ACSE: Association Control Service Element
	CMIP: Common Management Information Protocol
	CMIS: Common Management Information Service
	CMOT: CMIP over TCP/IP
	FTAM: File Transfer Access and Management
	ROSE: Remote Operation Service Element
	RTSE: Reliable Transfer Service Element Protocol
	VTP: ISO Virtual Terminal Protocol
	X.400: Message Handling Service (ISO email transmission service) Protocols
	X.500: Directory Access Service Protocol (DAP)
Presentation Layer	ISO-PP: OSI Presentation Layer Protocol
Session Layer	ISO-SP: OSI Session Layer Protocol
Transport Layer	ISO-TP: OSI Transport Protocols: TP0, TP1, TP2, TP3, TP4
Network Layer	ISO-IP: Connectionless Network Protocol (CLNP)
	CONP: Connection-Oriented Network Protocol
	ES-IS: End System to Intermediate System Routing Exchange protocol
	IDRP: Inter-Domain Routing Protocol
	IS-IS: Intermediate System to Intermediate System

Sponsor Source

International Organization for Standardization (www.iso.org)

Reference

http://www.doc.ua.pt/arch/itu/rec/product/X.htm:
X.200:
Information technology - Open Systems Interconnection - Basic Reference Model: The basic model
X.207:
Information technology - Open Systems Interconnection - Application layer structure
X.210:
Information technology - Open systems interconnection - Basic Reference Model: Conventions for the definition of OSI services
X.211:
Information technology - Open systems interconnection - Physical service definition
X.212:
Information technology - Open systems interconnection - Data Link service definition
X.213:
Information technology - Open Systems Interconnection - Network service definition
X.214:
Information technology - Open Systems Interconnection - Transport service definition
X.215:
Information technology - Open Systems Interconnection - Session service definition
X.216:
Information technology - Open Systems Interconnection - Presentation service definition
X.800:
Security architecture for Open Systems Interconnection for CCITT applications

Application Layer

Protocol Name

ISO ACSE: Association Control Service Element

Protocol Description

The ISO Association Control Service Element (ACSE), an application layer protocol in the OSI model, is designed to establish and release an application-association between two AEIs and to determine the application context of that association. The ACSE supports two modes of communication: connection-oriented and connectionless. For the connection-oriented mode, the application-association is established and released by the reference of ACSE connection-oriented services. For the connectionless mode, the application-association exists during the invocation of the single ACSE connectionless mode service, A UNIT-DATA .

The applications in the OSI reference model represent communication between a pair of application-processes (APs) in terms of communication between their application-entities (AEs) using the presentation-service. The functionality of an AE is factored into a number of application-service-elements (ASEs). The interaction between AEs is described in terms of the use of their ASEs' services. This Service Definition supports the modeling concepts of application-association and application context.

An application-association is a cooperative relationship between two AEIs. It provides the necessary frame of reference between the AEIs in order that they may interwork effectively. This relationship is formed by the communication of application-protocol-control-information between the AEIs through their use of the presentation-service.

An application context is an explicitly identified set of application-service-elements, related options and any other necessary information for the interworking of application-entities on an application association.

The ACSE service-user is that part of an application-entity that makes use of ACSE services. It may be the Control Function (CF) or an ASE or some combination of the two.

The services provided by ACSE are listed as follows:

Communication mode	Service	Type
Connection-oriented	A-ASSOCIATE A-RELEASE A-ABORT A-P-ABORT	Confirmed Confirmed Non-confirmed Provider-initiated
Connectionless	A-UNIT-DATA	Non-confirmed

Protocol Structure

The functions, services and message structure are listed as follows:

Functional Unit	Service	APDU	Field Name
Kernel	A-ASSOCIATE	AARQ	Protocol Version Application Context Name Calling AP Title Calling AE Qualifier Calling AP Invocation-identifier Calling AE Invocation-identifier Called AP Title Called AE Qualifier Called AP Invocation-identifier Called AE Invocation-identifier Implementation Information User Information
		AARE	Protocol Version Application Context Name Responding AP Title Responding AE Qualifier Responding AP Invocation-identifier Responding AE Invocation-identifier Result Result Source-Diagnostic Implementation Information User Information
	A-RELEASE	RLRQ	Reason User Information
		RLRE	Reason User Information
	A-ABORT	ABRT	Abort Source User Information
Authentication	A-ASSOCIATE	AARQ	ACSE Requirements Authentication-mechanism Name Authentication-value
		AARE	Ditto
		ABRT	Diagnostic
Application Context Negotiation	A-ASSOCIATE	AARQ	Application Context Name List ACSE Requirements
		AARE	Ditto

Related protocols

ISO-SP, ISO-PP, ROSE, ISO-VTP

Sponsor Source

ACSE is defined in ISO (www.iso.org) documents 8650, 8649 and ITU (www.itu.org) documents X.227, X.217 andX.237.

Reference

http://www.doc.ua.pt/arch/itu/rec/product/X.htm
X.217: Information technology - Open Systems Interconnection - Service definition for the Association Control Service Element
X.227: Information technology - Open Systems Interconnection - Connection-oriented protocol for the Association Control Service Element: Protocol specification
X.237: Information technology - Open Systems Interconnection - Connectionless protocol for the Association Control Service Element: Protocol specification

Protocol Name

ISO CMIP: Common Management Information Protocol

Protocol Description

Common Management Information Protocol (CMIP) is an ISO protocol used with Common Management Information Services (CMIS), supports information exchange between network management applications and management agents. CMIS defines a system of network management information services. CMIP supplies an interface that provides functions which maybe used to support both ISO and user-defined management protocols. The CMIP specification for TCP/IP networks is called CMOT (CMIP Over TCP) and the version for IEEE 802 LAN's is called CMOL (CMIP Over LLC). CMIP/CMIS are proposed as competing protocols to the Simple Network Management Protocol (SNMP) in the TCP/IP suite.

CMIP uses an ISO reliable connection-oriented transport mechanism and has built in security that supports access control, authorization and security logs. The management information is exchanged between the network management application and management agents through managed objects. Managed objects are a characteristic of a managed device that can be monitored, modified or controlled and can be used to perform tasks.

CMIP does not specify the functionality of the network management application, it only defines the information exchange mechanism of the managed objects and not how the information is to be used or interpreted.

The major advantages of CMIP over SNMP are:

- CMIP variables not only relay information, but also can be used to perform tasks. This is impossible under SNMP.
- CMIP is a safer system as it has built in security that supports authorization, access control, and security logs.
- CMIP provides powerful capabilities that allow management applications to accomplish more with a single request.
- CMIP provides better reporting of unusual network conditions

Access to managed information in the managed objects is provided by the Common Management Information Service Element (CMISE) that uses CMIP (Common Management Information Protocol) to issue requests for management services. The management services provided by CMIP/CMISE can be organized into two distinct groups, management operation services initiated by a manager to request that an agent provide certain services or information, and notification services, used by the management agents to inform the managers that some events or set of events have occurred.

Protocol Structure

CMIP is an ASN.1 based protocol, whose PDUs (Protocol Data Units) are based on ROSE. Each service element has its PDUs which are part of the ROSE user data. The CMISE primitives and CMIP operation are listed as follows:

Correspondence between CMISE primitives and CMIP operations

CMIS primitive	Mode	Linked-ID	CMIP operation
M CANCEL GET req/ind	Confirmed	Not applicable	m-Cancel-Get-Confirmed
M CANCEL GET rsp/conf	Not applicable	Not applicable	m-Cancel-Get-Confirmed
M EVENT REPORT req/ind	Non-confirmed	Not applicable	m-EventReport
M EVENT REPORT req/ind	Confirmed	Not applicable	m-EventReport-Confirmed
M EVENT REPORT rsp/conf	Not applicable	Not applicable	m-EventReport-Confirmed
M GET req/ind	Confirmed	Not applicable	m-Get
M GET rsp/conf	Not applicable	Absent	m-Get
M GET rsp/conf	Not applicable	Present	m-Linked-Reply
M SET req/ind	Non-confirmed	Not applicable	m-Set
M SET req/ind	Confirmed	Not applicable	m-Set-Confirmed
M SET rsp/conf	Not applicable	Absent	m-Set-Confirmed
M SET rsp/conf	Not applicable	Present	m-Linked-Reply
M ACTION req/ind	Non-confirmed	Not applicable	m-Action
M ACTION req/ind	Confirmed	Not applicable	m-Action-confirmed
M ACTION rsp/conf	Not applicable	Absent	m-Action-confirmed
M ACTION rsp/conf	Not applicable	Present	m-Linked-Reply
M CREATE req/ind	Confirmed	Not applicable	m-Create
M CREATE rsp/conf	Not applicable	Not applicable	m-Create
M DELETE req/ind	Confirmed	Not applicable	m-Delete
M DELETE rsp/conf	Not applicable	Absent	m-Delete
M DELETE rsp/conf	Not applicable	Present	m-Linked-Reply

Related protocols

ISO-SP, ISO-TP, ISO-IP, ISO-PP, ROSE, ACSE, SNMP

Sponsor Source

CMIP/CMIS are defined in ISO (www.iso.org) documents 9595, 9596 and ITU (www.itu.org) X.711.

Reference

http://www.doc.ua.pt/arch/itu/rec/product/X.htm
X.711: Information technology - Open Systems Interconnection - Common management information protocol: Specification
X.700: Management framework for Open Systems Interconnection (OSI) for CCITT applications
X.701: Information technology - Open Systems Interconnection - Systems management overview
X.702: Information technology - Open Systems Interconnection - Application context for systems management with transaction processing
X.703: Information technology - Open Distributed Management Architecture

Protocol Name

CMOT: CMIP Over TCP/IP

ISO Transportation ISO 8073

Protocol Description

Common Management Information Protocol (CMIP), an ISO protocol used with Common Management Information Services (CMIS) for the monitoring and control of heterogeneous networks. CMIS defines a system of network management information services. CMIP was proposed as a replacement for the less sophisticated Simple Network Management Protocol (SNMP) but has not been widely adopted. CMIP provides improved security and better reporting of unusual network conditions.

CMIP Over TCP/IP (CMOT) is a network management protocol using ISO CMIP to manage IP-based networks.CMOT defines a network management architecture that uses the International Organization for Standardization's (ISO) Common Management Information Services/Common Management Information Protocol (CMIS/CMIP) in the Internet. This architecture provides a means by which control and monitoring information can be exchanged between a manager and a remote network element.

Protocol Structure

The following seven protocols comprise the CMOT protocol suite: ISO ACSE, ISO DIS ROSE, ISO CMIP, the lightweight presentation protocol (LPP), UDP, TCP, and IP.

Management Application Processes	
CMISE ISO 9595/9596	
ACSE ISO IS 8649/8650	ROSE ISO DIS 9072-1/2
Lightweight Presentation Protocol (LPP) RFC 1085	
TCP RFC 793	UDP RFC 768
IP/IPv6 RFC 791, RFC 2460	

The following six protocols comprise the CMIP protocol suite: ISO ACSE, ISO DIS ROSE, ISO CMIP, ISO Presentation, ISO Session and ISO Transport.

Management Application Processes	
CMISE ISO 9595/9596	
ACSE ISO IS 8649/8650	ROSE ISO DIS 9072-1/2
ISO Presentation ISO 8822 8823	
ISO Session ISO 8326 8327	

Related protocols

TCP, UDP, IP, CMIP, CMIS, ACSE, ROSE, CMISE

Sponsor Source

CMOT is defined by ISO (www.iso.org) and IETF (www.itu.org).

Reference

http://www.javvin.com/protocol/rfc1189.pdf
The Common Management Information Services and Protocols for the Internet (CMOT and CMIP)

Protocol Name

ISO FTAM: File Transfer Access and Management protocol

Protocol Description

The File Transfer Access and Management protocol (FTAM), an ISO application protocol, offers file transfer services between client (initiator) and server (responder) systems in an open environment. FTAM also provides access to files and management of files on diverse systems. Similar to FTP File Transfer Protocol) and NFS (Network File System) in the TCP/IP environment, FTAM is designed to help users access files on diverse systems that use compatible FTAM implementations.

FTAM is a system in which connection-oriented information about the user and the session is maintained by a server until the session is taken down. Files are transferred between systems by first establishing a connection-oriented session. The FTAM client contacts the FTAM server and requests a session. Once the session is established, file transfer can take place. FTAM uses the concept of a virtual filestore, which provides a common view of files. The FTAM file system hides the differences between different vendor systems. FTAM specifies document types as files with straight binary information or text files in which each line is terminated with a carriage return. Data is interpreted as records and FTAM provides the virtual filestore capabilities that store record-oriented structured files.

With FTAM, users can manipulate files down to the record level, which is how FTAM stores files. In this respect, FTAM has some relational database features. For example, users can lock files or lock individual records.

FTAM provides the following FTAM service classes:

- The transfer service class allows the exchange of files or parts of files. It generally includes the simple basic file transfer tasks, allowing single operations with a minimum number of interactions.
- The access service class allows the initiating system to perform several operations on individual FADUs or on the whole file.
- The management service class allows the user control over the virtual filestore in order to create or delete files, read and modify attributes etc..
- The transfer-and-management service class combines the capabilities of the transfer service class with those of the limited file management functional unit to support directory navigation and simple functions. (See below for the functional units.)

Protocol Structure

All commands are in the format of ASN.1 messages. FTAM defines the following functional units :

- kernel functional unit
- read functional unit
- write functional unit
- file access functional unit
- limited file management functional unit
- enhanced file management functional unit
- grouping functional unit
- FADU locking functional unit
- recovery functional unit
- restart functional unit

FTAM has the following key user facilities to operate files locally and remotely:

- FTAM appending facility (APPEND/APPLICATION_PROTOCOL=FTAM) - enables user to append one or more input files to a single output file, within or between FTAM applications.
- FTAM copying facility (COPY/APPLICATION_PROTOCOL=FTAM) - enables user to copy one or more input files to a single output file, within or between FTAM applications.
- FTAM deletion facility (DELETE/APPLICATION_PROTOCOL=FTAM) - enables user to delete files
- FTAM directory facility (DIRECTORY/APPLICATION_PROTOCOL=FTAM) - enables user to display file attributes for one or more files
- FTAM renaming facility (RENAME/APPLICATION_PROTOCOL=FTAM) - enables user to rename files.

Related protocols

IS-SP, ISO-PP, ROSE, ACSE, FTP, NFS

Sponsor Source

FTAM is defined in ISO (www.iso.org) documents 8571.

Reference

http://www.nhsia.nhs.uk/napps/step/pages/ithandbook/h232-6.htm
OSI File Transfer Access and Management (FTAM) Standard

Protocol Name

ISO ROSE: Remote Operations Service Element Protocol

Protocol Description

The ISO Remote Operations Service Element Protocol (ROSE) is a protocol that provides remote operation capabilities, allows interaction between entities in a distributed application, and upon receiving a remote operations service request, allows the receiving entity to attempt the operation and report the results of the attempt to the requesting entity. The ROSE protocol itself is only a vehicle for conveying the arguments and results of the operation as defined by the application.

In the OSI environment, communication between application processes is represented in terms of communication between a pair of application entities (AEs) using the presentation service. Communication between some application-entities is inherently interactive. Typically, one entity requests that a particular operation be performed; the other entity attempts to perform the operation and then reports the outcome of the attempt. The generic structure of an operation is an elementary request/reply interaction. Operations are carried out within the context of an application-association.

Operations invoked by one AE (the invoker) are performed by the other AE (the performer). Operations may be classified according to whether the performer of an operation is expected to report its outcome. Operations may also be classified according to two possible operation modes: synchronous, in which the invoker requires a reply from the performer before invoking another operation; and asynchronous, in which the invoker may continue to invoke further operations without awaiting a reply.

The remote-operation-protocol-machine (ROPM) communicates with its service-user by means of primitives. Each invocation of the ROPM controls a single application-association. The ROPM is driven by ROSE service request primitives from its service-user, and by indication and confirm primitives of the RTSE services, or the presentation-service. The ROPM, in turn, issues indication primitives to its service-user, and request primitives on the RTSE services being used, or on the presentation-service.

The reception of an ROSE service primitive, or of an RTSE service or of a presentation-service primitive, and the generation of dependent actions are considered to be individual. During the exchange of APDUs, the existence of both, the association-initiating AE and the association- responding AE is presumed. During the execution of operations, the existence of an application-association between the peer AEs is presumed.

ROSE services summary

Service	Type
RO-INVOKE	Non-confirmed
RO-RESULT	Non-confirmed
RO-ERROR	Non-confirmed
RO-REJECT-U	Non-confirmed
RO-REJECT-P	Provider-initiated

Protocol Structure

ROSE messages:

ROSE Incoming event list

Abbreviated name	Source	Name and description
AA-ESTAB	RTSE	positive RT-OPEN response primitive or positive RT-OPEN confirm primitive
	ACSE	positive A-ASSOCIATE response primitive or positive A-ASSOCIATE
RO-INVreq	ROSE-user	confirm-primitive
RO-RESreq	ROSE-user	RO-INVOKE request primitive
RO-ERRreq	ROSE-user	RO-RESULT request primitive
RO-RJUreq	ROSE-user	RO-ERROR request primitive
ROIV	ROPM-peer	RO-REJECT-U request primitive
RORS	ROPM-peer	valid RO-INVOKE APDU as user data on a TRANSind event
ROER	ROPM-peer	valid RO-RESULT APDU as user data on a TRANSind event
RORJu	ROPM-peer	valid RO-ERROR APDU as user data on a TRANSind event
RORJp	ROPM-peer	valid RO-REJECT APDU (user -reject) as user data on a TRANSind event
APDUua	ROPM-peer	valid RO-REJECT APDU (provider-reject
TRANSind	ROPM-TR	with General-problem) as user data
TRANSreq	ROPM	on a TRANSind event
P-DATAind	PS-provider	unacceptable APDU as user data on a TRANSind event
RT-TRind	RTSE	transfer indication of an APDU
RT-TRcnf+	RTSE	transfer request for an APDU
RT-TRcnf-	RTSE	P-DATA indication primitive
RT-TPind	RTSE	RT-TRANSFER indication primitive
RT-TGind	RTSE	positive RT-TRANSFER confirm primitive
AA-REL	RTSE	negative RT-TRANSFER confirm primitive
	ACSE	RT-TURN-PLEASE indication primitive
AA-ABreq	ROPM	RT-TURN-GIVE indication primitive
AA-ABind	ROPM-TR	RT-CLOSE response primitive or RT-CLOSE confirm primitive
ABORTind	RTSE	positive A-RELEASE response primitive or
	ACSE	A-RELEASE confirm primitive
		abort application-association
		application-association aborted
		RT-P-ABORT indication primitive or the RT-U-ABORT indication primitive
		A-ABORT indication primitive or A-P-ABORT indication primitive

ROSE Outgoing event list

Abbreviated name	Target	Name and description
RO-INVind	ROSE-user	RO-INVOKE indication primitive
RO-RESind	ROSE-user	
RO-ERRind	ROSE-user	
RO-RJUind	ROSE-user	RO-RESULT indication primitive
RO-RJPind	ROSE-user	
ROIV	ROPM-peer	RO-ERROR indication primitive
RORS	ROPM-peer	RO-REJECT-U indication primitive
ROER	ROPM-peer	RO-REJECT-P indication primitive
RORJu	ROPM-peer	RO-INVOKE APDU as user data on a TRAN-
RORJp	ROPM-peer	Sreq event
TRANSreq	ROPM-TR	RO-RESULT APDU as user data on a
TRANSind	ROPM	TRANSreq event
P-DATAreq	PS-provider	RO-ERROR APDU as user data on a TRAN-
RT-TRreq	RTSE	Sreq event
RT-TPreq	RTSE	RO-REJECT user-reject APDU as user-data
RT-TGreq	RTSE	on a TRANSreq event
AA-ABreq	ROPM-TR	RO-REJECT provider-reject APDU as user
AA-ABind	ROPM	data on a TRANSreq event
ABORTreq	RTSE	transfer request for an APDU
	ACSE	transfer indication of an APDU
		P-DATA request primitive
		RT-TRANSFER request primitive
		RT-TURN-PLEASE request primitive
		RT-TURN-GIVE request primitive
		abort application-association
		application-association aborted
		RT-U-ABORT request primitive
		A-ABORT request primitive

Related protocols

ISO-PP, ISO-SP, ACSE

Sponsor Source

ROSE is defined in ISO (www.iso.org) documents 9072 and ITU (www.itu.org) documents X.229 and X.219.

Reference

http://www.doc.ua.pt/arch/itu/rec/product/X.htm
X.219: Remote Operations: Model, notation and service definition
X.229: Remote Operations: Protocol specification

Protocol Name

ISO RTSE: Reliable Transfer Service Element Protocol

Protocol Description

Reliable Transfer Service Element (RTSE), an ISO application layer protocol, provides for the reliable transfer of bulk data by transforming the data into a string of octets, then breaking the string into segments and handing each segment to the Presentation Layer for delivery. Checkpoints are established between segments. Through the services of the Presentation Layer, RTSE uses the activity management services of the Session Layer to manage the transfer of the collection of segments that makes up the bulk data. Activity and minor synchronization facilities of the Session Layer support interruption and possible resumption of data transfer if the underlying network connection is lost.

RTSE is used in the X.400 Message Handling Service (MHS) and is available for use by ROSE when remote operations require reliable transfer. Because of its use in X.400, RTSE is widely available.

Typically the Transport layer is supposed to ensure reliable delivery but this is insufficient for two reasons: 1) No class of transport protocol will recover from a failure of the underlying network, which results in the required QOS (Quality of Service) not being met. Under these circumstances, the underlying connection will be lost. 2) For historical reasons, MHS was designed to operate over TP0 which provides no recovery at all from signalled errors (including X.25 resets). In the event of either an X.25 reset or a disconnect, TP0 terminates the underlying connection

RTSE is required to re-establish the underlying failed connection and to repeat the transmission attempt, transparently to the user. However, RTSE cannot guarantee delivery if success cannot be achieved within a given time, RTSE will report failure. This may occur if there is a catastrophic failure either of the underlying network or of the peer application, which clearly neither RTSE nor any other ASE (Application Service Element) can do anything positive about.

RTSE is not viable on its own. RTSE has no knowledge of the context of the PDU which it is attempting to deliver, nor indeed would it have anything to deliver. There must be an 'RTSE user' which understands what RTSE is being used for, typically a MHS service element using ROSE.

RTSE uses Session Layer Activities for the following reasons: Each PDU (e.g. message) and the response confirming (or otherwise) its successful delivery are encapsulated within dialogue units (major synchronisation points). RTSE may also insert minor synchronization points at suitable intervals during the activity, as it sees fit. An activity may be interrupted in the event of minor errors occurring, and can be resumed later. In the case of more severe errors, such as loss of the application association itself, the activity may need to be discarded and in this case the transaction will start again from scratch at a later time, in a new activity.

Protocol Structure

RTSE Service Summary

Service	Type
RT-OPEN	Confirmed
RT-CLOSE	Confirmed
RT-TRANSFER	Confirmed
RT-TURN-PLEASE	Non-confirmed
RT-TURN-GIVE	Non-confirmed
RT-P-ABORT	Provider-initiated
RT-U-ABORT	Non-confirmed

Related protocols

ISO-SP, ISO-PP, ROSE, ACSE, X.400

Sponsor Source

RTSE is defined in ISO (www.iso.org) documents 9066 and ITU (www.itu.org) documents X.228 and X.218.

Reference

http://www.doc.ua.pt/arch/itu/rec/product/X.htm
X.218: Reliable Transfer: Model and service definition
X.228: Reliable Transfer: Protocol specification

Protocol Name

ISO VTP: ISO Virtual Terminal (VT) Protocol

Protocol Description

The ISO Virtual Terminal (VT) service and protocol (VTP) allows a host application to control a terminal with screen and keyboard and similar devices like printers. In addition, VTP also supports the less common application-application and terminal-terminal communication.

VTP is comparable to Telnet in the TCP/IP suite but more powerful. VTP also includes control of cursor movement, colors, character sets and attributes, access rights, synchronization, multiple pages, facility negotiation, etc. This means that the huge number of classic terminal type definitions (e.g. in UNIX termcap or terminfo) are unnecessary at each host in the net, as the VT protocol includes the corresponding commands for one abstract virtual terminal that only have to be converted by the local implementation to the actual terminal control sequences. Consequently, the use of VT means not every host needs to know every type of terminal.

As with most ISO standards that require general consensus amongst participating members, the OSI VT has many optional capabilities, two modes of operation and an almost infinite number of implementation-specific options. Profiles may help in reducing the optionality present (e.g., there exists a Telnet profile for VT), but it is doubtful if the OSI VT can completely put an end to the ‘m x n’ terminal incompatibility problem that exists in a heterogeneous computer network.

Related protocols

ISP-PP, Telnet

Sponsor Source

ISO VTP is defined in ISO (www.iso.org) documents 9040, 9041.

Protocol Name

X.400: Message Handling Service Protocol

Protocol Description

X.400 is the Message Handling Service protocol for e-mail transmission specified by the ITU-T and ISO. X.400 is common in Europe and Canada and is an alternative to the more popular e-mail protocol, Simple Mail Transfer Protocol (SMTP), which is defined by IETF. X.400 uses a binary format so it is easy to include binary contents without encoding it for transfer. Also it is harder for people to fake email addresses and contents, than with STMP where text messages are used.

X.400 and STMP have similar features but also unique features in themselves. Generally speaking, X.400 is a more complex protocol with the following features that are not in the SMTP:

- Delivery notifications - Delivery notifications are used both about delivery notifications (yes, the message got here) and about non-delivery notifications (no, the message did not get there).
- Receipt notifications - A receipt notification is passed back to the originating user indicating what happened to the message after it was delivered (for instance that it was read by the recipient). In X.400, receipt and non-receipt notifications may include notifications of something being automatically forwarded, messages deleted, etc.
- Security functions – X.400 defines a framework for mail transmission securities. It defines the concept of a “security label” and allows using an OID for identifying your security labeling scheme, but no labeling scheme is actually specified in the protocol.
- Priority markers (3 levels) – This feature is used for ordering the queue of mails to send, so that “important” mails get sent before “less important” mails.
- Deferred delivery – Schedule delivery time for messages. This feature has not been widely deployed.
- Conversion in the network –such as converting Teletex to plain text, or fax images to text saying “there was a picture here, but you are not allowed to see it”. Conversion never improves a message, and it is impossible to support security functions like signatures or encryption while doing conversion in the network.
- Reliable Transfer Service – This X.400 feature gives the ability to continue transferring a document after the transfer is interrupted.

SMTP has some functions that X.400 does not, including the following:

- Standard functionality to check each recipient for validity before transferring the message; X.400 requires transferring the complete message before checking recipients.
- Optional functionality for checking whether a message is too large to transfer before sending it.
- Ability to insert any data into the header of a message with a fair probability of it being presented to the user
- Ability (MIME Multipart/Alternative) to send several representations of the same content in the same message, guaranteeing both interoperability with the lowest common denominator and no loss of information between compatible UAs.

In addition, an X.400 address is different from that of an STMP. X.400 consists of a set of bindings for country (c), administrative domain (a), primary management domain (p), surname (s), and given name (g). An SMTP e-mail address that looks like this hypothetical address:

Jeff.warson@javvin.subdomain.us

looks like this in an X.400 e-mail message:

G=Jeff; S=warson; O=subdomain; OU=javvin; PRMD=attmail; ADMD=attmail; C=US

Protocol Structure

X.400 was designed with attributed addresses. The complete set of attributes is rather large:

Attribute Type	Abbreviation	Label
Given Name	Given Name	G
Initial	Initials	I
Surname	Surname	S
Generation Qualifier	Generation	Q
Common Name	Common Name	CN
Organization	Organization	O
Organizational Unit 1	Org.Unit.1	OU1
Organizational Unit 2	Org.Unit.2	OU2
Organizational Unit 3	Org.Unit.3	OU3
Organizational Unit 4	Org.Unit.4	OU4
Private Management Domain Name	PRMD	P
Administration Management Domain Name	ADMD	A
Country	Country	C
Physical Delivery Personal Name	PD-person	PD-PN
Extension of Postal O/R Address Components	PD-ext. address	PD-EA
Extension of Physical Delivery Address Components	PD-ext. delivery	PD-ED
Physical Delivery Office Number	PD-office number	PD-OFN

Physical Delivery Office Name	PD-office	PD-OF
Physical Delivery Organization Name	PD-organization	PD-O
Street Address	PD-street	PD-S
Unformatted Postal Address	PD-address	PD-A1
(there are individual labels for each line of the address)		PD-A2
		PD-A3
		PD-A4
		PD-A5
		PD-A6
Unique Postal Name	PD-unique	PD-U
Local Postal Attributes	PD-local	PD-L
Postal Restante Address	PD-restante	PD-R
Post Office Box Address	PD-box	PD-B
Postal Code	PD-code	PD-PC
Physical Delivery Service Name	PD-service	PD-SN
Physical Delivery Country Name	PD-country	PD-C
X.121 Network Address	X.121	X.121
E.163/E.164 Network Address	ISDN	ISDN
PSAP Network Address	PSAP	PSAP
User Agent Numeric ID	N-ID	N-ID
Terminal Identifier	T-ID	T-ID
Terminal Type	T-TY	T-TY
Domain Defined AttributE	DDA:	DDA:

Related protocols

SMTP, MIME, IMAP/IMAP4, POP/POP3

Sponsor Source

X.400 protocol is defined by ISO (www.iso.org) and ITU-T (www.itu.org).

Reference

http://www.itu.int/rec/recommendation.asp?type=products&parent=T-REC-f
X.400 Standards List
http://www.doc.ua.pt/arch/itu/rec/product/X.htm
X.402: Information technology - Message Handling Systems (MHS) - Overall Architecture
X.404: Information technology - Message Handling Systems (MHS): MHS routing - Guide for messaging systems managers

Protocol Name

X.500: Directory Access Protocol (DAP)

Protocol Description

X.500, the directory Access Protocol by ITU-T (X.500) and also ISO (ISO/IEC 9594), is a standard way to develop an electronic directory of people in an organization so that it can be part of a global directory available to anyone in the world with Internet access.

In the X.500 directory architecture, the client queries and receives responses from one or more servers in the server Directory Service with the Directory Access Protocol (DAP) controlling the communication between the client and the server

A Directory System Agent (DSA) is the database in which the directory information is stored. This database is hierarchical in form, designed to provide fast and efficient search and retrieval. The DSAs are interconnected from the Directory Information Tree (DIT). The user interface program for access to one or more DSAs is a Directory User Agent (DUA). DUAs include whois, finger, and programs that offer a graphical user interface.

The Directory System Protocol (DSP) controls the interaction between two or more Directory System Agents, and between a Directory User Agent and a Directory System Agent. This is done in such a way that an end user can access information in the Directory without needing to know the exact location of that specific piece of information.

X.500 offers the following key features:

- Decentralized Maintenance: Each site running X.500 is responsible ONLY for its local part of the Directory, so updates and maintenance can be done instantly.
- Powerful Searching Capabilities: X.500 provides powerful searching facilities that allow users to construct arbitrarily complex queries.
- Single Global Namespace: Much like the DNS, X.500 provides a single homogeneous namespace to users. The X.500 namespace is more flexible and expandable than the DNS.
- Structured Information Framework: X.500 defines the information framework used in the Directory, allowing local extensions.
- Standards-Based Directory Services: As X.500 can be used to build a standards-based directory, applications which require directory information (e-mail, automated resources locators, special-purpose directory tools) can access a planet's worth of information in a uniform manner.

X.500 is criticized as being too complex for most implementations. To address the issue, the University of Michigan developed a simpler TCP/IP-based version of DAP, the Lightweight Directory Access Protocol (LDAP), for use on the Internet. LDAP offers much of the same basic functionality as DAP and can be used to query data from proprietary directories as well as from an open X.500 service. Within the past year, most major suppliers of e-mail and directory-services software have expressed interest in LDAP, which is fast becoming a de facto directory protocol for the Internet.

Protocol Structure

X.500 has a complex data structure in the directory database and for its communication protocols such as DAP. One should read the specification documents from ISO and ITU.

Related protocols

LDAP, DNS, Finger

Sponsor Source

X.500 (DAP) protocol is defined by ISO (www.iso.org) and ITU-T (www.itu.org).

Reference

http://www.javvin.com/protocol/rfc1308.pdf
Executive Introduction to Directory Services Using the X.500 Protocol
http://www.javvin.com/protocol/rfc1309.pdf
Technical Overview of Directory Services Using the X.500 Protocol

Protocol Name

ISO-PP: OSI Presentation Layer Protocol

Protocol Description

The OSI presentation layer protocol (ISO-PP) is for the information transit between open systems using connection oriented or connectionless mode transmission at the presentation layer of the OSI 7 layer model. An application protocol is specified in terms of the transfer of presentation data values between application entities (PS users), using the User data parameter of presentation service primitives.

The Presentation Layer has two functions it carries out on behalf of PS users:

a) negotiation of transfer syntaxes;
b) transformation to and from transfer syntax.

The function of transfer syntax negotiation is supported by presentation protocols. Transformation of syntax is a function contained within a presentation entity and has no impact on presentation protocol design. For connectionless mode transmission, the sending presentation entity selects the transfer syntaxes. No transfer syntax negotiation occurs.

A set of presentation data value definitions associated with an application protocol constitutes an abstract syntax. For two application entities to communicate successfully they must have an agreement on the set of abstract syntaxes they intend to use. During the course of communication they may decide to modify this agreement. As a consequence, the set of abstract syntaxes in use may be changed. The abstract syntax specification identifies the information content of the set of presentation data values. It does not identify the transfer syntax to be used while presentation data values are transferred between presentation entities, nor is it concerned with the local representation of presentation data values.

The Presentation Layer exists to ensure that the information content of presentation data values is preserved during transfer. It is the responsibility of cooperating application entities to determine the set of abstract syntaxes they employ in their communication and inform the presentation entities of this agreement. Knowing the set of abstract syntaxes to be used by the application entities, the presentation entities are responsible for selecting mutually acceptable transfer syntaxes that preserve the information content of presentation data values.

For connectionless mode transmission, the abstract syntaxes used are determined by the sending application entity. For successful communication to take place, these must be acceptable to the receiving application entity.

For connectionless mode transmission, the presentation entities do not negotiate transfer syntaxes. The transfer syntaxes used are determined by the sending application entity. For successful communication to take place, these must be acceptable to the receiving application entity. The abstract syntaxes and the associated transfer syntaxes may be explicitly stated in the “Presentation context definition list” parameter as a user option.

Presentation entities support protocols that enhance the OSI session service in order to provide a presentation service. The PS user is provided with access to the session service which permits full use to be made of that service. This includes negotiation of and access to the session functional units. The role of the Presentation Layer in providing this access includes representation of presentation data values in the User data parameters of session service primitives.

Protocol Structure

The major presentation primitives:

Connection Release Primitive	Token Handling Primitive
P-RELEASE request P-RELEASE indication P-RELEASE response P-RELEASE confirm	P-TOKEN-GIVE request P-TOKEN-GIVE indication P-TOKEN-PLEASE request P-TOKEN-PLEASE indication P-CONTROL-GIVE request P-CONTROL-GIVE indication

Presentation Exception Reporting Primitive	Activity Management Primitive
P-P-EXCEPTION-REPORT indication P-U-EXCEPTION-REPORT request P-U-EXCEPTION-REPORT indication	P-ACTIVITY-START request P-ACTIVITY-START indication P-ACTIVITY-RESUME request P-ACTIVITY-RESUME indication P-ACTIVITY-INTERRUPT request P-ACTIVITY-INTERRUPT indication P-ACTIVITY-NTERRUPT response P-ACTIVITY-INTERRUPT confirm P-ACTIVITY-DISCARD request P-ACTIVITY-DISCARD indication P-ACTIVITY-DISCARD response P-ACTIVITY-DISCARD confirm P-ACTIVITY-END request P-ACTIVITY-END indication P-ACTIVITY-END response P-ACTIVITY-END confirm
Synchronization Services Primitive	
P-SYNC-MINOR request P-SYNC-MINOR indication P-SYNC-MINOR response P-SYNC-MINOR confirm P-SYNC-MAJOR request P-SYNC-MAJOR indication P-SYNC-MAJOR response P-SYNC-MAJOR confirm	

Related protocols

ISO-TP, ISO-SP, ACSE, ROSE

Sponsor Source

The ISO-PP (OSI Presentation Layer protocol) is defined in ISO (www.iso.org) documents 8823, 8822 and ITU (www.itu.org) documents X.226, X.216 and X.236.

Reference

http://www.doc.ua.pt/arch/itu/rec/product/X.htm
X.226: Information technology - Open Systems Interconnection - Connection-oriented
Presentation protocol: Protocol specification
X.216: Information technology - Open Systems Interconnection - Presentation service definition
X.236: Information technology - Open Systems Interconnection - Connectionless Presentation protocol: Protocol specification

Protocol Name

ISO-SP: OSI Session Layer Protocol

Protocol Description

The OSI Session Layer Protocol (ISO-SP) provides session management, e.g. opening and closing of sessions. In case of a connection loss it tries to recover the connection. If a connection is not used for a longer period, the session layer may close it down and re-open it for next use. This happens transparently to the higher layers. The Session layer provides synchronization points in the stream of exchanged packets.

The Session Protocol Machine (SPM), an abstract machine that carries out the procedures specified in the session layer protocol, communicates with the session service user (SS-user) through an session-service-access-point (SSAP) by means of the service primitives. Service primitives will cause or be the result of session protocol data unit exchanges between the peer SPMs using a transport connection. These protocol exchanges are effected using the services of the transport layer.

Session connection endpoints are identified in end systems by an internal, implementation dependent, mechanism so that the SS-user and the SPM can refer to each session connection.

The functions in the Session Layer are those necessary to bridge the gap between the services available from the Transport Layer and those offered to the SS-users.

The functions in the Session Layer are concerned with dialogue management, data flow synchronization, and data flow resynchronization.

These functions are described below; the descriptions are grouped into those concerned with the connection establishment phase, the data transfer phase, and the release phase.

Protocol Structure

ISO Session Layer Protocol Messages:

Functional unit	SPDU code	SPDU name
Kernel	CN	CONNECT
	OA	OVERFLOW ACCEPT
	CDO	CONNECT DATA OVERFLOW
	AC	ACCEPT
	RF	REFUSE
	FN	FINISH
	DN	DISCONNECT
	AB	ABORT
	AA	ABORT ACCEPT
	DT	DATA TRANSFER
	PR	PREPARE
Negotiated release	NF	NOT FINISHED
	GT	GIVE TOKENS
	PT	PLEASE TOKENS
Half-duplex	GT	GIVE TOKENS
	PT	PLEASE TOKENS
Duplex		No additional associated SPDUs
Expedited data	EX	EXPEDITED DATA
Typed data	TD	TYPED DATA
Capability data exchange	CD	CAPABILITY DATA
	CDA	CAPABILITY DATA ACK
Minor synchronize	MIP	MINOR SYNC POINT
	MIA	MINOR SYNC ACK
	GT	GIVE TOKENS
	PT	PLEASE TOKENS
Symmetric synchronize	MIP	MINOR SYNC POINT
	MIA	MINOR SYNC ACK
Data separation		No additional associated SPDUs
Major synchronize	MAP	MAJOR SYNC POINT
	MAA	MAJOR SYNC ACK
	PR	PREPARE
	GT	GIVE TOKENS
	PT	PLEASE TOKENS
Resynchronize	RS	RESYNCHRONIZE
	RA	RESYNCHRONIZE ACK
	PR	PREPARE
Exceptions	ER	EXCEPTION REPORT
	ED	EXCEPTION DATA
Activity management	AS	ACTIVITY START
	AR	ACTIVITY RESUME
	AI	ACTIVITY INTERRUPT
	AIA	ACTIVITY INTERRUPT ACK
	AD	ACTIVITY DISCARD
	ADA	ACTIVITY DISCARD ACK
	AE	ACTIVITY END
	AEA	ACTIVITY END ACK
	PR	PREPARE
	GT	GIVE TOKENS
	PT	PLEASE TOKENS
	GTC	GIVE TOKENS CONFIRM
	GTA	GIVE TOKENS ACK

Related protocols

ISO-TP, ISO-PP, CONP, CLNP

Sponsor Source

The ISO-SP (OSI Session Layer protocol) is defined in ISO (www.iso.org) documents 8326 and 8327 and ITU (www.itu.org) documents X.215, X.225, X.235.

Reference

http://www.doc.ua.pt/arch/itu/rec/product/X.htm
X.215: Information technology - Open Systems Interconnection - Session service definition
X.225: Information technology - Open Systems Interconnection - Connection-oriented Session protocol: Protocol specification
X.235: Information technology - Open Systems Interconnection - Connectionless Session protocol: Protocol specification

Protocol Name

ISO-TP: OSI Transport Layer Protocols TP0, TP1, TP2, TP3, TP4

Protocol Description

The OSI Transport layer protocol (ISO-TP) manages end-to-end control and error checking to ensure complete data transfer. It performs transport address to network address mapping, makes multiplexing and splitting of transport connections, and also provides functions such as Sequencing, Flow Control and Error detection and recover

Five transport layer protocols exist in the ISO-TP, ranging from Transport Protocol Class 0 through Transport Protocol Class 4 (TP0, TP1, TP2, TP3 & TP4). The protocols increase in complexity from 0-4. TP0-3 works only with connection-oriented communications, in which a session connection must be established before any data is sent; TP4 also works with both connection-oriented and connectionless communications.

Transport Protocol Class 0 (TP0) performs segmentation (fragmentation) and reassembly functions. TP0 discerns the size of the smallest maximum protocol data unit (PDU) supported by any of the underlying networks, and segments the packets accordingly. The packet segments are reassembled at the receiver.

Transport Protocol Class 1 (TP1) performs segmentation (fragmentation) and reassembly, plus error recovery. TP1 sequences protocol data units (PDUs) and will retransmit PDUs or reinitiate the connection if an excessive number of PDUs are unacknowledged.

Transport Protocol Class 2 (TP2) performs segmentation and reassembly, as well as multiplexing and demultiplexing of data streams over a single virtual circuit.

Transport Protocol Class 3 (TP3) offers error recovery, segmentation and reassembly, and multiplexing and demultiplexing of data streams over a single virtual circuit. TP3 also sequences PDUs and retransmits them or reinitiates the connection if an excessive number are unacknowledged.

Transport Protocol Class 4 (TP4) offers error recovery, performs segmentation and reassembly, and supplies multiplexing and demultiplexing of data streams over a single virtual circuit. TP4 sequences PDUs and retransmits them or reinitiates the connection if an excessive number are unacknowledged. TP4 provides reliable transport service and functions with either connection-oriented or connectionless network service. TP4 is the most commonly used of all the OSI transport protocols and is similar to the Transmission Control Protocol (TCP) in the TCP/IP suite.

Both TP4 and TCP are built to provide a reliable connection oriented end-to-end transport service on top of an unreliable network service. The network service may loose packets, store them, deliver them in the wrong order or even duplicate packets. Both protocols have to be able to deal with the most severe problems e.g. a subnetwork stores valid packets and sends them at a later date. TP4 and TCP have a connect, transfer and a disconnect phase. The principles of doing this are also quite similar.

One difference between TP4 and TCP that should be mentioned is that TP4 uses ten different TPDU (Transport Protocol Data Unit) types whereas TCP knows only one. This makes TCP simple but every TCP header has to have all possible fields and therefore the TCP header is at least 20 bytes long whereas the TP4 header maybe as little as 5 bytes. Another difference is the way both protocols react in case of a call collision. TP4 opens two bidirectional connections between the TSAPs whereas TCP opens just one connection. TP4 uses a different flow control mechanism for its messages. It also provides means for quality of service measurement.

Protocol Structure

The OSI transport protocols are quite complicated in terms of their structure, which has 10 different types, each with its own header and PDU structure. The ten types are:

CR - Connection Request. The header of this type of message has 7 bytes and the length of the entire TPDU is a variable.

CC- Connection Confirm. The header of this type of message has 7 bytes and the length of the entire TPDU is a variable.

DR – Disconnect Request. The header of this type of message has 7 bytes and the length of the entire TPDU is a variable.

DC – Disconnect Confirm. The header of this type of message has 6 bytes and the length of the entire TPDU is a variable.

DT – Data TPDU. The header of this type of message has 3 bytes and the length of the entire TPDU is a variable.

ED – Expedited Data TPDU. The header of this type of message has 5 bytes and the length of the entire TPDU is a variable.

DA – Data Acknowledgement TPDU. The header of this type of message has 5 bytes and the length of the entire TPDU is a variable.

EA – Expedited Data Acknowledgement TPDU. The header of this type of message has 5 bytes and the length of the entire TPDU is a variable.

RT – Reject TPDU. The header of this type of message has 5 bytes.

ER – Error TPDU. The header of this type of message has 5

bytes and the length of the entire TPDU is a variable.

Related protocols

IS-IS, CLNP, IDRP, CONP, ES-IS, ISO-SP, ISO-PP

Sponsor Source

CONP is defined in ISO (www.iso.org) 8208 and 8878 and ITU (www.itu.org) X.214, X.224 and X.234.

Reference

http://www.doc.ua.pt/arch/itu/rec/product/X.htm
X.214: Open System Interconnection Protocols Information technology - Open Systems Interconnection - Transport service definition
X.224: Information technology - Open Systems Interconnection - Protocol for providing the connection-mode transport service
X.234: Information technology - Protocol for providing the OSI connectionless-mode transport service

Network Layer

Protocol Name

CLNP: Connectionless Network Protocol (ISO-IP)

Protocol Description

Connectionless Network Protocol (CLNP) is an ISO network layer datagram protocol by the layers defined in the Reference Model for Open Systems Interconnection (ISO 7498). CLNP provides fundamentally the same underlying service to a transport layer as IP in the TCP/IP environment. Therefore, CLNP is also called ISO-IP. Another OSI protocol in the network layer is CONP (Connection-Oriented Network Protocol), which provides connection-oriented services at the network layer.

CLNP may be used between network-entities in end systems or in Network Layer relay systems (or both). CLNP provides the Connectionless-mode Network Service. CLNP is intended for use in the Subnetwork Independent Convergence Protocol (SNICP) role, which operates to construct the OSI Network Service over a defined set of underlying services, performing functions necessary to support the uniform appearance of the OSI Connectionless-mode Network Service over a homogeneous or heterogeneous set of interconnected subnetworks. CLNP is defined to accommodate variability where Subnetwork Dependent Convergence Protocols and/or Subnetwork Access Protocols do not provide all of the functions necessary to support the Connectionless-mode Network Service over all or part of the path from one NSAP to another. CLNP may also be used to fulfill other roles and may therefore be used in the context of other approaches to subnetwork interconnection.

CLNP uses NSAP addresses and titles to identify network devices. The Source Address and Destination Address parameters are OSI Network Service Access Point Addresses (NSAP addresses). A network-entity title is an identifier for a network-entity in an end-system or intermediate-system. Network-entity titles are allocated from the same name space as NSAP addresses, and the determination of whether an address is an NSAP address or a network-entity title depends on the context in which the address is interpreted.

CLNP provides the same maximum datagram size as IP, and for those circumstances where datagrams may need to traverse a network whose maximum packet size is smaller than the size of the datagram, CLNP provides mechanisms for fragmentation (data unit identification, fragment/total length and offset). Like IP, a checksum computed on the CLNP header provides a verification that the information used in processing the CLNP datagram has been transmitted correctly, and a lifetime control mechanism (“Time to Live”) imposes a limit on the amount of time a datagram is allowed to remain in the Internet system.

Protocol Structure

CLNP has the following PDU structure:

Header Part	Address Part	Segmentation Part	Option Part	Data

CLNP PDU header:

8	16	24	32	35	40	56	72bit
NLP ID	Length ID	Version	Lifetime	Flags	Type	Seg. Length	Checksum

- NLP ID - Network Layer Protocol Identifier. The value of this field is set to binary 1000 0001 to identify this Network Layer protocol as ISO 8473, Protocol for Providing the Connectionless- mode Network Service. The value of this field is set to binary 0000 0000 to identify the Inactive Network Layer protocol subset.
- Length ID - Length Indicator is the length in octets of the header
- Version - Version/Protocol ID Extension identifies the standard Version of ISO 8473
- Lifetime – PDU Lifetime representing the remaining lifetime of the PDU, in units of 500 milliseconds.
- Flags – three flags: segmentation permitted, more segments, error report
- Type – The Type code field identifies the type of the protocol data unit, which could be data PDU or Error Report PDU
- Seg. Length - The Segment Length field specifies the entire length, in octets, of the Derived PDU, including both header and data (if present).
- Checksum - The checksum is computed on the entire PDU header.
- Address Part – It contains information of destination and source addresses, which are defined in OSI 8348/AD2 with variable length.
- Segmentation Part - If the Segmentation Permitted Flag in the Fixed Part of the PDU Header (Octet 4, Bit 8) is set to one, the segmentation part of the header, illustrated in Figure 6, must be present: If the Segmentation Permitted flag is set to zero, the non-segmenting protocol subset is in use.
- Option Part - The options part is used to convey optional parameters.
- Data Part - The Data part of the PDU is structured as an ordered multiple of octets.

Related protocols

IS-IS, CLNP, IDRP, CONP, ES-IS, ISO-TP

Sponsor Source

CLNP is defined by ISO (www.iso.org) in document 8473 and ITU (www.itu.org) X.213 and X.233.

Reference

http://www.javvin.com/rfc994.pdf
Final Text of DIS 8473, Protocol for Providing the Connectionless-mode Network Service
http://www.doc.ua.pt/arch/itu/rec/product/X.htm
X.213: Information technology – Open Systems Interconnection – Network service definition
X.233: Information technology - Protocol for providing the connectionless-mode network service: Protocol specification

Protocol Name

ISO CONP: Connection-Oriented Network Protocol

Protocol Description

Connection-Oriented Network Protocol (CONP) is an OSI network layer protocol that carries upper-layer data and error indications over connection-oriented links. Two types of OSI network layer services are available to the OSI transport layer:

- Connectionless Network Service (CLNS)—CLNS performs datagram transport and does not require a circuit to be established before data is transmitted.
- Connection-Mode Network Service (CMNS)—CMNS requires explicit establishment of a path or circuit between communicating transport layer entities before transmitting data.

CONP, based on the X.25 Packet-Layer Protocol (PLP), provides the interface between CMNS and upper layers. It is a network layer service that acts as the interface between the transport layer and CMNS. Six services are provided to transport-layer entities: one for connection establishment, one for connection release, and four for data transfer. Services are invoked by some combination of four primitives: request, indication, response, and confirmation.

There are two types of addresses used in the network layer communication:

- Network Service Access Point (NSAP) - NSAP addresses identify network layer services, one for each service running.
- Network Entity Title (NET) - NET addresses identify network layer entities or processes instead of services.

Protocol Structure

The General Format of NSAP:

<table>
<tr><td>1byte</td><td>2bytes</td><td colspan="2">2-4bytes</td><td>0-13bytes</td><td>1-8bytes</td><td>1byte</td></tr>
<tr><td colspan="2">IDP</td><td colspan="5">DSP</td></tr>
<tr><td>AFI</td><td>IDI</td><td colspan="2">CDP</td><td colspan="3">CDSP</td></tr>
<tr><td>AFI</td><td>IDI</td><td>CFI</td><td>CDI</td><td>RDAA</td><td>ID</td><td>SEL</td></tr>
</table>

- IDP - Initial Domain Part
- AFI - Authority and Format Identifier, a two-decimal-digit, 38 for decimal abstract syntax of the DSP or 39 for binary abstract syntax of the DSP
- IDI - Initial Domain Identifier, a three-decimal-digit country code, as defined in ISO 3166
- DSP - Domain Specific Part
- CDP - Country Domain Part, 2..4 octets
- CFI - Country Format Identifier, one digit
- CDI - Country Domain Identifier, 3 to 7 digits, fills CDP to an octet boundary
- CDSP - Country Domain Specific Part
- RDAA - Routing Domain and Area Address
- ID - System Identifier (1..8 octet)
- SEL - NSAP Selector

While RDIs and RDCIs need not be related to the set of addresses within the domains they depict, RDIs and RDCIs are assigned based on the NSAP prefixes assigned to domains. A subscriber RD should use the NSAP prefix assigned to it as its RDI. A multihomed RD should use one of the NSAP prefixes assigned to it as its RDI.

Related protocols

IS-IS, CLNP, IDRP, CONP, ES-IS

Sponsor Source

CONP is defined in ISO (www.iso.org) 8208 and 8878 and Itu (www.itu.org) X.213 and X.223.

Reference

http://www.doc.ua.pt/arch/itu/rec/product/X.htm
X.213: Information technology – Open Systems Interconnection – Network service definition
X.223: Use of X.25 to provide the OSI connection-mode Network service for ITU-T applications

Protocol Name

ES-IS: End System to Intermediate System Routing Exchange Protocol

Protocol Description

End System to Intermediate System Routing Exchange Protocol (ES-IS), developed by ISO, permits End Systems and Intermediate Systems to exchange configuration and routing information to facilitate the operation of the routing and relaying functions of the Network Layer in the ISO network environment. In an ISO network, there are End Systems, Intermediate Systems, Areas and Domains. End systems are user devices. Intermediate systems are routers. Routers are organized into local groups called 'areas', and several areas are grouped together into a 'domain'. ES-IS, working in conjunction with CLNP, IS-IS, and IDRP, provides complete routing over the entire network.

ES-IS provides solutions for the following practical problems:

1. For end systems to discover the existence and reachability of intermediate systems that can route NPDUs to destinations on subnetworks other than the one(s) to which the end system is directly connected.
2. For end systems to discover the existence and reachability of other end systems on the same subnetwork.
3. For intermediate systems to discover the existence and reachability of end systems on each of the subnetworks to which they are directly connected.
4. For end systems to decide which intermediate system to use to forward NPDUs to a particular destination when more than one intermediate system is accessible.

ES-IS provides two types of information to Network entities which support its operation: a) Configuration Information, which permits End Systems to discover the existence and reachability of Intermediate Systems and permits Intermediate Systems to discover the existence and reachability of End Systems; and b) Route Redirection Information which allows Intermediate Systems to inform End Systems of (potentially) better paths to use when forwarding NPDUs to a particular destination. A Network Entity may choose to support either the Configuration Information, the Route Redirection Information, neither, or both.

Protocol Structure

ES-IS Protocol Data Unit contains the following:

ES-IS Header	Network address	Subnetwork address	Option

ES-IS Header:

1byte	1byte	1byte	1byte	1byte				2 bytes	2 bytes
NLPID	Length	Version	Reserved	0	0	0	Type	H-Time	Checksum

- NLPID – Network Layer Protocol Identification. The value of this field shall be 1000 0010
- Length – Length Indicator is the length of the entire PDU
- Version – Protocol ID extension. This identifies a standard version of ISO xxxx, End System to Intermediate System Routing Exchange Protocol for use in conjunction with ISO 8473.
- Reserved – Must be zero.
- Type –The Type code field identifies the type of the protocol data unit.
- H-Time – Holding time field specifies for how long the receiving Network entity should retain the configuration/routing information contained in this PDU.
- Checksum – Error checking which is computed on the entire PDU header.

Related protocols

IS-IS, CLNP, IDRP, CONP

Sponsor Source

ES-IS is defined in ISO (www.iso.org).

Reference

http://www.javvin.com/protocol/rfc955.pdf
End System to Intermediate System Routing Exchange Protocol for use in conjunction with ISO 8473

Protocol Name

IDRP: Inter-Domain Routing Protocol

Protocol Description

The Inter-Domain Routing Protocol (IDRP), which provides routing for OSI defined network environments, is similar to BGP in the TCP/IP network. In an OSI network, there are End Systems, Intermediate Systems, Areas and Domains. End systems are user devices. Intermediate systems are routers. Routers are organized into local groups called 'areas', and several areas are grouped together into a 'domain'. Inter-Domain Routing Protocol (IDRP) is designed to provide routing among domains. IDRP, working in conjunction with CLNP, ES-IS, and IS-IS, provides complete routing over the entire network.

A router that participates in IDRP is called a Boundary Intermediate System (BIS) and may belong to only one domain. IDRP governs the exchange of routing information between a pair of neighbors, either external or internal. IDRP is self-contained with respect to the exchange of information between external neighbors. Exchange of information between internal neighbors relies on additional support provided by intra-domain routing (unless internal neighbors share a common subnetwork).

To facilitate routing information aggregation/abstraction, IDRP allows grouping of a set of connected domains into a Routing Domain Confederation (RDC). A given domain may belong to more than one RDC. The ability to group domains in RDCs provides a simple, yet powerful mechanism for routing information aggregation and abstraction. It allows reduction of topological information by replacing a sequence of RDIs carried by the RD_PATH attribute with a single RDCI. It also allows reduction of the amount of information related to transit policies, and simplifies the route selection policies.

Each domain participating in IDRP is assigned a unique Routing Domain Identifier (RDI), which is basically an OSI network layer address. Each RDC is assigned a unique Routing Domain Confederation Identifier (RDCI). RDCIs are assigned out of the address space allocated for RDIs. RDCIs and RDIs are syntactically indistinguishable. It is expected that RDI and RDCI assignment and management would be part of the network layer assignment and management procedures.

Protocol Structure

The General Format of NSAP:

<table>
<tr><td>1byte</td><td>2bytes</td><td colspan="2">2-4bytes</td><td>0-13bytes</td><td>1-8bytes</td><td>1byte</td></tr>
<tr><td colspan="2">IDP</td><td colspan="5">DSP</td></tr>
<tr><td>AFI</td><td>IDI</td><td colspan="2">CDP</td><td colspan="3">CDSP</td></tr>
<tr><td>AFI</td><td>IDI</td><td>CFI</td><td>CDI</td><td>RDAA</td><td>ID</td><td>SEL</td></tr>
</table>

- IDP - Initial Domain Part
- AFI - Authority and Format Identifier, a two-decimal-digit, 38 for decimal abstract syntax of the DSP or 39 for binary abstract syntax of the DSP
- IDI - Initial Domain Identifier, a three-decimal-digit country code, as defined in ISO 3166
- DSP - Domain Specific Part
- CDP - Country Domain Part, 2..4 octets
- CFI - Country Format Identifier, one digit
- CDI - Country Domain Identifier, 3 to 7 digits, fills CDP to an octet boundary
- CDSP - Country Domain Specific Part
- RDAA - Routing Domain and Area Address
- ID - System Identifier (1..8 octet)
- SEL - NSAP Selector

While RDIs and RDCIs need not be related to the set of addresses within the domains they depict, RDIs and RDCIs are assigned based on the NSAP prefixes assigned to domains. A subscriber RD should use the NSAP prefix assigned to it as its RDI. A multihomed RD should use one of the NSAP prefixes assigned to it as its RDI.

Related protocols

ES-IS, CLNP, IDRP, IS-IS, BGP, CONP

Sponsor Source

IDRP is defined in ISO (www.iso.org) 10747 and discussed in IETF (www.ietf.org).

Reference

http://www.javvin.com/protocol/rfc1629.pdf
Guidelines for OSI NSAP Allocation in the Internet
http://www.acm.org/sigcomm/standards/iso_stds/IDRP/10747.TXT
Protocol for the Exchange of Inter-Domain Routing Information among Intermediate Systems to Support Forwarding of ISO 8473 PDUs

Protocol Name

IS-IS: Intermediate System to Intermediate System Routing Protocol

Protocol Description

Intermediate System-to-Intermediate System (IS-IS) is a routing protocol developed by the ISO. It is a link-state protocol where ISs (routers) exchange routing information based on a single metric to determine network topology. It behaves similar to Open Shortest Path First (OSPF) in the TCP/IP network.

In an IS-IS network, there are End Systems, Intermediate Systems, Areas and Domains. End systems are user devices. Intermediate systems are routers. Routers are organized into local groups called 'areas', and several areas are grouped together into a 'domain'. IS-IS is designed primarily for providing intradomain routing or routing within an area. IS-IS, working in conjunction with CLNP, ES-IS and IDRP, provides complete routing over the entire network.

IS-IS routing makes use of two-level hierarchical routing. Level 1 routers know the topology in their area, including all routers and hosts, but they do not know the identity of routers or destinations outside of their area. Level 1 routers forward all traffic for destinations outside of their area to a level 2 router within their area which knows the level 2 topology. Level 2 routers do not need to know the topology within any level 1 area, except to the extent that a level 2 router may also be a level 1 router within a single area.

IS-IS adapted to carry IP network information is called Integrated IS-IS. Integrated IS-IS has the most important characteristic necessary in a modern routing protocol: It supports VLSM and converges rapidly. It is also scalable to support very large networks.

There are two types of IS-IS addresses:

Network Service Access Point (NSAP) - NSAP addresses identify network layer services, one for each service running.

Network Entity Title (NET) - NET addresses identify network layer entities or processes instead of services.

Devices may have more than one of each of the two types of addresses. However NET's should be unique, and the System ID portion of the NSAP must be unique for each system.

Protocol Structure

IS-IS PDU Header:

<table>
<tr><th colspan="4">8</th><th>16 bit</th></tr>
<tr><td colspan="4">Intradomain routing protocol discriminator</td><td>Length indicator</td></tr>
<tr><td colspan="4">Version/protocol ID extension</td><td>ID length</td></tr>
<tr><td>R</td><td>R</td><td>R</td><td>PDU type</td><td>Version</td></tr>
<tr><td colspan="4">Reserved</td><td>Maximum area addresses</td></tr>
</table>

- Intradomain routing protocol discriminator - Network layer protocol identifier assigned to this protocol
- Length indicator - Length of the fixed header in octets.
- Version/protocol ID extension - Equal to 1.
- ID length - Length of the ID field of NSAP addresses and NETs used in this routing domain.
- R - Reserved bits.
- PDU type - Type of PDU. Bits 6, 7 and 8 are reserved.
- Version - Equal to 1.
- Maximum area addresses - Number of area addresses permitted for this intermediate system's area.

Format of NSAP for IS-IS:

<table>
<tr><td colspan="2">< IDP ></td><td colspan="3">< DSP ></td></tr>
<tr><td colspan="2"></td><td>< HO-DSP ></td><td colspan="2"></td></tr>
<tr><td>AFI</td><td>IDI</td><td colspan="3">Contents assigned by authority identified in IDI field</td></tr>
<tr><td colspan="3">< Area Address ></td><td>< ID ></td><td>< SEL ></td></tr>
</table>

- IDP - Initial Domain Part
- AFI - Authority and Format Identifier (1-byte); Provides information about the structure and content of the IDI and DSP fields.
- IDI - Initial Domain Identifier (variable length)
- DSP - Domain Specific Part
- HO-DSP - High Order Domain Specific Part
- Area Address (variable)
- ID - System ID (1- 8 bytes)
- SEL - n-selector (1-byte value that serves a function similar to the port number in Internet Protocol).

Related protocols

OSPF, ES-IS, CLNP, IDRP, CONP

Sponsor Source

IS-IS is defined in ISO (www.iso.org) 10589 and reviewed by IETF (www.ietf.org) RFC 1629.

Reference

http://www.javvin.com/protocol/rfc1629.pdf
Guidelines for OSI NSAP Allocation in the Internet

Cisco Protocols

Cisco Systems plays an active role in the IETF committees to bring Cisco technology initiatives into the standards track. At the same time, Cisco created many proprietary protocols, mostly in the data link layer (layer 2) and network layer (layer 3). Some of the Cisco protocols are listed as follows:

CDP: Cisco Discovery Protocol

CGMP: Cisco Group Management Protocol

DTP: Cisco Dynamic Trunking Protocol

EIGRP: Enhanced Interior Gateway Routing Protocol

HSRP: Hot Standby Router Protocol

IGRP: Interior Gateway Routing Protocol

ISL: Cisco Inter-Switch Link Protocol

DISL: Dynamic Inter-Switch Link Protocol

RGMP: Cisco Router Port Group Management Protocol

TACACS: Terminal Access Controller Access Control Protocol

VTP: Cisco VLAN Trunking Protocol

XOT: Cisco X.25 Over TCP Protocol

Protocol Name

CDP: Cisco Discovery Protocol

Protocol Description

Cisco Discovery Protocol (CDP) is primarily used to obtain protocol addresses of neighboring devices and discover the platform of those devices. CDP can also be used to show information about the interfaces your router uses. CDP is media- and protocol-independent, and runs on all Cisco-manufactured equipment including routers, bridges, access servers, and switches.

Use of SNMP with the CDP Management Information Base (MIB) allows network management applications to learn the device type and the SNMP agent address of neighboring devices, and to send SNMP queries to those devices. Cisco Discovery Protocol uses the CISCO-CDP-MIB.

CDP runs on all media that support Subnetwork Access Protocol (SNAP), including local-area network (LAN), Frame Relay, and Asynchronous Transfer Mode (ATM) physical media. CDP runs over the data link layer only. Therefore, two systems that support different network-layer protocols can learn about each other.

Each device configured for CDP sends periodic messages, known as advertisements, to a multicast address. Each device advertises at least one address at which it can receive SNMP messages. The advertisements also contain time-to-live, or holdtime, information, which indicates the length of time a receiving device should hold CDP information before discarding it. Each device also listens to the periodic CDP messages sent by others in order to learn about neighboring devices and determine when their interfaces to the media go up or down.

CDP Version-2 (CDPv2), the most recent release of the protocol, provides more intelligent device tracking features. These features include a reporting mechanism which allows for more rapid error tracking, thereby reducing costly downtime. Reported error messages can be sent to the console or to a logging server, and cover instances of unmatching native VLAN IDs (IEEE 802.1Q) on connecting ports, and unmatching port duplex states between connecting devices.

Protocol Structure

CDPv2 show commands can provide detailed output on VLAN Trunking Protocol (VTP) management domain and duplex modes of neighbor devices, CDP-related counters, and VLAN IDs of connecting ports. The following table lists the CDP commands:

Command	Purpose
clear cdp counters	Resets the traffic counters to zero.
clear cdp table	Deletes the CDP table of information about neighbors.
show cdp	Displays the interval between transmissions of CDP advertisements, the number of seconds the CDP advertisement is valid for a given port, and the version of the advertisement.
show cdp entry entry-name [protocol | version]	Displays information about a specific neighbor. Display can be limited to protocol or version information.
show cdp interface [type number]	Displays information about interfaces on which CDP is enabled.
show cdp neighbors [type number] [detail]	Displays the type of device that has been discovered, the name of the device, the number and type of the local interface (port), the number of seconds the CDP advertisement is valid for the port, the device type, the device product number, and the port ID. Issuing the detail keyword displays information on the native VLAN ID, the duplex mode, and the VTP domain name associated with neighbor devices.
show cdp traffic	Displays CDP counters, including the number of packets sent and received and checksum errors.
show debugging	Displays information about the types of debugging that are enabled for your router. See the Cisco IOS Debug Command Reference for more information about CDP debug commands.

Related protocols

SNMP, SNAP

Sponsor Source

CDP is a Cisco protocol.

Reference

http://www.cisco.com/en/US/products/sw/iosswrel/ps1831/products_configuration_guide_chapter09186a00800ca66d.html
Configuring the Cisco Discovery Protocol

Protocol Name

CGMP: Cisco Group Management Protocol

Protocol Description

Cisco Group Management Protocol (CGMP) limits the forwarding of IP multicast packets to only those ports associated with IP multicast clients. These clients automatically join and leave groups that receive IP multicast traffic, and the switch dynamically changes its forwarding behavior according to these requests. CGMP provides the following services:

- Allows IP multicast packets to be switched only to those ports that have IP multicast clients.
- Saves network bandwidth on user segments by not propagating unnecessary IP multicast traffic.
- Does not require changes to the end host systems.
- Does not incur the overhead of creating a separate VLAN for each multicast group in the switched network.

When CGMP is enabled, it automatically identifies the ports to which the CGMP-capable router is attached. CGMP is enabled by default and supports a maximum of 64 IP multicast group registrations. Multicast routers that support CGMP periodically send CGMP join messages to advertise themselves to switches within a network. A receiving switch saves the information and sets a timer equal to the router hold time. The timer is updated every time the switch receives a CGMP join message advertising itself. When the last router hold time expires, the switch removes all IP multicast groups learned from CGMP.

CGMP works in conjunction with IGMP messages to dynamically configure Cisco Catalyst switch ports so that IP multicast traffic is forwarded only to those ports associated with IP multicast hosts. A CGMP-capable IP multicast router sees all IGMP packets and therefore can inform the Catalyst switches when specific hosts join or leave IP multicast groups. When the CGMP-capable router receives an IGMP control packet, it creates a CGMP packet that contains the request type (either join or leave), the multicast group address, and the actual MAC address of the host. The router then sends the CGMP packet to a well-known address to which all Catalyst switches listen. When a switch receives the CGMP packet, the switch interprets the packet and modifies the forwarding behavior of the multicast group. From then on, this multicast traffic is sent only to ports associated with the appropriate IP multicast clients. This process is done automatically, without user intervention.

Protocol Structure

CGMP message format:

1 byte	6 bytes	1 byte	6 bytes	1 byte
Count	Group Destination Address	Type	Unicast Source Address	Version

- Count: Unsigned 8 bit integer
- Group Destination Address: The hardware MAC address of the destination device.
- Type: Message Type
- Unicast Source Address: The hardware MAC address of the unicast source device
- Version: CGMP version number

Related protocols

IPv4, IGMP, PIM-SM, RGMP

Sponsor Source

CGMP is a Cisco protocol.

Reference

http://www.cisco.com/univercd/cc/td/doc/product/lan/28201900/1928v67x/eescg67x/03cgmpl.pdf
Cisco Group Management Protocol

Protocol Name

DTP: Cisco Dynamic Trunking Protocol

Protocol Description

Dynamic Trunking Protocol (DTP), a Cisco proprietary protocol in the VLAN group, is for negotiating trunking on a link between two devices and for negotiating the type of trunking encapsulation (802.1Q) to be used.

There are different types of trunking protocols. If a port can become a trunk, it may also have the ability to trunk automatically and, in some cases, even negotiate what type of trunking to use on the port. This ability to negotiate the trunking method with the other device is called dynamic trunking.

The first issue is that both ends of a trunk cable had better agree they're trunking, or they're going to be interpreting trunk frames as normal frames. End stations will be confused by the extra tag information in the frame header, their driver stacks won't understand it, and the end systems may lock up or fail in odd ways. To resolve this problem, Cisco created a protocol for switches to communicate intentions. The first version of it was VTP, VLAN Trunking Protocol, which worked with ISL. The newer version works with 802.1q as well, and is called Dynamic Trunking Protocol (DTP).

The second issue is creating VLAN's. To configure VLAN's individually switch by switch, it is lot of work and easy to cause inconsistency, in that VLAN 100 could be Engineering on one switch, and Accounting on another. That would be a source of confusion in troubleshooting, and might also defeat your carefully crafted VLAN security scheme. This issue is also addressed by VTP/DTP. You can create or delete a VLAN on one switch, and have the information propagate automatically to a group of switches under the same administrative control. This group of switches would be a VTP domain.

Protocol Structure

On a Catalyst set-based switch, the syntax for setting up a link as a trunk is:

set trunk mod_num/port_num [on | desirable | auto | nonegotiate] [isl | dot1q | negotiate] [vlan_range]

Use this command to set the specified port or ports to trunking. The first set of keyword arguments govern the DTP modes:

Mode	What the Mode Does
on	Forces the link into permanent trunking, even if the neighbor doesn't agree
off	Forces the link to permanently not trunk, even if the neighbor doesn't agree
desirable	Causes the port to actively attempt to become a trunk, subject to neighbor agreement (neighbor set to on, desirable, or auto)
auto	Causes the port to passively be willing to convert to trunking. The port will not trunk unless the neighbor is set to on or desirable. This is the default mode. Note that auto-auto (both ends default) links will not become trunks.
nonegotiate	Forces the port to permanently trunk but not send DTP frames. For use when the DTP frames confuse the neighboring (non-Cisco) 802.1q switch. You must manually set the neighboring switch to trunking.

The second set of keywords governs the type of VLAN tagging to use: ISL, 802.1q, or negotiate which to use.

Related protocols

IEEE 802.1Q, VTP, ISL, DISL

Sponsor Source

DTP is a Cisco protocol.

Reference

http://www.cisco.com/en/US/products/hw/switches/ps708/products_configuration_guide_chapter09186a008019f048.html#1017196
Understanding and Configuring VLAN Trunking Protocol
http://www.cisco.com/univercd/cc/td/doc/product/lan/cat6000/sw_5_2/cofigide/e_trunk.htm
configuring VLAN Trunks

Protocol Name

EIGRP: Enhanced Interior Gateway Routing Protocol

Protocol Description

Enhanced Interior Gateway Routing Protocol (EIGRP) is an enhanced version of IGRP. IGRP is Cisco's Interior Gateway Routing Protocol used in TCP/IP and OSI internets. It is regarded as an interior gateway protocol (IGP) but has also been used extensively as an exterior gateway protocol for inter-domain routing.

Key capabilities that distinguish Enhanced IGRP from other routing protocols include fast convergence, support for variable-length subnet mask, support for partial updates, and support for multiple network layer protocols.

A router running Enhanced IGRP stores all its neighbors' routing tables so that it can quickly adapt to alternate routes. If no appropriate route exists, Enhanced IGRP queries its neighbors to discover an alternate route. These queries propagate until an alternate route is found.

Its support for variable-length subnet masks permits routes to be automatically summarized on a network number boundary. In addition, Enhanced IGRP can be configured to summarize on any bit boundary at any interface.

Enhanced IGRP does not make periodic updates. Instead, it sends partial updates only when the metric for a route changes. Propagation of partial updates is automatically bounded so that only those routers that need the information are updated. As a result of these two capabilities, Enhanced IGRP consumes significantly less bandwidth than IGRP.

Protocol Structure

8	16	32bit
Version	Opcode	Checksum
Flags		
Sequence number		
Acknowledge number		
Asystem: Autonomous system number		
Type		Length

- Version -- The version of the protocol.
- Opcode –Operation code indicating the message type: 1 Update. 2 Reserved. 3 Query. 4 Hello. 5 IPX-SAP.
- Checksum – IP checksum which is computed using the same checksum algorithm as a UDP checksum
- Flag -- Initialization bit and is used in establishing a new neighbor relationship
- Sequence number -- Used to send messages reliably
- Acknowledge number -- Used to send messages reliably
- Asystem -- Autonomous system number. A gateway can participate in more than one autonomous system where each system runs its own IGRP. For each autonomous system, there are completely separate routing tables. This field allows the gateway to select which set of routing tables to use.
- Type – Value in the type field: 1 EIGRP Parameters. 2 Reserved. 3 Sequence. 4 Software version. 5 Next Multicast sequence.
- Length -- Length of the frame.

Related protocols

IP, TCP, IGRP, EGP, BGP, GRE, RIP

Sponsor Source

EIGRP is a Cisco protocol.

Reference

http://www.cisco.com/univercd/cc/td/doc/cisintwk/ito_doc/en_igrp.htm
Enhanced IGRP

Protocol Name

HSRP: Hot Standby Router Protocol

Protocol Description

Hot Standby Router Protocol (HSRP) is designed to support non-disruptive failover of IP traffic in certain circumstances and to allow hosts to appear to use a single router and to maintain connectivity even if the actual first hop router they are using fails. In other words, the protocol protects against the failure of the first hop router when the source host cannot learn the IP address of the first hop router dynamically. Multiple routers participate in this protocol and in concert create the illusion of a single virtual router. The protocol insures that one and only one of the routers is forwarding packets on behalf of the virtual router. End hosts forward their packets to the virtual router.

The router forwarding packets is known as the active router. A standby router is selected to replace the active router should it fail. The protocol provides a mechanism for determining active and standby routers, using the IP addresses on the participating routers. If an active router fails a standby router can take over without a major interruption in the host's connectivity.

HSRP runs on top of UDP, and uses port number 1985. Routers use their actual IP address as the source address for protocol packets, not the virtual IP address. This is necessary so that the HSRP routers can identify each other.

Protocol Structure

8	16	24	32bit
Version	Op code	State	Hellotime
Holdtime	Priority	Group	Reserved
Authentication data			
Authentication data			
Virtual IP address			

- Version -- HSRP version number. The current version is 0.
- Op code -- Type of message contained in the packet. Possible values are:
  - 0 Hello, sent to indicate that a router is running and is capable of becoming the active or standby router.
  - 1 Coup, sent when a router wishes to become the active router.
  - 2 Resign, sent when a router no longer wishes to be the active router.
- State -- Internally, each router in the standby group implements a state machine. The State field describes the current state of the router sending the message. Possible values are: 0 Initial; 1 Learn; 2 Listen; 4 Speak; 8 Standby; 16 Active.
- Hellotime -- Approximate period between the Hello messages that the router sends (for Hello messages only). If the Hellotime is not configured on a router, then it may be learned from the Hello message from the active router.
- Holdtime -- The amount of time, in seconds, that the current Hello message should be considered valid. (For Hello messages only.)
- Priority -- Used to elect the active and standby routers. When comparing priorities of two different routers, the router with the numerically higher priority wins. In the case of routers with equal priority the router with the higher IP address wins.
- Group -- Identifies the standby group. For Token Ring, values between 0 and 2 inclusive are valid. For other media, values between 0 and 255 inclusive are valid.
- Authentication data -- Clear-text 8 character reused password. If no authentication data is configured, the recommended default value is 0x63 0x69 0x73 0x63 0x6F 0x00 0x00 0x00.
- Virtual IP address -- Virtual IP address used by this group. If the virtual IP address is not configured on a router, then it may be learned from the Hello message from the active router. An address should only be learned if no address was configured and the Hello message is authenticated.

Related protocols

IIP, UDP

Sponsor Source

HSRP is a Cisco protocol and circulated by IETF RFC 2281.

Reference

http://www.javvin.com/protocol/rfc2281.pdf
Hot Standby Router Protocol

Protocol Name

IGRP: Interior Gateway Routing Protocol

Protocol Description

The Interior Gateway Routing Protocol (IGRP) is a routing protocol to provide routing within an autonomous system (AS). In the mid-1980s, the most popular interior routing protocol was the Routing Information Protocol (RIP). Although RIP was quite useful for routing within small- to moderate-sized, relatively homogeneous internetworks, its limits were being pushed by network growth. The popularity of Cisco routers and the robustness of IGRP encouraged many organizations with large internetworks to replace RIP with IGRP. Cisco developed Enhanced IGRP in the early 1990s to improve the operating efficiency of IGRP.

IGRP is a distance vector Interior Gateway Protocol (IGP). Distance vector routing protocols mathematically compare routes using some measurement of distance. This measurement is known as the distance vector. Distance vector routing protocols are often contrasted with link-state routing protocols, which send local connection information to all nodes in the internetwork.

To provide additional flexibility, IGRP permits multipath routing. Dual equal-bandwidth lines can run a single stream of traffic in round-robin fashion, with automatic switchover to the second line if one line goes down. Multiple paths can have unequal metrics yet still be valid multipath routes. For example, if one path is three times better than another path (its metric is three times lower), the better path will be used three times as often. Only routes with metrics that are within a certain range or variance of the best route are used as multiple paths. Variance is another value that can be established by the network administrator.

Protocol Structure

8	16	24	32bit
Version	Op code	Edition	ASystem
Ninterior	Nsystem	Nexterior	Checksum

- Version -- IGRP version number (currently 1).
- Opcode -- Operation code indicating the message type: 1 Update; 2 Request.
- Edition -- Serial number which is incremented whenever there is a change in the routing table.
- Asystem -- Autonomous system number. A gateway can participate in more than one autonomous system where each system runs its own IGRP. For each autonomous system, there are completely separate routing tables. This field allows the gateway to select which set of routing tables to use.
- Ninterior, Nsystem, Nexterior -- Indicate the number of entries in each of these three sections of update messages. The first entries (Ninterior) are taken to be interior, the next entries (Nsystem) as being system, and the final entries (Nexterior) as exterior.
- Checksum -- IP checksum which is computed using the same checksum algorithm as a UDP checksum.

Related protocols

EIGRP, EGP, BGP, GRE, IP, TCP, RIP

Sponsor Source

IGRP is a Cisco protocol.

Reference

http://www.cisco.com/univercd/cc/td/doc/cisintwk/ito_doc/igrp.htm
Interior Gateway routing Protocol

Protocol Name

ISL & DISL: Cisco Inter-Switch Link Protocol and Dynamic ISL Protocol

Protocol Description

Inter-Switch Link. Protocol (ISL), a Cisco-proprietary protocol, maintains VLAN information as traffic flows between switches and routers.

Inter-Switch Link (ISL) tagging accomplishes the same task as 802.1Q trunking but uses a different frame format. ISL trunks are Cisco proprietary and define only a point-to-point connection between two devices, typically switches. The name Inter-Switch Link hints at this design. ISL frame tagging uses a low-latency mechanism for multiplexing traffic from multiple VLANs on a single physical path. ISL has been implemented for connections among switches, routers, and network interface cards (NICs) used on nodes such as servers. To support the ISL feature, each connecting device must be ISL-configured. A router that is ISL-configured can allow inter-VLAN communications. A non-ISL device that receives ISL-encapsulated Ethernet frames will most likely consider them protocol errors because of the format and size of the frames.

Like 802.1Q, ISL functions at Layer 2 of the OSI model, but it differs by encapsulating the entire Layer 2 Ethernet frame inside an ISL header and trailer. Because ISL encapsulates the entire frame, it is protocol-independent and can carry any type of Layer 2 frame or upper-layer protocol between the switches. The encapsulated frames may be token-ring or Fast Ethernet, and are carried unchanged from transmitter to receiver. ISL has the following characteristics:

- Performed with application-specific integrated circuits (ASIC)
- Not intrusive to client stations; client does not see the ISL header
- Effective between switches, routers and switches, and switches and servers with ISL NICs

Dynamic Inter-Switch Link Protocol (DISL), also a Cisco protocol, simplifies the creation of an ISL trunk from two interconnected Fast Ethernet devices. Fast EtherChannel technology enables aggregation of two full-duplex Fast Ethernet links for high-capacity backbone connections. DISL minimizes VLAN trunk configuration procedures because only one end of a link needs to be configured as a trunk.

Protocol Structure

ISL header structure:

40	4	4	48	16	8	24	15	1	16	16bits
DA	Type	User	SA	Len	AAA03	HSA	VLAN	BPDU	Index	Resv

- DA - 40-bit multicast destination address.
- Type - 4-bit descriptor of the encapsulated frame types—Ethernet (0000), Token Ring (0001), FDDI (0010), and ATM (0011).
- User - 4-bit descriptor used as the type field extension or to define Ethernet priorities. This is a binary value from 0, the lowest priority, to 3, the highest priority.
- SA - 48-bit source MAC address of the transmitting Catalyst switch.
- LEN - 16-bit frame-length descriptor minus DA type, user, SA, LEN, and CRC.
- AAAA03 - Standard SNAP 802.2 LLC header.
- HSA - First 3 bytes of SA (manufacturer's ID or organizational unique ID).
- VLAN - 15-bit VLAN ID. Only the lower 10 bits are used for 1024 VLANs.
- BPDU - 1-bit descriptor identifying whether the frame is a Spanning Tree bridge protocol data unit (BPDU). Also set if the encapsulated frame is a Cisco Discovery Protocol (CDP) frame.
- INDEX - 16-bit descriptor that identifies the transmitting port ID. Used for diagnostics.
- RES - 16-bit reserved field used for additional information, such as Token Ring and Fiber Distributed Data Interface (FDDI) frame Frame Check (FC) field.

Related protocols

IEEE 802.1Q,VTP, DTP, Ethernet, Token Ring

Sponsor Source

ISP and DISL are Cisco protocols.

Reference

http://www.cisco.com/en/US/products/hw/switches/ps708/products_configuration_guide_chapter09186a008019f048.html#1017196
Understanding and Configuring VLAN Trunking Protocol
http://www.cisco.com/univercd/cc/td/doc/product/lan/cat6000/sw_5_2/cofigide/e_trunk.htm
Configuring VLAN Trunks

Protocol Name

RGMP: Cisco Router Port Group Management Protocol

Protocol Description

The Cisco Router Port Group Management Protocol (RGMP) is defined to address the limitations of Internet Group Management Protocol (IGMP) in its Snooping mechanism. RGMP is used between multicast routers and switches to restrict multicast packet forwarding in switches to those routers where the packets may be needed. RGMP is designed for backbone switched networks where multiple, high speed routers are interconnected.

The main limitation of IGMP Snooping is that it can only restrict multicast traffic onto switch ports where receiving hosts are connected directly or indirectly via other switches. IGMP Snooping can not restrict multicast traffic to ports where at least one multicast router is connected. It must instead flood multicast traffic to these ports. Snooping on IGMP messages alone is an intrinsic limitation. Through it, a switch can only learn which multicast flows are being requested by hosts. A switch cannot learn through IGMP which traffic flows need to be received by router ports to be routed because routers do not report these flows via IGMP.

The RGMP protocol restricts multicast traffic to router ports. To effectively restrict traffic, it must be supported by both the switches and the routers in the network. Backbone switches use RGMP to learn which groups are desired at each of their ports. Multicast routers use RGMP to pass such information to the switches. Only routers send RGMP messages. They ignore received RGMP messages. When a router no longer needs to receive traffic for a particular group, it sends an RGMP Leave message for the group. A switch enabled for RGMP on a network consumes RGMP messages received from ports of the network and processes them. If enabled for RGMP, the switch must NOT forward/flood received RGMP messages out to other ports of the network.

RGMP is designed to work in conjunction with multicast routing protocols where explicit join/prune to the distribution tree is performed. PIM-SM is an example of such a protocol. The RGMP protocol specifies operations only for IP version 4 multicast routing. IP version 6 is not considered.

Protocol Structure

RGMP message format is the same as the IGMPv2:

8	16	32bit
Type	Reserved	Checksum
Group Address		

- Type - There are four types of RGMP messages of concern to the router-switch interaction. The type codes are defined to be the highest values in an octet to avoid the re-use of already assigned IGMP type codes: 0xFF = Hello; 0xFE = Bye; 0xFD = Join a group; 0xFC = Leave a group.
- Reserved - The reserved field in the message MUST be transmitted as zeros and ignored on receipt.
- Checksum - Checksum covers the RGMP message (the entire IPv4 payload). The algorithm and handling of checksum are the same as those for IGMP messages.
- Group Address - In an RGMP Hello or Bye message, the group address field is set to zero. In an RGMP Join or Leave message, the group address field holds the IPv4 multicast group address of the group being joined or left.

Related protocols

IPv4, IGMP, PIM-SM, CGMP

Sponsor Source

RGMP is a Cisco protocol.

Reference

http://www.javvin.com/protocol/rfc3488.pdf
Cisco Systems Router Port Group Management Protocol

Protocol Name

TACACS (and TACACS+): Terminal Access Controller Access Control System

Protocol Description

The Terminal Access Controller Access Control System (TACACS) provides access control for routers, network access servers and other networked computing devices via one or more centralized servers. TACACS+ provides separate authentication, authorization and accounting services.

TACACS allows a client to accept a username and password and send a query to a TACACS authentication server, sometimes called a TACACS daemon or simply TACACSD. This server is normally a program running on a host. The host determines whether to accept or deny the request and sends a response back. The TIP then allows access or not, based upon the response. In this way, the process of making the decision is “opened up” and the algorithms and data used to make the decision are under the complete control of whoever is running the TACACS daemon. The extensions to the protocol provide for more types of authentication requests and more types of response codes than were in the original specification.

There are three versions of TACACS and the third version called TACACS+, is not compatible with previous versions.

Protocol Structure

4	8	16	24	32bit
Major	Minor	Packet type	Sequence no.	Flags
Session ID				
Length				

- Major version - The major TACACS+ version number.
- Minor version - The minor TACACS+ version number. This is intended to allow revisions to the TACACS+ protocol while maintaining backwards compatibility.
- Packet type - Possible values are:
  TAC_PLUS_AUTHEN:= 0x01 (Authentication).
  TAC_PLUS_AUTHOR:= 0x02 (Authorization).
  TAC_PLUS_ACCT:= 0x03 (Accounting).
- Sequence number - The sequence number of the current packet for the current session. The first TACACS+ packet in a session must have the sequence number 1 and each subsequent packet will increment the sequence number by one. Thus clients only send packets containing odd sequence numbers, and TACACS+ daemons only send packets containing even sequence numbers.
- Flags - This field contains various flags in the form of bitmaps. The flag values signify whether the packet is encrypted.
- Session ID - The ID for this TACACS+ session.
- Length - The total length of the TACACS+ packet body (not including the header).

Related protocols

TCP, TELNET, SMTP, FTP, RADIUS

Sponsor Source

TACACS (and TACACS+) is a Cisco protocol.

Reference

http://www.javvin.com/protocol/rfc1492.pdf
An Access Control Protocol, Sometimes Called TACACS
http://www.javvin.com/protocol/tacacs.html
Introduction to TACACS+

Protocol Name

VTP: Cisco VLAN Trunking Protocol

Protocol Description

VLAN Trunking Protocol (VTP) is a Cisco Layer 2 messaging protocol that manages the addition, deletion, and renaming of VLANs on a network-wide basis. Virtual Local Area Network (VLAN) Trunk Protocol (VTP) reduces administration in a switched network. When you configure a new VLAN on one VTP server, the VLAN is distributed through all switches in the domain. This reduces the need to configure the same VLAN everywhere. VTP is a Cisco-proprietary protocol that is available on most of the Cisco Catalyst Family products.

VTP ensures that all switches in the VTP domain are aware of all VLANs. There are occasions, however, when VTP can create unnecessary traffic. All unknown unicasts and broadcasts in a VLAN are flooded over the entire VLAN. All switches in the network receive all broadcasts, even in situations where few users are connected in that VLAN. VTP pruning is a feature used to eliminate (or prune) this unnecessary traffic.

By default, all Cisco Catalyst switches are configured to be VTP servers. This is suitable for small-scale networks where the size of the VLAN information is small and easily stored in all switches (in NVRAM). In a large network, a judgment call must be made at some point when the NVRAM storage needed is wasted, because it is duplicated on every switch. At this point, the network administrator should choose a few well-equipped switches and keep them as VTP servers. Everything else participating in VTP can be turned into a client. The number of VTP servers should be chosen so as to provide the degree of redundancy desired in the network.

There are three version of VTP so far. VTP Version 2 (V2) is not much different from VTP Version 1 (V1). The major difference is that VTP V2 introduces support for Token Ring VLANs. If you are using Token Ring VLANs, you need to enable VTP V2. Otherwise, there is no reason to use VTP V2.

VTP version 3 differs from earlier VTP versions in that it does not directly handle VLANs. VTP version 3 is a protocol that is only responsible for distributing a list of opaque databases over an administrative domain. When enabled, VTP version 3 provides the following enhancements to previous VTP versions:

- Support for extended VLANs.
- Support for the creation and advertising of private VLANs.
- Improved server authentication.
- Protection from the “wrong” database accidentally being inserted into a VTP domain.
- Interaction with VTP version 1 and VTP version 2.
- Provides the ability to be configured on a per-port basis.
- Provides the ability to propagate the VLAN database and other databases.

Protocol Structure

The format of the VTP header can vary depending on the type of VTP message. However, they all contain the following fields in the header:

- VTP protocol version: 1 or 2 or 3
- VTP message types:
  - Summary advertisements
  - Subset advertisement
  - Advertisement requests
  - VTP join messages
- Management domain length
- Management domain name

Summary Advertisements

When the switch receives a summary advertisement packet, it compares the VTP domain name to its own VTP domain name. If the name is different, the switch simply ignores the packet. If the name is the same, the switch then compares the configuration revision to its own revision. If its own configuration revision is higher or equal, the packet is ignored. If it is lower, an advertisement request is sent.

Summary Advert Packet Format:

8	16	24	32bit
Version	Code	Followers	MgmtD Len
Management Domain Nance (zero-padded to 32 bytes)			
Configuration Revision Number			
Updater Identity			
Update Timestamp (12 bytes)			
MDS Digest (16 bytes)			

- Followers indicate that this packet is followed by a Subset Advertisement packet.
- The updater identity is the IP address of the switch that is the last to have incremented the configuration revision.
- Update timestamps are the date and time of the last increment of the configuration revision.
- Message Digest 5 (MD5) carries the VTP password if it is configured and used to authenticate the validation of a VTP update.

Subset Advertisements

When you add, delete, or change a VLAN in a switch, the server switch where the changes were made increments the configura-

tion revision and issues a summary advertisement, followed by one or several subset advertisements. A subset advertisement contains a list of VLAN information. If there are several VLANS, more than one subset advertisement may be required in order to advertise them all.

Subset Advert Packet Format:

8	16	24	32bit
Version	Code	Sequence Number	MgmtD Len
Management Domain Nance (zero-padded to 32 bytes)			
Configuration Revision			
VLAN - info field 1			
.....			
VLAN - info field N			

The following formatted example shows that each VLAN information field contains information for a different VLAN (ordered with lower valued ISL VLAN IDs occurring first):

V - info - len	Status	VLAN - Type	VLAN - name Len
ISL VLAN - id		MTU Size	
802.10 index			
VLAN - name (padded with zeros to multiple of 4 bytes)			

Most of the fields in this packet are easy to understand. Below are two clarifications:

- Code—The format for this is 0x02 for subset advertisement.
- Sequence number—This is the sequence of the packet in the stream of packets following a summary advertisement. The sequence starts with 1.

Advertisement Requests

A switch needs a VTP advertisement request in the following situations:

- The switch has been reset.
- The VTP domain name has been changed.
- The switch has received a VTP summary advertisement with a higher configuration revision than its own.

Upon receipt of an advertisement request, a VTP device sends a summary advertisement, followed by one or more subset advertisements. Below is an example.

8	16	24	32bit
Version	Code	Rsvd	MgmtD Len
Management Domain Nance (zero-padded to 32 bytes)			
Start - Value			

- Code—The format for this is 0x03 for an advertisement request
- Start Value—This is used in cases where there are several subset advertisements. If the first (N) subset advertisement has been received and the subsequent one (N+1) has not, the Catalyst only requests advertisements from the (N+1)th one.

Related protocols

IEEE 802.1Q

Sponsor Source

VTP is a Cisco protocol.

Reference

http://www.cisco.com/en/US/products/hw/switches/ps708/products_configuration_guide_chapter09186a008019f048.html#1017196
Understanding and Configuring VLAN Trunking Protocol

Protocol Name

XOT: X.25 over TCP Protocol by Cisco

Protocol Description

The X.25 over TCP protocol (XOT) is designed by Cisco to transport X.25 over IP internets. The X.25 Packet Level requires a reliable link level below it and normally uses LAPB. XOT is a method of sending X.25 packets over IP internets by encapsulating the X.25 Packet Level in TCP packets.

TCP provides a reliable byte stream. X.25 requires that the layer below it provide message semantics, in particular the boundary between packets. To provide this, a small (4-bytes) XOT header is used between TCP and X.25. The primary content of this header is a length field, which is used to separate the X.25 packets within the TCP stream.

In general, the normal X.25 protocol packet formats and state transition rules apply to the X.25 layer in XOT. Exceptions to this are noted.

Protocol Structure

16	32bit
Version	Length

- Version - The version number. It must be 0. If no zero number is received, the TCP session must be closed.
- Length - The length of the packet. Values must be legal X.25 packet lengths. If the length field has an illegal value, then the TCP connection MUST be closed.

Related protocols

IP, TCP, X.25

Sponsor Source

XOT is a Cisco protocol and circulated in IETF (http://www.ietf.org) RFC1613.

Reference

http://www.javvin.com/protocol/rfc1613.pdf
Cisco Systems X.25 over TCP (XOT)

Novell NetWare and Protocols

Description

NetWare is a Novell network operating system (NOS) that provides transparent remote file access and numerous other distributed network services, including printer sharing and support for various applications such as electronic mail transfer and database access. NetWare specifies the upper five layers of the OSI reference model and runs on any media-access protocol (Layer 2). In addition, NetWare runs on virtually any kind of computer system, from PCs to mainframes. NetWare and its supporting protocols often coexist on the same physical channel with many other popular protocols, including TCP/IP, DECnet, and AppleTalk.

Novell NetWare, introduced in the early 1980s, is based on Xerox Network Systems (XNS) client-server architecture. Clients (sometimes called workstations) request services, such as file and printer access, from servers. NetWare's client/server architecture supports remote access, transparent to users, through remote procedure calls. A remote procedure call begins when the local computer program running on the client sends a procedure call to the remote server. The server then executes the remote procedure call and returns the requested information to the local client.

The most popular protocols in the Novell NetWare suite are:

IPX: Internetwork Packet Exchange protocol- Routing and networking protocol at layer 3. When a device to be communicated with is located on a different network, IPX routes the information to the destination through any intermediate networks. IPX is similar to IP (Internet Protocol) in the TCP/IP suite.

SPX: Sequenced Packet Exchange protocol, control protocol at the transport layer (layer 4) for reliable, connection-oriented datagram transmission. SPX is similar to TCP in the TCP/IP suite.

NCP: Network Core Protocol is a series of server routines designed to satisfy application requests coming from, for example, the NetWare shell. Services provided by NCP include file access, printer access, name management, accounting, security and file synchronization.

NetBIOS: Network Basic Input/Output System (NetBIOS) session-layer interface specification from IBM and Microsoft. NetWare's NetBIOS emulation software allows programs written to the industry-standard NetBIOS interface to run within the NetWare system.

NetWare application-layer services: NetWare Message Handling Service (NetWare MHS), Btrieve, NetWare Loadable Modules (NLMs), and various IBM connectivity features. NetWare MHS is a message delivery system that provides electronic mail transport. Btrieve is Novell's implementation of the binary tree (btree) database access mechanism. NLMs are implemented

as add-on modules that attach into the NetWare system. NLMs for alternate protocol stacks, communication services, database services, and many other services are currently available from Novell and third parties.

Since NetWare 5.0, all Novell network services can be run on top of TCP/IP. There, IPS and SPX became Novell legacy network and transport layer protocols.

Architecture

The following figure illustrates the NetWare protocol suite, the media-access protocols on which NetWare runs, and the relationship between the NetWare protocols and the OSI reference model.

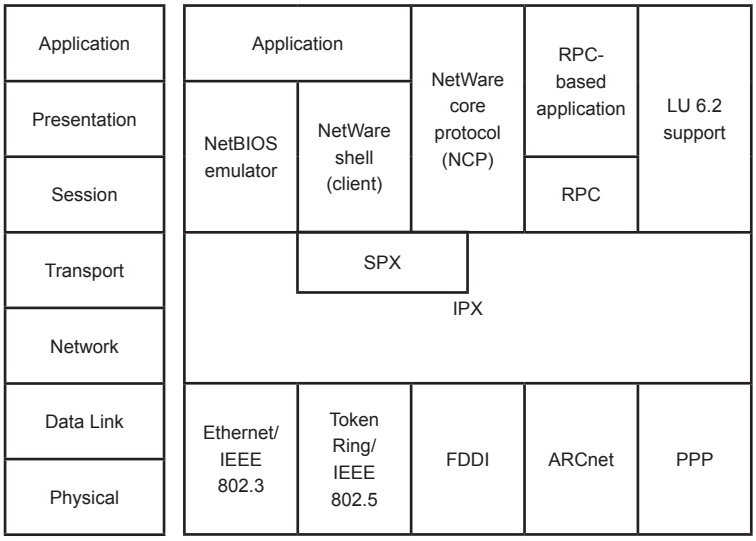


Figure 2-22: Novell Netware Protocol Stack Architecture

Related protocols

IPX, SPX, Novell NetBIOS, NCP, RPC

Sponsor Source

NetWare is a Novell Operating System.

Reference

http://www.cisco.com/univercd/cc/td/doc/cisintwk/ito_doc/netwarep.htm
NetWare Protocols

Protocol Name

IPX: Internetwork Packet Exchange protocol

Protocol Description

Internetwork Packet Exchange (IPX) is the legacy network protocol used by the Novell NetWare operating systems to route packets through an internetwork. IPX is a datagram protocol used for connectionless communications similar to IP (Internet Protocol) in the TCP/IP suite. Higher-level protocols, such as SPX and NCP, are used for additional error recovery services.

To make best-path routing decisions, IPX uses the services of a dynamic distance vector routing protocol such as Routing Information Protocol [RIP]) or NetWare Link-State Protocol [NLSP]).

Novell IPX network addresses are unique and are represented in a hexadecimal format that consists of two parts: a network number and a node number. The IPX network number, which is assigned by the network administrator, is 32 bits long. The node number, which usually is the Media Access Control (MAC) address for one of the system's network interface cards (NICs), is 48 bits long. IPX's use of a MAC address for the node number enables the system to send nodes to predict what MAC address to use on a data link.

Novell NetWare IPX supports four encapsulation schemes on a single router interface:

- Novell Proprietary—Also called 802.3 raw or Novell Ethernet_802.3, Novell proprietary serves as the initial encapsulation scheme that Novell uses.
- 802.3—Also called Novell_802.2, 802.3 is the standard IEEE 802.3 frame format.
- Ethernet version 2—Also called Ethernet-II or ARPA, Ethernet version 2 includes the standard Ethernet Version 2 header, which consists of Destination and Source Address fields followed by an EtherType field.
- SNAP—Also called Ethernet_SNAP, SNAP extends the IEEE 802.2 header by providing a type code similar to that defined in the Ethernet version 2 specification.

The maximum length of the data section of an IPX packet varies from a minimum of 30 bytes (the header only) depending on the lower layer MAC protocol (Ethernet or token ring) that is being used.

Protocol Structure

The NetWare IPX Packet Header:

8	16bit
Checksum	
Packet Length	
Transport control	Packet Type
Destination Network (4 bytes)	
Destination node (6 bytes)	
Destination socket (2 bytes)	
Source network (4 bytes)	
Source node (6 bytes)	
Source socket (2 bytes)	

- Checksum—Indicates that the checksum is not used when this 16-bit field is set to 1s (FFFF).
- Packet length—Specifies the length, in bytes, of a complete IPX datagram. IPX packets can be any length, up to the media maximum transmission unit (MTU) size (no packet fragmentation allowed).
- Transport control—Indicates the number of routers through which the packet has passed. When this value reaches 16, the packet is discarded under the assumption that a routing loop might be occurring.
- Packet type—Specifies which upper-layer protocol should receive the packet's information. It has two common values:
  - 5—Specifies Sequenced Packet Exchange (SPX)
  - 17—Specifies NetWare Core Protocol (NCP)
- Destination network, Destination node, and Destination socket—Specify destination information.
- Source network, Source node, and Source socket—Specify source information.

Related protocols

NetWare, SPX, RIP, NLSP

Sponsor Source

IPX is a Novell protocol.

Reference

http://www.novell.com/documentation/lg/nw65/index.html?page=/documentation/lg/nw65/ipx_enu/data/hc1w6pvi.html
IPX Structure
http://www.cisco.com/univercd/cc/td/doc/cisintwk/ito_doc/netwarep.htm
NetWare Protocols

Protocol Name

NCP: NetWare Core Protocol

Protocol Description

The Novell NetWare Core Protocol (NCP) manages access to the primary NetWare server resources. NCP makes procedure calls to the NetWare File Sharing Protocol (NFSP) that services requests for NetWare file and print resources. NCP is the principal protocol for transmitting information between a NetWare server and its clients.

NCP handles login requests and many other types of requests to the file system and the printing system. NCP is a client/server LAN protocol. Workstations create NCP requests and use IPX to send them over the network. At the server, NCP requests are received, unpacked, and interpreted.

NCP services include file access, file locking, security, tracking of resource allocation, event notification, synchronization with other servers, connection and communication, print services and queue and network management.

NCP uses the underlying Internetwork Packet Exchange Layer Services (IPX). More recent NetWare versions (after NetWare 5.0) can also use TCP/IP.

Protocol Structure

The format of the NCP Request header is shown below.

8	16bit
Request type	
Sequence number	Connection number low
Task number	Connection number high
Request code	

- Request type - Identifies the packet type:
  - 1111H. Allocate slot request
  - 2222H File server request.
  - 3333H File server reply.
  - 5555H Deallocate slot request.
  - 7777H Burst mode packet (BMP).
  - 9999H Positive acknowledge.
  - H signifies hexadecimal notation.
- Sequence number - Number used by the workstation and file server to identify packets which are sent and received.
- Connection number low - Low connection ID number assigned to the workstation.
- Task number - Identifies the operating system e.g., DOS, task.
- Connection number high - High Connection ID number assigned to the workstation. Used only on the 1000-user version of NetWare, on all other versions will be set to 0.
- Request code - Identifies the specific request function code.

The structure of the NCP Reply header is the same as the Request header, but the last 2 bytes differ after Connection Number High. This is shown below:

Completion code
Connection status

- Completion code - The completion code indicates whether or not the Client's request was successful. A value of 0 in the Completion Code field indicates that the request was successful. Any other value indicates an error.
- Connection status - The fourth bit in this byte will be set to 1 if DOWN is typed at the console prompt, to bring the server down.

Related protocols

NetWare, SPX, RIP, NLSP, IPX

Sponsor Source

NetWare Core Protocol (NCP) is a Novell protocol.

Reference

http://www.cisco.com/univercd/cc/td/doc/cisintwk/ito_doc/netwarep.htm
NetWare Protocols

Protocol Name

NLSP: NetWare Link Services Protocol

Protocol Description

The NetWare Link Services Protocol (NLSP) is a link-state routing protocol in the Novell NetWare architecture. NLSP is based on the OSI Intermediate System-to-Intermediate System (IS-IS) protocol and was designed to replace IPX RIP (Routing Information Protocol) and SAP (Service Advertisement Protocol), Novell's original routing protocols that were designed for small scale internetworks.

Compared to RIP and SAP, NLSP provides improved routing, better efficiency, and scalability. The following are the key features of the NLSP:

- NLSP-based routers use a reliable delivery protocol, so delivery is guaranteed.
- NLSP facilitates improved routing decisions because NLSP-based routers store a complete map of the network, not just next-hop information.
- NLSP is efficient, particularly over a WAN link, because its support of IPX header compression makes it possible to reduce the size of packets. NLSP also supports multicast addressing so that routing information is sent only to other NLSP routers, not to all devices, as RIP does.
- NLSP supports load balancing across parallel paths and improves link integrity. It periodically checks links for connectivity and for the data integrity of routing information.
- NLSP is scalabe because NLSP can support up to 127 hops (RIP supports only 15 hops) and permits hierarchical addressing of network nodes, which allows networks to contain thousands of LANs and servers.
- NLSP-based routers are backward compatible with RIP-based routers.

Similar to IS-IS, NLSP supports hierarchical routing with area, domain, and global internetwork components. Areas can be linked to create routing domains, and domains can be linked to create a global internetwork. NLSP supports three levels of hierarchical routing: Level 1, Level 2, and Level 3 routing.

An NLSP router extracts certain information from the adjacency database and adds locally derived information. Using this information, the router constructs a link-state packet (LSP) that describes its immediate neighbors. All LSPs constructed by all routers in the routing area make up the link-state database for the area. The link-state database is synchronized by reliably propagating LSPs throughout the routing area when a router observes a topology change. Two methods ensure that accurate topology-change information is propagated: flooding and receipt confirmation.

NLSP supports a hierarchical addressing scheme. Each routing area is identified by two 32-bit quantities: a network address and a mask.

Protocol Structure

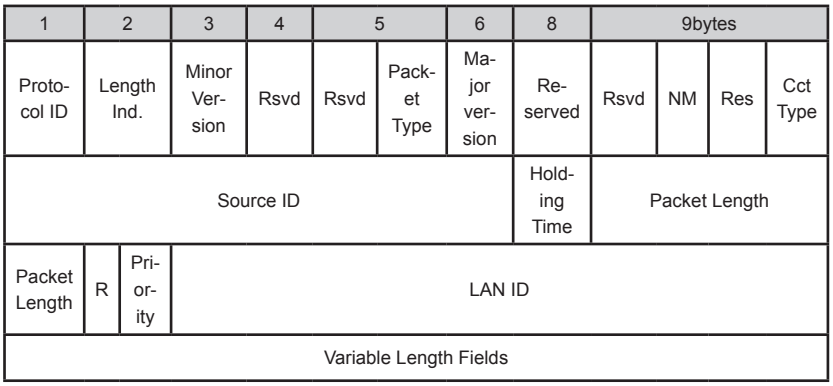
NLSP WAN Hello Packet:

<table>
<tr><th>1</th><th>2</th><th>3</th><th>4</th><th colspan="2">5</th><th>6</th><th>8</th><th colspan="3">9bytes</th></tr>
<tr><td>Protocol ID</td><td>Length Ind.</td><td>Minor Version</td><td>Rsvd</td><td>Rsvd</td><td>Packet Type</td><td>Major version</td><td>Reserved</td><td>Rsvd</td><td>State</td><td>Cct Type</td></tr>
<tr><td colspan="7">Source ID</td><td>Holding Time</td><td colspan="3">Packet Length</td></tr>
<tr><td>Packet Length</td><td>Local Wan Circuit ID</td><td colspan="9">Variable Length Fields</td></tr>
</table>

- Protocol ID—Identifies the NLSP routing layer with the 0x83 hex number.
- Length indicator—Determines the number of bytes in the fixed portion of the header.
- Minor version—Contains one possible decimal value and is ignored on receipt.
- Reserved—Contains no decimal values and is ignored on receipt.
- Packet type (5 bits)—Contains 17 possible decimal values.
- Major version—Contains one possible decimal value.
- Reserved—Contains no decimal values and is ignored on receipt.
- State (2 bits)—Sends the router's state associated with the link (0 = up, 1 = initializing, 2 = down).
- Circuit type (Cct type)—Consists of 2 bits. This field can have one of the following values:
  - 0—Reserved value; ignore entire packet.
  - 1—Level 1 routing only.
  - 2—Level 2 routing only. (The sender uses this link for Level 2 routing.)
  - 3—Both Level 1 and Level 2. (The sender is a Level 2 router and uses this link for Level 1 and Level 2 traffic.)
- Source ID—Serves as the system identifier of the sending router.
- Holding time—Contains the holding timer, in seconds, to be used for the sending router.
- Packet length—Determines the entire length of the packet, in bytes, including the NLSP header.
- Local WAN circuit ID—Acts as a unique identifier as-

signed to this circuit when it is created by the router.
- Variable length field—Consists of a series of optional fields.

NLSP LAN Hello Packet:



- Protocol ID—Identifies the NLSP routing layer with the 0x83 hex number.
- Length indicator—Determines the number of bytes in the fixed portion of the header (up to and including the LAN ID field).
- Minor version—Contains one possible decimal value and is ignored on receipt.
- Reserved—Contains no possible decimal values and is ignored on receipt.
- Packet type (5 bits)—Contains 15 possible decimal values.
- Major version—Contains one possible decimal value.
- Reserved—Contains no possible decimal values and is ignored on receipt.
- No multicast (NM) (1 bit)—Indicates, when set to 1, that the packet sender cannot receive traffic addressed to a multicast address. (Future packets on this LAN must be sent to the broadcast address.)
- Circuit type (Cct Type) (2 bits)—Can have one of the following values:
  - 0—Reserved value; ignore entire packet.
  - 1—Level 1 routing only.
  - 2—Level 2 routing only. (The sender uses this link for Level 2 routing.)
  - 3—Both Level 1 and Level 2. (The sender is a Level 2 router and uses this link for Level 1 and Level 2 traffic.)
- Source ID—Contains the system ID of the sending router.
- Holding time—Contains the holding timer, in seconds, to be used for the sending router.
- Packet length—Determines the entire length of the packet, in bytes, including the NLSP header.
- R—Contains no possible decimal values and is ignored on receipt.
- Priority (7 bits)—Serves as the priority associated with being the LAN Level 1 designated router. (Higher numbers have higher priority.)
- LAN ID—Contains the system ID (6 bytes) of the LAN Level 1 designated router, followed by a field assigned by that designated router.
- Variable length fields—Consists of a series of optional fields.

Related protocols

NetWare, SPX, RIP, NCP, IPX, SAP

Sponsor Source

NetWare Link Service Protocol (NLSP) is a Novell protocol.

Reference

http://www.cisco.com/univercd/cc/td/doc/cisintwk/ito_doc/nlsp.htm
NetWare Link Services Protocol

Protocol Name

SPX: Sequenced Packet Exchange protocol

Protocol Description

The Sequenced Packet Exchange (SPX) protocol is Novell's legacy transport layer protocol providing a packet delivery service for Novell NetWare network. SPX is based on the Xerox Sequenced Packet Protocol (SPP). SPX, operates on top of IPX and is used in Novell NetWare (prior to NetWare 5.0) systems for communications in client/server application programs, e.g. BTRIEVE (ISAM manager). SPX performs equivalent functions to TCP. The newer versions of NetWare services are run on top of TCP/IP.

IPX receives packets from the network and passes on those for SPX to handle. SPX guarantees that packets are received intact, in the order they were sent, and eliminates duplicate packets. SPX prepares the sequence of packets that a message is divided into and manages the reassembly of received packets, confirming that all have been received and requesting retransmission when they haven't. SPX works directly with the Internetwork Packet Exchange (IPX) protocol, which manages the forwarding of packets in the network. SPX does not provide connections to the file server itself, which uses the NetWare Core Protocol (NCP). SPX has been extended as SPX-II (SPX2)

SPX does not provide group broadcast support; packets can only be sent to a single session partner. SPX can detect if its partner has disappeared.

Protocol Structure

The structure of the SPX packet is shown in the following illustration:

8	16bit
Connection control flag	Datastream type
Source connection ID	
Destination connection ID	
Sequence number	
Acknowledge number	
Allocation number	
Data (0-534 bytes)	

- Connection control flag - Four flags which control the bi-directional flow of data across an SPX connection. These flags have a value of 1 when set and 0 if not set.
  - Bit 4 Eom: End of message.
  - Bit 5 Att: Attention bit, not used by SPX.
  - Bit 6 Ack: Acknowledge required.
  - Bit 7 Sys: Transport control.
- Datastream type - Specifies the data within the packet:
- Source connection ID - A 16-bit number assigned by SPX to identify the connection.
- Destination connection ID - The reference number used to identify the target end of the transport connection.
- Sequence number - A 16-bit number, managed by SPX, which indicates the number of packets transmitted.
- Acknowledge number - A 16-bit number, indicating the next expected packet.
- Allocation number - A 16-bit number, indicating the number of packets sent but not yet acknowledged.

The SPX II header is the same as the SPX header described above, except for the following differences:

- Connection control flag - Bit 2 - Size negotiation. Bit 3 - SPX II type.
- Datastream type - 252 - Orderly release request. 253 - Orderly release acknowledgment.

There is also an additional 2-byte Extended Acknowledgement field at the end.

Related protocols

NetWare, SPX, RIP, NLSP, IPX, NCP

Sponsor Source

SPX is a Novell protocol.

Reference

http://www.cisco.com/univercd/cc/td/doc/cisintwk/ito_doc/netwarep.htm
NetWare Protocols
http://docsrv.sco.com/SDK_netware/CTOC-Enhanced_Sequenced_Packet_Exchange_SPXII_Protocol.html
Enhanced Sequenced Packet Exchange protocol (SPXII)

IBM Systems Network Architecture (SNA) and Protocols

Protocol Description

Along with the OSI Model, the Systems Network Architecture (SNA) proposed by IBM, is one of the most popular network architecture models. Although the SNA model is now considered a legacy networking model, SNA is still widely deployed. SNA was designed around the host-to-terminal communication model that IBM's mainframes use. IBM expanded the SNA protocol to support peer-to-peer networking. This expansion was deemed Advanced Peer-to-Peer Networking (APPN) and Advanced Program-to-Program Communication (APPC). Advanced Peer-to-Peer Networking (APPN) represents IBM's second-generation SNA. In creating APPN, IBM moved SNA from a hierarchical, mainframe-centric environment to a peer-to-peer (P2P) networking environment. At the heart of APPN is an IBM architecture that supports peer-based communications, directory services, and routing between two or more APPC systems that are not directly attached.

The IBM SNA model has many similarities with the OSI 7 layers model. However, the SNA model has only 6 layers and does not define specific protocols for its physical control layer. The physical control layer is assumed to be implemented via other standards. The functions of each SNA layer are described as follows:

- Data link control (DLC)—Defines several protocols, including the Synchronous Data Link Control (SDLC) protocol for hierarchical communication, and the Token Ring Network communication protocol for LAN communication between peers. SDLC provided a foundation for ISO HDSL and IEEE 802.2.
- Path control—Performs many OSI network layer functions, including routing and datagram segmentation and reassembly (SAR)
- Transmission control—Provides a reliable end-to-end connection service (similar to TCP), as well as encrypting and decrypting services
- Data flow control—Manages request and response processing, determines whose turn it is to communicate, groups messages, and interrupts data flow on request
- Presentation services—Specify data-transformation algorithms that translate data from one format to another, coordinate resource sharing, and synchronize transaction operations
- Transaction services—Provides application services in the form of programs that implement distributed processing or management services

The following figure illustrates how the IBM SNA model maps to the OSI 7 layers reference model.

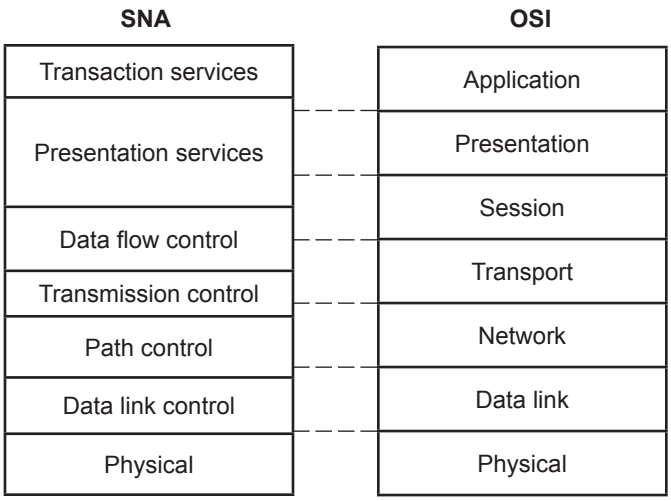


Figure 2-23: IBM SNA vs. OSI Model

Protocol Structure

The IBM main protocols are listed as follows:

SNA Layer	IBM Protocols
Transaction Services	SMB: Server Message Block protocol
Presentation Services	IPDS: Intelligent Printer Data Stream
Data Flow Control	APPC: Advanced Program to Program Communication (LU 6.2)
	LU: Logic Units - LU 0, LU 1, LU2, LU 3 LU 6.2
	NetBIOS: Network Basic Input Output System
Transmission Control	NetBEUI: NetBIOS Extended User Interface
Path Control	NAU: Network Addressable Units
	APPN: Advanced Peer to Peer Networking
Data Link Control	DLSw: Data Link Switching protocol
	QLLC: Qualified Logic Link Control for SNA over X.25
	SDLC: Synchronous Data Link Control protocol

Reference

http://www-306.ibm.com/software/network
Systems Network Architecture
http://publib-b.boulder.ibm.com/Redbooks.nsf/RedbookAbstracts/
Inside APPN and HPR - The Essential Guide to the Next-Generation SNA
http://publibz.boulder.ibm.com/cgi-bin/bookmgr_OS390/BOOKS/
SNA APPN Architecture Reference

Protocol Name

IBM SMB: Server Message Block protocol

Protocol Description

Server Message Block (SMB) protocol is an IBM protocol for sharing files, printers, serial ports, etc. between computers. The SMB protocol can be used over the Internet on top of the TCP/IP protocol or other network protocols such as Internetwork Packet Exchange (Novell IPX) and NetBEUI.

SMB is a client server, request-response protocol, which provides a method for client applications in a computer to read and write to files and to request services from server programs in various types of network environment. Using the SMB protocol, an application can access files as well as other resources including printers, mailslots and named pipes, at a remote server.

In the TCP/IP environment, clients connect to servers using NetBIOS over TCP/IP (or NetBEUI/TCP or SPX/IPX). Once they have established a connection, clients can then send SMB commands to the server that allow them to access shares, open files, read and write files, and generally do all the things that you want to do with a file system.

Microsoft Windows operating systems since Windows 95 include client and server SMB protocol support. Microsoft has offered an open source version of SMB for the Internet, called the Common Internet File System (CIFS), which provides more flexibility than existing Internet applications such as the File Transfer Protocol (FTP). For UNIX systems, a shareware program, Samba, is available.

The Server Message Block (SMB) protocol defines two levels of security:

- Share Level - Protection is applied at the share level on a server. Each share can have a password, and a client only needs that password to access all files under that share.
- User Level - Protection is applied to individual files in each share and is based on user access rights. Each user (client) must log in to the server and be authenticated by the server. When it is authenticated, the client is given a UID which it must present on all subsequent accesses to the server.

Protocol Structure

SMB has many variants to handle the complexity of the underneath network environments in which it is employed. The following table displays part of the SBM variants:

SMB Protocol Variant	Protocol Name	Comments
PC NETWORK PROGRAM 1.0	Core Protocol	The original version of SMB as defined in IBM's PC Network Program. Some versions were called PCLAN1.0
MICROSOFT NETWORKS 1.03	Core Plus Protocol	Included Lock&Read and Write&Unlock SMBs with different versions of raw read and raw write SMBs
MICROSOFT NETWORKS 3.0	DOS LAN Manager 1.0	The same as LANMAN1.0, but OS/2 errors must be translated to DOS errors.
LANMAN1.0	LAN Manager 1.0	The full LANMAN1.0 protocol.
DOS LM1.2X002	LAN Manager 2.0	The same as LM1.2X002, but errors must be translated to DOS errors.
LM1.2X002	LAN Manager 2.0	The full LANMAN2.0 protocol.
DOS LANMAN2.1	LAN Manager 2.1	The same as LANMAN2.1, but errors must be translated to DOS errors.
LANMAN2.1	LAN Manager 2.1	The full LANMAN2.1 protocol.
Windows for Workgroups 3.1a	LAN Manager 2.1	Windows for Workgroups 1.0?
NT LM 0.12	NT LAN Manager 1.0	Contains special SMBs for NT
Samba	NT LAN Manager 1.0	Samba's version of NT LM 0.12?
CIFS 1.0	NT LAN Manager 1.0	Really NT LM 0.12 plus a bit

Related protocols

TCP, SPX, IPX, NetBIOS, NetBEUI

Sponsor Source

Server Message Block (SMB) is an IBM protocol.

Reference

http://samba.anu.edu.au/cifs/docs/what-is-smb.html
What is SMB?

Protocol Name

APPC: Advanced Program to Program Communications (SNA LU6.2)

Protocol Description

Advanced Program-to-Program Communications (APPC), a protocol roughly in the OSI resentation and session layer, is a programming interface standard that allows interconnected systems to communicate and share the processing of programs. Originally developed by IBM as a remote transaction processing tool between Logic Units (LUs), APPC is now used to provide distributed services within a heterogeneous computing environment.

APPC software enables high-speed communication to take place between programs residing on different computers, and between workstations and midrange and mainframe computer servers. APPC allows user written programs to perform transactions in a Client-Server network. APPC is a standardized application programming interface which allows an application to use a pre-defined set of VERBS/API for sending and receiving data to/from another program located in a remote node. This set of verbs/API, can only be used with an LU 6.2, which is why the terms APPC, LU 6.2 and PU type 2.1 are very often used interchangeably.

LU 6.2 is the set of SNA parameters used to support APPC when it runs in the SNA network. Basically, LU6.2 acts as an interface, or protocol boundary, between SNA and an end user's application. However, not all APPC communications are based on SNA. APPC also runs between two MVS or CMS program using mainframe system services. Under Anynet, APPC can also run over the TCP/IP protocol used by the Internet.

Protocol Structure

APPC functions and commands:

Sessions, Conversations and Job Management - A client generates a request. It is passed to the subsystem on the client node and is sent through the network to the subsystem on the server node. A subsystem component called the Attach Manager then either queues the request to a running program or starts a new server program.

Attach Manager and Transaction Programs:

- Allocate - acquires temporary ownership of one of the sessions to the server node.
- Deallocate - frees the session and ends the conversation.

Sending Data (or Objects)

- Send_Data - moves a record of data from the application memory to buffers controlled by the subsystem.
- Receive_and_Wait
- Send_Error - Send_Error breaks the logical chain of incoming records. APPC is responsible for flushing any pending data (from the server node, the network, or the client node). As soon as possible, the client is notified of the problem with a characteristic return code on the next APPC operation.
- Confirm - Any pending data is sent, and after the other program receives the data it gets an indication that Confirm is pending. If everything is acceptable, then the correct response is to call the Confirmed verb.

Higher Level Programming:

- Send file (disk to network)
- Send SQL table (DBMS to network)
- Send stack (REXX)
- Send hyperspace (MVS)
- Send clipboard (Windows)

Related protocols

SNA, APPN

Sponsor Source

APPC (LU6.2) is an IBM protocol.

Reference

http://publib-b.boulder.ibm.com/Redbooks.nsf/RedbookAbstracts/gg242537.html?Open
A CM/2 APPC/APPN Tutorial
http://publib-b.boulder.ibm.com/Redbooks.nsf/RedbookAbstracts/sg243669.html?Open
Inside APPN and HPR - The Essential Guide to the Next-Generation SNA

Protocol Name

SNA NAU: Network Accessible Units (PU, LU and CP)

Protocol Description

Network Accessible Units (NAUs), formaerly called “Network Addressable Units”, are the IBM Systems Network Architecture (SNA) components to facilitate the communication between a Transaction Program (TP) and the SNA network. NAUs are unique network resources that can be accessed through unique local addresses by other network resources. SNA provides the following types of NAUs:

PU- Physical units

Each SNA node contains a physical unit (PU). The PU manages resources (such as link resources) and supports communication with a host.

LU-Logical Units

Each SNA node contains one or more logical units (LUs). An LU provides a set of functions that are used by TPs and end users to provide access to the network. LUs communicate directly with local TPs and devices.

SNA defines several types of LUs, each optimized for a specific class of applications. LUs of different types cannot communicate with each other, but LUs of the same type can communicate even though they reside on different kinds of systems.

For example, a TP running on a workstation that uses the AIX operating system can communicate with a TP on an AS/400 computer as easily as it can with a TP on another AIX workstation, as long as both TPs use the same LU type.

IBM Communication Server (CS/AIX) supports the following LU types:

LU 6.2 (for APPC, 5250, APPC Application Suite, and CPI-C)

LU 6.2 supports program-to-program communication in a distributed data processing environment. The LU 6.2 data stream is either an SNA general data stream (GDS), which is a structured-field data stream, or a user-defined data stream. LU 6.2 can be used for communication between two type 5 nodes, a type 5 node and a type 2.0 or 2.1 node, or two type 2.1 nodes. (Type 2.1 nodes can serve as APPN nodes.)

This LU type provides more functions and greater flexibility than any other LU type. Unless you are constrained by existing hardware or software, LU 6.2 is the logical choice when developing new applications.

LU 3 (for 3270 printing)

LU 3 supports application programs and printers using the SNA 3270 data stream.

For example, LU 3 can support an application program running under Customer Information Control System (CICS) and sending data to an IBM 3262 printer attached to an IBM 3174 Establishment Controller.

LU 2 (for 3270 displays)

LU 2 supports application programs and display workstations communicating in an interactive environment using the SNA 3270 data stream. Type 2 LUs also use the SNA 3270 data stream for file transfer.

For example, the LU 2 protocol can support 3270 emulation programs, which enable workstations to perform the functions of IBM 3270-family terminals. In addition, LU 2 is used by other programs to communicate with host applications that normally provide output to 3270 display devices. Such TPs enable the workstation to achieve a form of cooperative processing with the host.

LU 1 (for SCS printing and RJE)

LU 1 supports application programs and single- or multiple-device data processing workstations communicating in an interactive, batch-data transfer, or distributed data processing environment. The data streams used by LU type 1 conform to the SNA character string or Document Content Architecture (DCA).

For example, LU type 1 can support an application program running under Information Management System/Virtual Storage (IMS/VS) and communicating with an IBM 8100 Information System. This enables a workstation operator to correct a database that the application program maintains.

Applications that use LU 1 are often described as remote job entry (RJE) applications.

LU 0 (for LUA)

LU 0, an early LU definition, supports primitive program-to-program communication. Certain host database systems, such as IMS/VS (Information Management System/Virtual Storage) and some point-of-sale systems for the retail and banking industries (such as the IBM 4680 Store System Operating System) use LU 0. Current releases of these products also support LU 6.2 communication, which is the preferred protocol for new applications.

CP- Control Points

A control point (CP) is an NAU that manages network resources within its domain, controlling resource activation, deactivation, and status monitoring. The CP manages both physical resources such as links, and logical information such as network addresses.

SNA defines the following types of network control points:

System services control point

On a type 5 node, the CP is called a system services control point (SSCP). It manages and controls the network resources in a subarea network. For example, an SSCP can use a directory of network resources to locate a specific LU under its control, and can establish communication between two LUs in its domain. An SSCP can also cooperate with other SSCPs to establish connectivity between LUs in different subarea domains.

The SSCP also provides an interface to network operators at the host system, who can inspect and control resources in the network.

Physical unit control point

On type 4 nodes and type 2.0 nodes in a subarea network, the control point is called a physical unit control point (PUCP).

Control point

On type 2.1 nodes, the control point provides both PU and LU functions, such as activating local link stations, interacting with a local operator, and managing local resources. It can also provide network services, such as partner LU location and route selection for local LUs.

In a subarea network, the CP on a CS/AIX node acts as a type 2.0 PU. It communicates with an SSCP on a host and does not communicate with other CPs in the subarea network.

When participating in an APPN network, the CP exchanges network control information with the CPs in adjacent nodes. The CP can also function as an independent LU of type 6.2. The CP acts as the default LU for TPs on the local node.

Related protocols

SNA, APPN, APPC, SSCP, LU0, LU1, Lu2, LU3, LU6.2,

Sponsor Source

NAUs are IBM SNA components.

Reference

http://www-306.ibm.com/software/network/commserver/library/publications/csaix_60/dyvl1m02.htm#ToC_14
Systems Network Architecture

Protocol Name

NetBIOS: Network Basic Input Output System

Protocol Description

Network Basic Input Output System (NetBIOS) was created by IBM. NetBIOS defines a software interface and standard methods providing a communication interface between the application program and the attached medium. NetBIOS, a session layer protocol, is used in various LAN (Ethernet, Token Ring etc) as well as WAN environments such as TCP/IP, PPP and X.25 networks.

NetBIOS frees the application from having to understand the details of the network, including error recovery (in session mode). A NetBIOS request is provided in the form of a Network Control Block (NCB) which, among other things, specifies a message location and the name of a destination.

NetBIOS provides the session and transport services described in the Open Systems Interconnection (OSI) model. However, it does not provide a standard frame or data format for transmission. A standard frame format is provided by NetBEUI (NetBIOS Extended User Interface), which provides transport and network layer support of NetBIOS.

NetBIOS provides two communication modes: session or datagram. Session mode lets two computers establish a connection for a “conversation,” allows larger messages to be handled, and provides error detection and recovery. Datagram mode is “connectionless” (each message is sent independently), messages must be smaller, and the application is responsible for error detection and recovery. Datagram mode also supports the broadcast of a message to every computer on the LAN.

NetBIOS names are 16 bytes long (padded if necessary) and there are very few restraints on the byte values which can be used. There are three methods of mapping NetBIOS names to IP addresses on small networks that don’t perform routing:

1. IP broadcasting - A data packet with the NetBIOS computer name is broadcast when an associated address is not in the local cache. The host with that name returns its address.
2. The lmhosts file - This is a file that maps IP addresses and NetBIOS computer names.
3. NBNS - NetBIOS Name Server. A server that maps NetBIOS names to IP addresses. This service is provided by the nmbd daemon on Linux.

Protocol Structure

NetBIOS packets have many different formats depending on the services and message types as well as on the transport protocols used to carry the NetBIOS packets. NetBIOS has three basic services: NAME, SESSION and DATAGRAM. As an example, we display the NetBIOS name packet format in the TCP/IP environment:

Header (12 bytes)
Question Entry (variable)
Answer Resource Records (variable)
Authority Resource Records (variable)
Additional Resource Records (variable)

The format of the NetBIOS header is shown below:

2	2	1	1	2	2	2 bytes
Length	Delimi-nator	Com-mand	Data1	Data2	XMIT Cor	RSP Cor
Destination name (16 bytes)						
Source name (16 bytes)						

- Len - The length of the NETBIOS header.
- Deliminator - A delimiter indicating that subsequent data is destined for the NetBIOS function.
- Command - A specific protocol command that indicates the type of function of the frame.
- Data 1 - One byte of optional data per specific command.
- Data 2 - Two bytes of optional data per specific command.
- Xmit/response correlator - Used to associate received responses with transmitted requests.
- Destination name/num - In non-session frames this field contains the 16-character name.
- Source name/num - In non-session frames this field contains the 16-character source name. In session frames this field contains a 1 byte source session number.

Related protocols

TCP, SMP, Ethernet, Token Ring, X.25, UDP, IPX, NetBEUI, PPP

Sponsor Source

NetBIOS and NetBEUI are IBM protocols.

Reference

http://ourworld.compuserve.com/homepages/TimothyDEvans/contents.htm
NetBios, NetBEUI, NBF, SMB, CIFS Networking
http://www.javvin.com/protocol/rfc1001.pdf
PROTOCOL STANDARD FOR A NetBIOS SERVICE ON A TCP/UDP TRANSPORT: CONCEPTS AND METHODS
http://www.javvin.com/protocol/rfc1002.pdf
PROTOCOL STANDARD FOR A NetBIOS SERVICE ON A TCP/UDP TRANSPORT: DETAILED SPECIFICATIONS

Protocol Name

NetBEUI: NetBIOS Extended User Interface

Protocol Description

NetBIOS Extended User Interface (NetBEUI) is an extended version of NetBIOS, that lets computers communicate within a local area network. NetBEUI formalizes the frame format that was not specified as part of NetBIOS, so is sometimes called the NetBIOS frame (NBF) protocol.

NetBEUI provides data transportation but it is not a routable transport protocol. NetBEUI works at the Transport and Network layers of a local area network (LAN). NetBEUI is a good performance choice for communication within a single LAN. For internetwork routing, its interface must be adapted to other protocols such as Internetwork Packet Exchange (IPX) or TCP/IP. Very often, both NetBEUI and TCP/IP are installed in each computer and the server is set up to use NetBEUI for communication within the LAN and TCP/IP for communication beyond the LAN.

NetBIOS and NetBEUI are developed by IBM for its LAN Manager product and have been adopted by Microsoft for its Windows NT, XP and 2000, LAN Manager, and Windows for Workgroups products. Novell, Hewlett-Packard and DEC use them in comparable products.

Protocol Structure

NetBEUI frame header is the same as for NETBIOS:

2	2	1	1	2	2	2 bytes
Length	Delimi-nator	Com-mand	Data1	Data2	XMIT Cor	RSP Cor
Destination name (16 bytes)						
Source name (16 bytes)						

- Length - The length of the header.
- Deliminator - A delimiter indicating that subsequent data is destined for the NetBIOS function.
- Command - A specific protocol command that indicates the type of function of the frame.
- Data 1 - One byte of optional data per specific command.
- Data 2 - Two bytes of optional data per specific command.
- Xmit/response correlator - Used to associate received responses with transmitted requests.
- Destination name/num - In non-session frames this field contains the 16-character name.
- Source name/num - In non-session frames this field contains the 16-character source name. In session frames this field contains a 1 byte source session number.

Related protocols

TCP, SMP, Ethernet, Token Ring, X.25, UDP, NetBIOS, PPP

Sponsor Source

NetBIOS and NetBEUI are IBM protocols.

Reference

http://ourworld.compuserve.com/homepages/TimothyDEvans/contents.htm
NetBios, NetBEUI, NBF, SMB, CIFS Networking
http://www.javvin.com/protocol/rfc1001.pdf
PROTOCOL STANDARD FOR A NetBIOS SERVICE ON A TCP/UDP TRANSPORT: CONCEPTS AND METHODS
http://www.javvin.com/protocol/rfc1002.pdf
PROTOCOL STANDARD FOR A NetBIOS SERVICE ON A TCP/UDP TRANSPORT: DETAILED SPECIFICATIONS

Protocol Name

APPN: Advanced Peer-to-Peer Networking

Protocol Description

Advanced Peer-to-Peer Networking (APPN) is an enhancement to the original IBM SNA architecture. APPN, which includes a group of protocols, handles session establishment between peer nodes, dynamic transparent route calculation, and traffic prioritization. Using APPN, a group of computers can be automatically configured by one of the computers acting as a network controller so that peer programs in various computers will be able to communicate with other using specified network routing.

APPN features include:

- Better distributed network control; because the organization is peer-to-peer rather than solely hierarchical, terminal failures can be isolated
- Dynamic peer-to-peer exchange of information about network topology, which enables easier connections, reconfigurations, and routing
- Dynamic definition of available network resources
- Automation of resouce registration and directory lookup
- Flexibility, which allows APPN to be used in any type of network topology

An APPN network is composed of three types of APPN node:

- Low Entry Networking (LEN) Node - An APPN LEN node provides peer to peer connectivity with all other APPN nodes.
- End Node - An End Node is similar to a LEN node in that it participates at the periphery of an APPN network. An End Node includes a Control Point (CP) for network control information exchange with an adjacent network node.
- Network Node - The backbone of an APPN network is composed of one or more Network Nodes which provide network services to attached LEN and End Nodes.

The APPN network has the following major functional processors:

- Connectivity - The first phase of operation in an APPN network is to establish a physical link between two nodes. When it has been established, the capabilities of the two attached nodes are exchanged using XIDs. At this point, the newly attached node is integrated into the network.
- Location of a Targeted LU - Information about the resources (currently only LUs) within the network is maintained in a database which is distributed across the End and Network Nodes in the network. End Nodes hold a directory of their local LUs. If the remote LU is found in the directory, a directed search message is sent across the network to the remote machine to ensure that the LU has not moved since it was last used or registered. If the local search is unsuccessful, a broadcast search is initiated across the network. When the node containing the remote LU receives a directed or broadcast search message, it sends back a positive response. A negative response is sent back if a directed or broadcast search fails to find the remote LU.
- Route Selection - When a remote LU has been located, the originating Network Node server calculates the best route across the network for a session between the two LUs. Every Network Node in the APPN network backbone maintains a replicated topology database. This is used to calculate the best route for a particular session, based on the required class of service for that session. The class of service specifies acceptable values for such session parameters as propagation delay, throughput, cost and security. The route chosen by the originating Network Node server is encoded in a route selection control vector (RSCV).
- Session Initiation - A BIND is used to establish the session. The RSCV describing the session route is appended to the BIND. The BIND traverses the network following this route. Each intermediate node puts a session connector for that session in place, which links the incoming and outgoing paths for data on the session.
- Data Transfer - Session data follow the path of the session connectors set up by the initial BIND. Adaptive pacing is used between each node on the route. The session connectors on each intermediate node are also responsible for segmentation and segment assembly when the incoming and outgoing links support different segment sizes.
- Dependent LU Requestor - Dependent LUs require a host based System Services Control Point (SSCP) for LU-LU session initiation and management. This means that dependent LUs must be directly attached to a host via a single data link.
- High-performance routing (HPR) - HPR is an extension to the APPN architecture. HPR can be implemented on an APPN network node or an APPN end node. HPR does not change the basic functions of the architecture. HPR has the following key functions:
  - Improves the performance of APPN routing by taking advantage of high-speed, reliable links
  - Improves data throughput by using a new rate-based congestion control mechanism
  - Supports nondisruptive re-routing of sessions around failed links or nodes
  - Reduces the storage and buffering required in intermediate nodes.

Protocol Structure

A simple APPN network is illustrated in the diagram below:

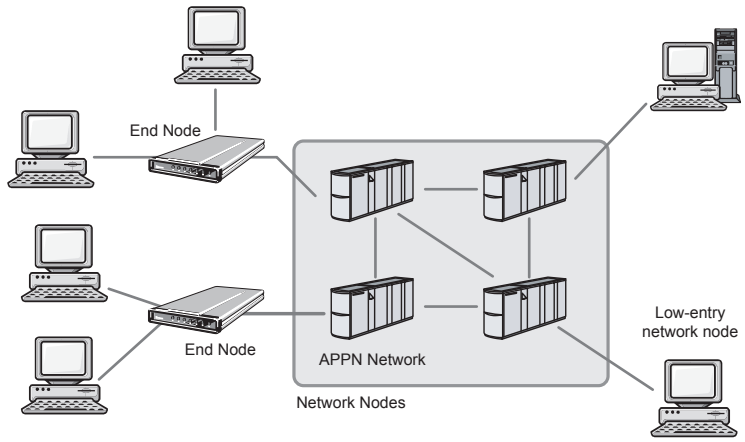


Figure 2-24: IBM APPN Network Illustration.

Related protocols

SNA, APPC

Sponsor Source

APPN is an IBM network architecture, extended from the IBM SNA.

Reference

http://publib-b.boulder.ibm.com/Redbooks.nsf/RedbookAbstracts/sg243669.html?Open
Inside APPN and HPR - The Essential Guide to the Next-Generation SNA
http://publibz.boulder.ibm.com/cgi-bin/bookmgr_OS390/BOOKS/D50L0000/CCONTENTS
SNA APPN Architecture Reference
http://www.cisco.com/univercd/cc/td/doc/cisintwk/idg4/nd2006.htm#17531
Designing APPN Internetworks
http://www.javvin.com/protocol/rfc2353.pdf
APPN/HPR in IP Networks (APPN Implementers' Workshop Closed Pages Document)

Protocol Name

DLSw: Data-Link Switching protocol

Protocol Description

Data-link switching (DLSw) provides a forward mechanism for transporting IBM Systems Network Architecture (SNA) and network basic input/output system (NetBIOS) traffic over an IP network. DLSw does not provide full routing, but instead provides switching at the SNA Data Link layer (i.e., layer 2 in the SNA architecture) and encapsulation in TCP/IP for transport over the Internet.

DLSw, originally a proprietary IBM protocol, was adopted by IETF as a standard. DSLw version 1 (DSLw v1) defines three primary functions:

- The Switch-to-Switch Protocol (SSP) is the protocol maintained between two DLSw nodes or routers.
- The termination of SNA data-link control (DLC) connections helps to reduce the likelihood of link layer timeouts across WANs.
- The local mapping of DLC connections to a DLSw circuit.

DLSw version 2 (DLSw v2), which was introduced in 1997 in IETF, provides the following enhancements to the version 1:

- IP multicast
- UDP unicast responses to DLSw broadcasts
- Enhanced peer-on-demand routing
- Expedited TCP connections

Each of these features enables DLSw as a scalable technology over WANs. In DLSw Version 1, transactions occur with TCP. As a result, many operations in a DLSw environment consume circuits between peers. For example, a multicast requires multiple TCP connections from the source to each peer. With DLSw Version 2, multicast is distributed using unreliable transport following traditional multicast methods.

Cisco supports a third version of DLSw called DLSw+. DLSw+ predates DLSw Version 2 and provides even further enhancements to basic DLSw.

Protocol Structure

8	16	24	32bit
Version number	Header Length	Message Length	
Remote data link correlator			
Remote DLC port ID			
Reserved Field		Message type	Flow control byte

- Version number - Set to 0x31 (ASCII 1) indicating a decimal value of 49. This is used to indicate DLSw version 1.
- Header length - Set to 0x48 for control messages and 0x10 for information and Independent Flow Control messages.
- Message length - Specifies the number of bytes within the data field following the header.
- Remote data link correlator - Works in tandem with the remote DLC port ID to form a 64-bit circuit ID that identifies the DLC circuit within a single DLSw node. The circuit ID is unique in a single DLSw node and is assigned locally. An end-to-end circuit is identified by a pair of circuit IDs that, along with the data-link IDs, uniquely identifies a single end-to-end circuit.
- Remote DLC port ID - Works in tandem with the remote data-link correlator to form a 64-bit circuit ID that identifies the DLC circuit within a single DLSw node. The contents of the DLC and DLC Port ID have local significance only. The values received from a partner DLSw must not be interpreted by the DLSw that receives them and should be echoed “as is” to a partner DLSw in subsequent messages.
- Message type - Indicates a specific DLSw message type. The value is specified in two different fields (offset 14 and 23 decimal) of the control message header. Only the first field is used when parsing a received SSP message. The second field is ignored by new implementations on reception, but is retained for backward compatibility.
- Flow control byte - Carries the flow-control indicator, flow-control acknowledgment, and flow-control operator bits.

Related protocols

SDLC, NetBIOS, TCP, SMP, Ethernet, Token Ring, SNA

Sponsor Source

Data-Link Switching (DSLw) was originated by IBM and adopted as a standard by IETF.

Reference

http://www.javvin.com/protocol/rfc1795.pdf
Data Link Switching: Switch-to-Switch Protocol AIW DLSw RIG: DLSw Closed Pages, DLSw Standard Version 1.0
http://www.javvin.com/protocol/rfc2166.pdf
DLSw v2.0 Enhancements

Protocol Name

QLLC: Qualified Logic Link Control

Protocol Description

Qualified Logical Link Control (QLLC) is an IBM-defined data-link-layer protocol that allows SNA data to be transported across X.25 networks. When SNA is used over X.25, it uses the qualifier-bit (Q-bit) in the X.25 packet header to indicate special link control information. This information is relevant for SNA control between the two systems communicating with each other but is of no concern to X.25 link control. These qualified packets help SNA to determine who is calling whom between the two communicating systems and indicate such items as maximum message size.

QLLC commands are implemented in X.25 packets with the use of the Q-bit. X.25 packets containing QLLC primitives are typically 5 bytes, or the length of the X.25 packet header, plus 2 bytes of QLLC control information. After the QLLC connection is established, the X.25 connection's unique virtual circuit is used to forward data traffic. LLC (Logical Link Control) is a subset of HDLC (High Level Data Link Control). SDLC (Synchronous Data Link Control) and QLLC are also subsets of HDLC.

Typical QLLC network architecture is shown below:

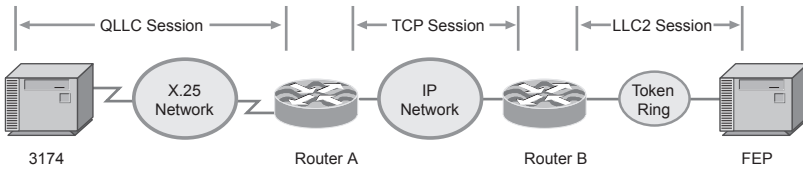


Figure 2-25: QLLC Network Architecture

QLLC supports the following X.25 optional facilities:

- Modulo 8/128 packet sequence numbering
- Closed user groups
- Recognized private operating agencies
- Network user identification
- Reverse charging
- Packet-size negotiation
- Window-size negotiation
- Throughput class negotiation

Protocol Structure

QLLC has the same frame structure as HDLC with the following frame types:

QRR	Receive Ready.
QDISC	Disconnect.
QUA	Unnumbered Acknowledgement.
QDM	Disconnect Mode.
QFRMR	Frame Reject.
QTEST	Test.
QRD	Request Disconnect.
QXID	Exchange Identification.
QSM	Set Mode.

Related protocols

SNA, APPN, X.25, HDCL, SDCL

Sponsor Source

QLLC is an IBM protocol.

Protocol Name

SDLC: Synchronous Data Link Control

Protocol Description

The Synchronous Data Link Control (SDLC) protocol is an IBM data link layer protocol for use in the Systems Network Architecture (SNA) environment.

The data link control Layer provides the error-free movement of data between the Network Addressable Units (NAUs) within a given communication network via the Synchronous Data Link Control (SDLC) Protocol. The flow of information passes down from the higher layers through the data link control Layer and is passed into the physical control Layer. It then passes into the communication links through some type of interface. SDLC supports a variety of link types and topologies. It can be used with point-to-point and multipoint links, bounded and unbounded media, half-duplex and full-duplex transmission facilities, and circuit-switched and packet-switched networks.

SDLC identifies two types of network nodes: primary and secondary. Primary nodes control the operation of other stations, called secondaries. The primary polls the secondaries in a predetermined order, and secondaries can then transmit if they have outgoing data. The primary also sets up and tears down links and manages the link while it is operational. Secondary nodes are controlled by a primary, which means that secondaries can send information to the primary only if the primary grants permission.

SDLC primaries and secondaries can be connected in four basic configurations:

- Point-to-point—Involves only two nodes, one primary and one secondary.
- Multipoint—Involves one primary and multiple secondaries.
- Loop—Involves a loop topology, with the primary connected to the first and last secondaries. Intermediate secondaries pass messages through one another as they respond to the requests of the primary.
- Hub go-ahead—Involves an inbound and an outbound channel. The primary uses the outbound channel to communicate with the secondaries. The secondaries use the inbound channel to communicate with the primary. The inbound channel is daisy-chained back to the primary through each secondary.

SDLC has a few derivatives which are adopted in different environment:

- HDLC, an ISO protocol for the x.25 network
- LAPB, an ITU-T protocol used in the ISDN network
- LAPF, an ITU-T protocol used in the Frame Relay network
- IEEE 802.2, often referred to as LLC, has three types and is used in the local area network
- QLLC, used to transport SNA data across X.25 networks

Protocol Structure

1 byte	1-2 bytes	1-2 bytes	variable	2 bytes	1 byte
Flag	Address field	Control field	Data	FCS	Flag

- Flag—Initiates and terminates error checking.
- Address—Contains the SDLC address of the secondary station, which indicates whether the frame comes from the primary or secondary.
- Control—Employs three different formats, depending on the type of SDLC frame used:
  - Information (I) frame—Carries upper-layer information and some control information.
  - Supervisory (S) frame—Provides control information. An S frame can request and suspend transmission, report on status, and acknowledge receipt of I frames. S frames do not have an information field.
  - Unnumbered (U) frame—Supports control purposes and is not sequenced. A U frame can be used to initialize secondaries. Depending on the function of the U frame, its control field is 1 or 2 bytes. Some U frames have an information field.
- Data—Contains a path information unit (PIU) or exchange identification (XID) information.
- Frame check sequence (FCS)—Precedes the ending flag delimiter and is usually a cyclic redundancy check (CRC) calculation remainder.

Related protocols

LAPB, X.25, Frame Relay, HDLC, LAPF, QLLC, LLC

Sponsor Source

SDLC is defined by IBM.

Reference

http://www.cisco.com/univercd/cc/td/doc/cisintwk/ito_doc/sdlcetc.htm
Synchronous Data Link Control and Derivatives

AppleTalk: Apple Computer Protocols Suite

Description

AppleTalk is a multi-layered protocol of Apple Computers providing internetwork routing, transaction and data stream service, naming service and comprehensive file and print sharing among Apple systems using the LocalTalk interface built into the Apple hardware. AppleTalk ports to other network media such as Ethernet by the use of LocalTalk to Ethernet bridges or by Ethernet add-in boards for Apple machines. Many third-party applications exist for the AppleTalk protocols.

An AppleTalk network can support up to 32 devices and data can be exchanged at a speed of 230.4 kilobits per second (Kbps). Devices can be as much as 1,000 feet apart. At the physical level, AppleTalk is a network with a bus topology that uses a trunk cable between connection modules.

The LocalTalk Link Access Protocol (LLAP) must be common to all systems on the network bus and handles the node-to-node delivery of data between devices connected to a single AppleTalk network. Data link layer interfaces to Ethernet, Token ring and FDDI are defined.

The Datagram Delivery Protocol (DDP) is the AppleTalk protocol implemented at the network layer. DDP is a connectionless datagram protocol providing best-effort delivery, which is similar to IP in the TCP/IP suite.

At the Transport Layer, several protocols exist to add different types of functionality to the underlying services. The Routing Table Maintenance Protocol (RTMP) allows bridges and internet routers to dynamically discover routes to the different AppleTalk networks in an internet. The AppleTalk Transaction Protocol (ATP) is responsible for controlling the transactions between requestor and responder sockets.

The Name Binding Protocol (NBP) is for the translation of a character string name into the internet address of the corresponding client. The AppleTalk Echo Protocol (AEP) allows a node to send data to any other node on an AppleTalk internet and receive an echoed copy of that data in return. The AppleTalk Data Stream Protocol (ADSP) is designed to provide byte-stream data transmission in a full duplex mode between any two sockets on an AppleTalk internet. The Zone Information Protocol (ZIP) is used to maintain an internet-wide mapping of networks to zone names.

In the Session Layer, the AppleTalk Session Protocol (ASP) is designed to interact with AppleTalk Transaction Protocol (ATP) to provide for establishing, maintaining and closing sessions.

The AppleTalk Filing Protocol (AFP) is an application or presentation layer protocol designed to control access to remote file systems. A key application using this protocol is the AppleShare for file sharing among a variety of user computers.

Architecture

AppleTalk protocols in the OSI layers:

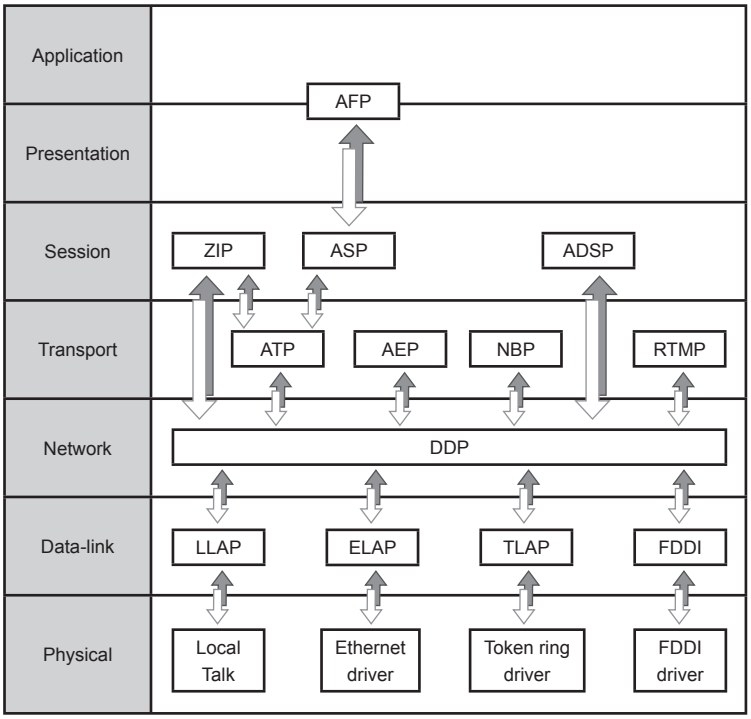


Figure 2-26: AppleTalk Protocol Stack Architecture

Application	AppleShare: for platform sharing of a variety of user computers
Presentation	AFP: AppleTalk Filing Protocol
Session	ADSP: AppleTalk Data Stream Protocol
	ASP: AppleTalk Session Protocol
	PAP: Printer Access Protocol
	ZIP: Zone Information Protocol
Transport	AEP: AppleTalk Echo Protocol
	ATP: AppleTalk Transaction Protocol
	NBP: Name Binding Protocol
	RTMP: Routing Table Maintenance Protocol
Network	DDP: Datagram Delivery Protocol
Data Link	AARP: AppleTalk Address Resolution Protocol
	LLAP: LocalTalk Link Access Protocol
	EtherTalk: AppleTalk Ethernet interface
	TokenTalk: AppleTalk Token Ring interface

Related protocols

Ethernet, Token Ring, FDDI

Sponsor Source

AppleTalk protocols are defined by Apple Computers.

Reference

http://developer.apple.com/macos/opentransport/docs/dev/Inside_AppleTalk.pdf
Inside AppleTalk

DECnet and Protocols

Description

DECnet is a protocol suite developed and supported by Digital Equipment Corporation (Digital or DEC, now part of HP). Several versions of DECnet have been released. The original DECnet allowed two directly attached minicomputers to communicate. Subsequent releases expanded the DECnet functionality by adding support for additional proprietary and standard protocols. Currently, two versions of DECnet are in wide use: DECnet Phase IV and DECnet plus (DECnet V). The DECnet now is part of the HP OpenVMS.

DECnet is developed under the framework of the Digital Network Architecture (DNA), which is a comprehensive layered network architecture that supports a large set of proprietary and standard protocols.

The DECnet Phase IV DNA is similar to the OSI architecture, which utilizes a seven layered approach. However, the Phase IV DNA is comprised of eight layers. The DECnet Phase IV DNA specifies four upper layers to provide user interaction services, network-management capabilities, file transfer, and session management. Specifically, these are referred to as the user layer, network management layer, network application layer, and session control layer.

The DECnet phase V (or DECnet Plus or DECnet/OSI) defines a layered model that implements three protocol suites: OSI, DECnet, and TCP/IP. DECnet plus conforms to the seven-layer OSI reference model and supports many of the standard OSI protocols. DECnet plus provides backward compatibility with DECnet Phase IV and supports multiple proprietary Digital protocols. DECnet plus supports functionality in the application, presentation, and session layers. The TCP/IP implementation of DECnet plus supports the lower-layer TCP/IP protocols and enables the transmission of DECnet traffic over TCP transport protocols.

Key Protocols

DECnet DNA phase IV and V in the OSI model and comparison with the TCP/IP suite:

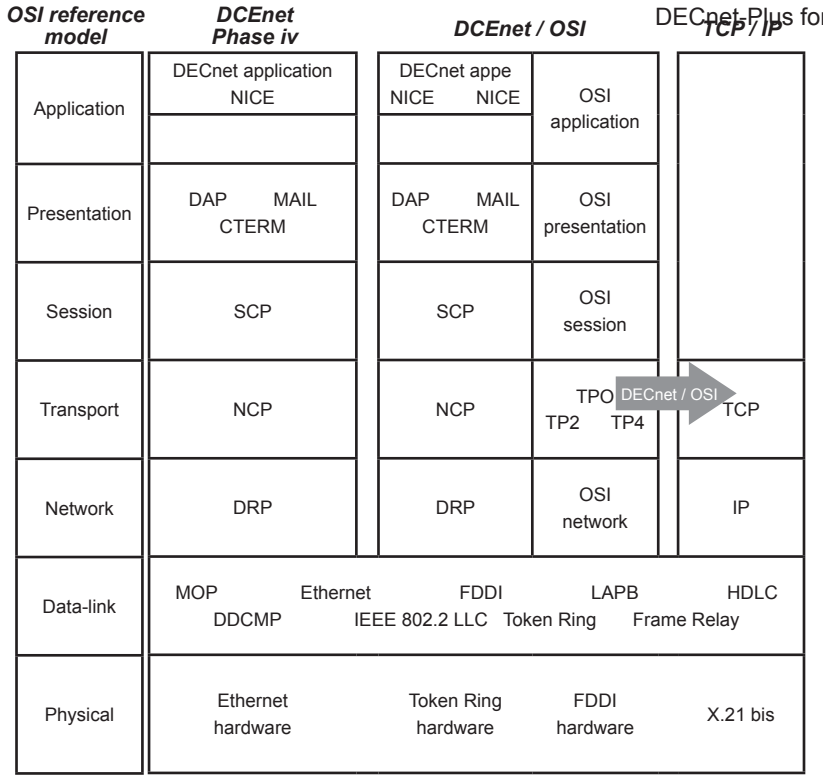


Figure 2-27: DECnet Protocol Suite Architecture

The key protocols in DECnet protocol suite:

Application	NICE: Network Information and Control Exchange protocol
Presentation	DAP: Data Access Protocol
	CTERM: Command Terminal
Session	SCP: Session Control Protocol
Transport	NSP: Network Service Protocol
Network	DRP: DECnet Routing Protocol
Data Link	MOP: Maintenance Operation Protocol
	DDCMP: Digital Data Communications Message Protocol

Related protocols

Ethernet, Token Ring, FDDI, TCP, IP, ISO-TP, Frame Relay, LAPB, HDLC, IEEE 802.2

Sponsor Source

DECnet protocols are defined by Digital Equipment Corporation (now part of HP).

Reference

http://ftp.digital.com/pub/DEC/DECnet/PhaseIV/
DECnet Phase IV Specifications
http://h71000.www7.hp.com/DOC/73final/6501/6501pro.HTML
DECnet Plus for OpenVMS Introduction and User's Guide

SS7 / C7 Protocols: Signaling System # 7 for Telephony

Protocol Description

Signalling System #7 (SS7) is a telecommunications protocol suite defined by the ITU-T which is used by the telephone companies for interoffice signalling. SS7 uses out of band or common-channel signalling (CCS) techniques, which uses a separated packet-switched network for the signalling purpose. SS7 is known as C7 outside North America.

The primary function of SS7 / C7 is to provide call control, remote network management, and maintenance capabilities for the inter- office telephone network. SS7 performs these functions by exchanging control messages between SS7 telephone exchanges (signalling points or SPs) and SS7 signalling transfer points (STPs). Basically, the SS7 control network tells the switching office which paths to establish over the circuit-switched network. The STPs route SS7 control packets across the signalling network. A switching office may or may not be an STP.

The SS7 / C7 network and protocol are used for providing intelligent network services such as:

- basic call setup, management, and tear down
- wireless services such as personal communications services (PCS), wireless roaming, and mobile subscriber authentication
- local number portability (LNP)
- toll-free (800/888) and toll (900) wireline services
- 911, 411 services
- enhanced call features such as call forwarding, caller ID display, and three-way calling
- efficient and secure worldwide telecommunications

The current SS7 / C7 network, one of the largest data network in the world, connects together local telcos, cellular, and long-distance networks nationwide and worldwide.

Protocol Structure

The SS7 / C7 protocol suite covers all 7 layers of the OSI model as shown in the following diagram:

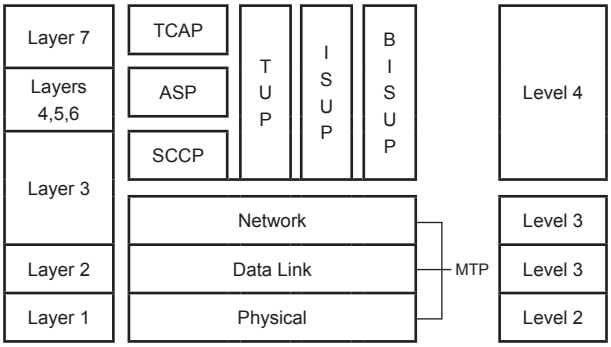


Figure 2-28: SS7/C7 Protocol Suite Architecture

ASP	Application Service Part	ASP provides the functions of Layers 4 through 6 of the OSI model.
BICC	Bearer Independent Call Control protocol	BICC is a call control protocol based on ISUP used between serving nodes to support the ISDN services independent of the bearer technology and signalling message transport technology used.
BISUP	B-ISDN User Part	BISUP is an ATM protocol intended to support services such as high-definition television (HDTV), multilingual TV, voice and image storage and retrieval, video conferencing, high-speed LANs and multimedia.
DUP	Data User Part	DUP defines the necessary call control, and facility registration and cancellation related elements for international common channel signalling by use of SS7 for circuit-switched data transmission services.
ISUP	ISDN User Part	ISUP supports basic telephone call connect/disconnect between end offices. ISUP was derived from TUP, but supports ISDN and intelligent networking functions. ISUP also links the cellular and PCS network to the PSTN.
MAP	Mobile Application Part	MAP is used to share cellular subscriber information among different networks.
MTP	Message Transfer Part	MTP crosses physical, data link and network layers. It defines what interface to be used, provides the network with sequenced delivery of all SS7 message packets; and provides routing, message discrimination and message distribution functions.
SCCP	Signalling Connection Control Part	SCCP provides end-to-end routing. SCCP is required for routing TCAP messages to their proper database.
TCAP	Transaction Capabilities Application Part	TCAP facilitates connection to an external database
TUP	Telephone User Part	TUP is an analog protocol that performs basic telephone call connect and disconnect.

Related protocols

ASP, BICC, BISUP, DUP, ISUP, MTP, SCCP, TCAP, TUP, MAP

Sponsor Source

SS7 / C7 protocols are defined by ITU-T in Q.700 documents series.

Reference

http://www.cisco.com/univercd/cc/td/doc/product/tel_pswt/vco_prod/ss7_fund/
SS7 Fundamentals
http://www.itu.int/rec/recommendation.asp?type=folders&lang=e&parent=T-REC-Q.700
Introduction to Signaling System No. 7

Protocol Name

BISUP: Broadband ISDN User Part

Protocol Description

Broadband ISDN User Part (BISUP) is a protocol intended to support services such as high-definition television (HDTV), multilingual TV, voice and image storage and retrieval, video conferencing, high-speed LANs and multimedia. Since BISDN is not deployed widely so far, so is not the BISUP.

Protocol Structure

The structure of the B-ISUP protocol is as follows:

Octets	1	2	3	4
1	Message Type	Length Indicator		Transit at intermed exch. ind
2				Release call ind
3				Send notification ind
4				Discard message ind
5				Pass on not possible ind
6				Broadband/narrow-band interworking ind
7				
8				Ext.

- Message Type - The different message types. The following message types are available:
- Message Length - The message length in octets.
- Broadband/narrow-band Iinterworking Indicator for passing on, discard mesage, release call, etc.
- Pass on not Possible Indicator for release call and discard information
- Discard Message Indicator for dicard or do not discard message
- Send Notification Indicator for sending or do no sending notification
- Release call indicator for release or do not release call
- Transit at intermed exchance Indicator for transit interpretation or end node interpretation

Related protocols

SS7, ASP, BICC, DAP, ISUP, MTP, SCCP, TCAP, TUP, MAP

Sponsor Source

BISUP is defined by ITU-T Q.2762 and Q.2763.

Reference

http://www.itu.int/rec/recommendation.asp?type=folders&lang=e&parent=T-REC-Q.2762
General Functions of messages and signals of the B-ISUP of Signalling System No. 7

Protocol Name

DUP: Data User Part

Protocol Description

Data User Part (DUP), an application protocol in the SS7/C7 protocol suite, defines the necessary call control and facility registration and cancellation related elements for circuit-switched data transmission services. The data signalling messages are divided into two categories:

- Call and circuit related messages: used to set up and clear a call or control and supervise the circuit state.
- Facility registration and cancellation related messages: used to exchange information between originating and destination exchanges to register and cancel information related to user facilities.

While Data User Part (DUP) is still in use currently, but it is fallen out of favor except in certain parts of the world such as China.

Protocol Structure

The pouting label of DUP contains the DPS, OPC, BIC and TSC fields. It is contained in a signalling message and used to identify particulars to which the message refers. This is also use by the message transfer part to route the message towards its destination point.

The general format of the header of call and circuit related messages is shown as follows:

OPC	DPS
BIC	OPC
TCS	BIC
Message specific parameters	Heading Code

The general format of the header of facility registration and cancellation messages is shown as follows:

OPC	DPS
Spare bits	OPC
Message specific parameters	Heading code

OPC - The originating point code (14bits) is the code applicable to the data switching exchange from which the message is sent.

DPS - The destination point code (14bits) is the code applicable to the data switching exchange to which the message is to be delivered.

BIC - Bearer identification code (12 bits).

TSC - Time slot code (8 bits). If the data circuit is derived from the data multiplex carried by the bearer, identified by the bearer identification code:

Heading code - The heading code (4 bits) contains the message type code which is mandatory for all messages. It uniquely defines the function and format of each DAP message.

Message specific parameters - Contains specific fields for each message.

Spare bits - Not used, should be set to “0000”.

Related protocols

SS7, ASP, BICC, BISUP, ISUP, MTP, SCCP, TCAP, TUP, MAP

Sponsor Source

DUP is defined by ITU-T Q.741 (or X.61).

Reference

http://www.itu.int/rec/recommendation.asp?type=folders&lang=e&parent=T-REC-Q.741: SS7 Data User Part

Protocol Name

ISUP: ISDN User Part

Protocol Description

The ISDN User Part (ISUP), a key protocol in the SS7 / C7 signalling system, defines the protocol and procedures used to set-up, manage, and release trunk circuits that carry voice and data calls over the public switched telephone network (PSTN) between different switches. ISUP is used for both ISDN and non-ISDN calls. A simple call flow using ISUP signaling is as follows:

Call set up: When a call is placed to an out-of-switch number, the originating SSP transmits an ISUP initial address message (IAM) to reserve an idle trunk circuit from the originating switch to the destination switch. The destination switch rings the called party line if the line is available and transmits an ISUP address complete message (ACM) to the originating switch to indicate that the remote end of the trunk circuit has been reserved. The STP routes the ACM to the originating switch which rings the calling party's line and connects it to the trunk to complete the voice circuit from the calling party to the called party.

Call connection: When the called party picks up the phone, the destination switch terminates the ringing tone and transmits an ISUP answer message (ANM) to the originating switch via its home STP. The STP routes the ANM to the originating switch which verifies that the calling party's line is connected to the reserved trunk and, if so, initiates billing.

Call tear down: If the calling party hangs-up first, the originating switch sends an ISUP release message (REL) to release the trunk circuit between the switches. The STP routes the REL to the destination switch. If the called party hangs up first, or if the line is busy, the destination switch sends an REL to the originating switch indicating the release cause (e.g., normal release or busy). Upon receiving the REL, the destination switch disconnects the trunk from the called party's line, sets the trunk state to idle, and transmits an ISUP release complete message (RLC) to the originating switch to acknowledge the release of the remote end of the trunk circuit. When the originating switch receives (or generates) the RLC, it terminates the billing cycle and sets the trunk state to idle in preparation for the next call.

Protocol Structure

The ANSI and ITU-T have slightly different ISUP format. ITU-T ISUP message format:

Routing label (5bytes)
Circuit identification code (2 bytes)
Message type code (1 byte)
Parameters – varies according to message type values

- Routing label - The routing label is used by the relevant user part to identify particulars to which the message refers. It is also used by the Message Transfer Part (MTP) to route the message towards its destination point.
- Circuit identification code - The allocation of circuit identification codes to individual circuits is determined by bilateral agreement and/or in accordance with applicable predetermined rules.
- Message type code - The message type code uniquely defines the function and format of each ISDN User Part message. Each message consists of a number of parameters. Message types may be:
  - Address complete
  - Answer
  - Blocking
  - Blocking acknowledgement
  - Call progress
  - Circuit group blocking
  - Circuit group blocking acknowledgement
  - Circuit group query
  - Circuit group query response
  - Circuit group reset
  - Circuit group reset acknowledgement
  - Circuit group unblocking
  - Circuit group unblocking acknowledgement
  - Charge information
  - Confusion
  - Connect
  - Continuity
  - Continuity check request
  - Facility
  - Facility accepted
  - Facility reject
  - Forward transfer
  - Identification request
  - Identification response
  - Information
  - Information request
  - Initial address
  - Loop back acknowledgement
  - Network resource management
  - Overload
  - Pass-along
  - Release
  - Release complete
  - Reset circuit
  - Resume
  - Segmentation
  - Subsequent address
  - Suspend
  - Unblocking
  - Unblocking acknowledgement
  - Unequipped CIC

User Part available
User Part test
User-to-user information

- Parameters - Each parameter has a name which is coded as a single octet. The length of a parameter may be fixed or variable, and a length indicator for each parameter may be included.

Related protocols

SS7, ASP, BICC, BISUP, DUP, MTP, SCCP, TCAP, TUP, MAP

Sponsor Source

ISUP is defined by ITU-T Q.763 documents.

Reference

http://www.itu.int/rec/recommendation.asp?type=folders&lang=e&parent=T-REC-Q.763
SS7 ISDN User Part

Protocol Name

MAP: Mobile Application Part

Protocol Description

The Mobile Application Part (MAP), one of the protocols in the SS7 suite, allows for the implementation of the mobile network (GSM) signaling infrastructure. The premise behind MAP is to connect the distributed switching elements, called mobile switching centers (MSCs) with a master database called the Home Location Register (HLR). The HLR dynamically stores the current location and profile of a mobile network subscriber. The HLR is consulted during the processing of an incoming call. Conversely, the HLR is updated as the subscriber moves about the network and is thus serviced by different switches within the network.

MAP has been evolving as wireless networks grow, from supporting strictly voice, to supporting packet data services as well. The fact that MAP is used to connect NexGen elements such as the Gateway GPRS Support node (GGSN) and Serving Gateway Support Node (SGSN) is a testament to the sound design of the GSM signaling system.

MAP has several basic functions:

- Mechanism for a Gateway-MSC (GMSC) to obtain a routing number for an incoming call
- Mechanism for an MSC via integrated Visitor Location Register (VLR) to update subscriber status and routing number.
- Subscriber CAMEL trigger data to switching elements via the VLR
- Subscriber supplementary service profile and data to switching elements via the VLR

Protocol Structure

8	16bit
Operation specifier	Length
MAP Parameters	

- Length - The length of the packet.
- MAP parameters - Various parameters depending on the operation.
- Operation specifier - The type of packet. The following operations are defined:
  - AuthenticationDirective
  - AuthenticationDirectiveForward
  - AuthenticationFailureReport
  - AuthenticationRequest
  - AuthenticationStatusReport
  - BaseStationChallenge
  - Blocking
  - BulkDeregistration
  - CountRequest
  - FacilitiesDirective
  - FacilitiesDirective2
  - FacilitiesRelease
  - FeatureRequest
  - FlashRequest
  - HandoffBack
  - HandoffBack2
  - HandoffMeasurementRequest
  - HandoffMeasurementRequest2
  - HandoffToThird
  - HandoffToThird2
  - InformationDirective
  - InformationForward
  - InterSystemAnswer
  - InterSystemPage
  - InterSystemPage2
  - InterSystemSetup
  - LocationRequest
  - MobileOnChannel
  - MSInactive
  - OriginationRequest
  - QualificationDirective
  - QualificationRequest
  - RandomVariableRequest
  - RedirectionDirective
  - RedirectionRequest
  - RegistrationCancellation
  - RegistrationNotification
  - RemoteUserInteractionDirective
  - ResetCircuit
  - RoutingRequest
  - SMSDeliveryBackward
  - SMSDeliveryForward
  - SMSDeliveryPointToPoint
  - SMSNotification
  - SMSRequest
  - TransferToNumberRequest
  - TrunkTest
  - TrunkTestDisconnect
  - Unblocking
  - UnreliableRoamerDataDirective
  - UnsolicitedResponse

Related protocols

SS7, ASP, BICC, BISUP, DUP, MTP, SCCP, TCAP, TUP

Sponsor Source

MAP is defined by ITU-T as part of SS7 protocols.

Reference

http://www.cisco.com/univercd/cc/td/doc/product/tel_pswt/vco_

prod/ss7_fund/
SS7 Fundamentals
http://www.itu.int/rec/recommendation.asp?type=folders&lang=e&parent=T-REC-Q.700
Introduction to Signaling System No. 7

Protocol Name

MTP2 and MTP3: Message Transfer Part level 2 and level 3

Protocol Description

Message Transfer Part (MTP), a protocol in the SS7/C7 protocol suite, transfers signal messages and performs associated functions, such as error control and signaling link security. Message Transfer Part (MTP) also provides reliable routing within a network. MTP has two parts, MTP level 2 (MTP2) and level 3 (MTP3), that performs functions at the layers 2 and 3 respectively of the OSI 7 layers model.

Message Transfer Part Level 2 (MTP2) resides at Layer 2 in the SS7 protocol stack. It is responsible for the reliable transmission of signalling units over an individual Signalling Link. MTP2 reliability is achieved through retransmission techniques.

Message Transfer Part level 3 (MTP3) is the network layer in the SS7 protocol stack. It routes SS7 signalling messages to public network nodes by means of Destination Point Codes, and to the appropriate signalling entity within a node by means of a Service Info Octet. MTP3 is specified as part of the SS7 protocol and is also referred to as part of the B-ICI interface for ATM. MTP3 sits between MTP2 and the user parts (ISUP, TUP, SCCP and TCAP) of the SS7 protocol stack. B-ISUP is an Application Layer protocol run over MTP3.

MTP3 is split into two distinct parts, SMH (Signalling Message Handling) and SNM(Signalling Network Management). The SNM part looks after the general management of MTP, the SHM part deals with the discrimination, distribution and routing of signalling messages. MTP3 defines the functions and procedures of the signalling system for signalling message handling and signalling network management. Signalling message handling consists of the actual transfer of a signalling message and directing the message to the proper signalling link or user part. Signalling network management consists of controlling the signalling message routing and configuration of the signalling network facilities based on predetermined information and the status of the signalling network facilities.

MTP3 provides a connectionless message transfer system for passing information across a network. MTP3 includes a number of link-protection features, to allow automatic rerouting of signalling messages around broken signalling transfer points. It includes certain management functions for congestion control on signalling links.

MTP2 User Adaptation Layer (M2UA) is used to access MTP2 functions using SCTP (Streaming Control Transmission Protocol). MTP3 User Adaptation Layer (M3UA) is a protocol for supporting the transport of any SS7 MTP3-User signaling (e.g., ISUP, SCCP and TUP messages) over the IP Network.

Protocol Structure

The format of the header of MTP2 is shown as follows:

7	8bit
Flag	
BSN (7 bits)	BIB
FSN (7 bits)	FIB
LI (6 + 2 bits)	
SIO	
SIF	
Checksum (16 bits)	
Flag	

- BSN - Backward sequence number. Used to acknowledge message signal units which have been received from the remote end of the signalling link.
- BIB - Backward indicator bit. The forward and backward indicator bit together with the forward and backward sequence numbers are used in the basic error control method to perform the signal unit sequence control and acknowledgment functions.
- FSN - Forward sequence number.
- FIB - Forward indicator bit.
- LI - Length indicator. This indicates the number of octets following the length indicator octet.
- SIO - Service information octet.
- SIF - Signalling information field.
- Checksum - Every signal unit has 16 check bits for error detection.

The structure of the MTP-3 header is shown as follows:

4	8bit
Service indicator	Subservice field

- Service indicator - Used to perform message distribution and in some cases to perform message routing. The service indicator codes are used in international signalling networks for the following purposes:
  - Signalling network management messages
  - Signalling network testing and maintenance messages
  - SCCP
  - Telephone user part
  - ISDN user part
  - Data user part
  - Reserved for MTP testing user part.
- Sub-service field - The sub-service field contains the network indicator and two spare bits to discriminate between national and international messages.

Related protocols

SS7, ASP, BICC, BISUP, DUP, ISUP, SCCP, TCAP, TUP

Sponsor Source

MTP level 2 and level 3 protocols are defined by ITU-T documents Q.703 and Q.704.

Reference

http://www.itu.int/rec/recommendation.asp?type=products&parent=T-REC-q
ITU-T Q documents.
http://www.javvin.com/protocol/rfp3331.pdf
Signaling System 7 (SS7) Message Transfer Part 2 (MTP2) - User Adaptation Layer (M2UA)
http://www.javvin.com/protocol/rfp3332.pdf
Signaling System 7 (SS7) Message Transfer Part 3 (MTP3) - User Adaptation Layer (M3UA)

Protocol Name

SCCP: Signalling Connection Control Part of SS7

Protocol Description

Signaling Connection Control Part (SCCP), a routing protocol in the SS7 protocol suite (in layer 4), provides end-to-end routing for TCAP messages to their proper databases.

SCCP provides connectionless and connection-oriented network services above MTP Level 3. While MTP Level 3 provides point codes to allow messages to be addressed to specific signaling points, SCCP provides subsystem numbers to allow messages to be addressed to specific applications or subsystems at these signaling points. SCCP is used as the transport layer for TCAP-based services such as free phone (800/888), calling card, local number portability, wireless roaming, and personal communications services (PCS).

SCCP also provides the means by which an STP can perform global title translation (GTT), a procedure by which the destination signaling point and subsystem number (SSN) is determined from digits (i.e., the global title) present in the signaling message. The global title digits may be any sequence of digits, such as 800/888 number, pertinent to the service requested.

Protocol Structure

SCCP messages are contained within the Signaling Information Field (SIF) of an MSU. There are two formats for the SCCP messages. One is defined by ANSI and the other is defined by ITU-T.

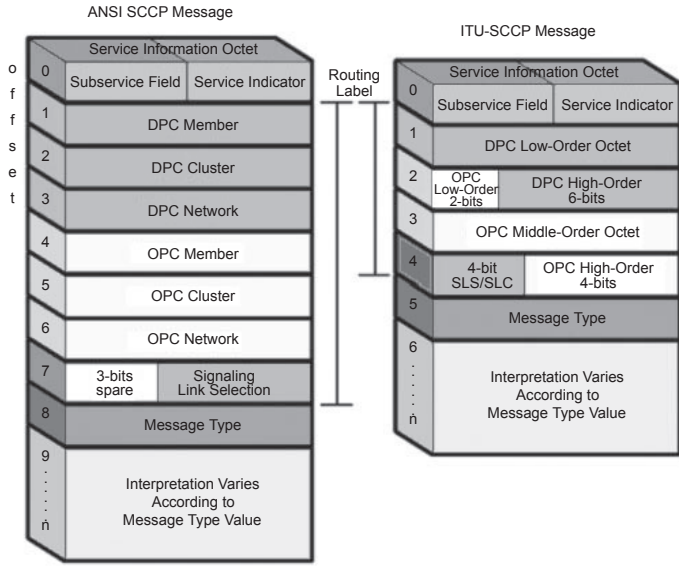


Figure 2-29: SCCP Protocol Structure

The SIF contains the routing label followed by the SCCP message header with the following structure:

Routing label
Message type
Mandatory fixed part
Mandatory variable part
Optional part

- Routing label - A standard routing label – see the picture regarding the ANSI and ITU SCCP message for more information.
- Message type code - A one octet code which is mandatory for all messages. The message type code uniquely defines the function and format of each SCCP message.
- Mandatory fixed part - The parts that are mandatory and of fixed length for a particular message type will be contained in the mandatory fixed part.
- Mandatory variable part - Mandatory parameters of variable length will be included in the mandatory variable part. The name of each parameter and the order in which the pointers are sent is implicit in the message type.
- Optional part - The optional part consists of parameters that may or may not occur in any particular message type. Both fixed length and variable length parameters may be included. Optional parameters may be transmitted in any order. Each optional parameter will include the parameter name (one octet) and the length indicator (one octet) followed by the parameter contents.

Related protocols

SS7/C7, ASP, BICC, BISUP, DUP, ISUP, SCCP, TCAP, TUP

Sponsor Source

SCCP is defined by ITU-T documents Q.713.

Reference

http://www.itu.int/itudoc/itu-t/rec/q/q500-999/q713_23786.html
Q.713: SCCP Specification

Protocol Name

TCAP: Transaction Capabilities Application Part

Protocol Description

Transaction Capabilities Application Part (TCAP), a protocol in the SS7 protocol suite, enables the deployment of advanced intelligent network services by supporting non-circuit related information exchange between signaling points using the Signalling Connection Control Part (SCCP) connectionless service. TCAP also supports remote control—ability to invoke features in another remote network switch.

An SSP uses TCAP to query an SCP to determine the routing number(s) associated with a dialed 800, 888, or 900 number. The SCP uses TCAP to return a response containing the routing number(s) (or an error or reject component) back to the SSP. Calling card calls are also validated using TCAP query and response messages. When a mobile subscriber roams into a new mobile switching center (MSC) area, the integrated visitor location register requests service profile information from the subscriber's home location register (HLR) using mobile application part (MAP) information carried within TCAP messages.

TCAP messages are contained within the SCCP portion of an MSU. A TCAP message is comprised of a transaction portion and a component portion.

Protocol Structure

The TCAP transaction portion contains the package type identifier with the following package types:

- Unidirectional: Transfers component(s) in one direction only (no reply expected).
- Query with Permission: Initiates a TCAP transaction (e.g., a 1-800 query). The destination node may end the transaction.
- Query without Permission: Initiates a TCAP transaction. The destination node may not end the transaction.
- Response: Ends the TCAP transaction. A response to a 1-800 query with permission may contain the routing number(s) associated with the 800 number.
- Conversation with Permission: Continues a TCAP transaction. The destination node may end the transaction.
- Conversation without Permission: Continues a TCAP transaction. The destination node may not end the transaction.
- Abort: Terminates a transaction due to an abnormal situation.

The transaction portion also contains the Originating Transaction ID and Responding Transaction ID fields, which associate the TCAP transaction with a specific application at the originating and destination signaling points respectively.

The TCAP component portion contains components as follows:

- Invoke (Last): Invokes an operation. For example, a Query with Permission transaction may include an Invoke (Last) component to request SCP translation of a dialed 800 number. The component is the "last" component in the query.
- Invoke (Not Last): Similar to the Invoke (Last) component except that the component is followed by one or more components.
- Return Result (Last): Returns the result of an invoked operation. The component is the "last" component in the response.
- Return Result (Not Last): Similar to the Return Result (Last) component except that the component is followed by one or more components.
- Return Error: Reports the unsuccessful completion of an invoked operation.
- Reject: Indicates that an incorrect package type or component was received.

Components include parameters which contain application-specific data carried unexamined by TCAP.

The TCAP header structure:

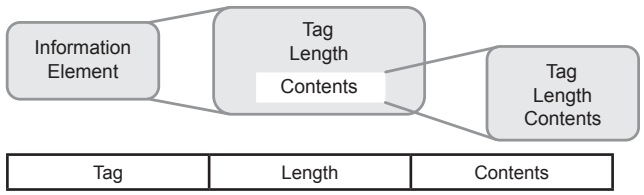


Figure 2-30: TCAP Protocol Structure

- Information Element - An information element is first interpreted according to its position within the syntax of the message. Each information element within a TCAP message has the same structure. An information element consists of three fields: Tag, Length and Contents.
- Tag - The Tag distinguishes one information element from another and governs the interpretation of the Contents. It may be one or more octets in length. The Tag is composed of Class, Form and Tag codes.
- Length - Specifies the length of the Contents.
- Contents - Contains the substance of the element, containing the primary information the element is intended to convey.

Related protocols

SS7/C7, ASP, BICC, BISUP, DUP, ISUP, SCCP, TCAP, TUP

Sponsor Source

TCAP is defined by ITU-T documents Q.773.

Reference

http://www.itu.int/itudoc/itu-t/rec/q/q500-999/q773_24880.html
Q.773: TCAP Specification

Protocol Name

TUP: Telephone User Part

Protocol Description

The Telephone User Part (TUP) provides the signaling backbone between switching elements for basic call establishment, supervision, and release of circuit switched network connections for telecommunications services. TUP supports analog and digital circuits, and limited call management signaling.

TUP controls the circuits used to carry voice traffic. Also using TUP, the state of circuits can be verified and managed. TUP is good in supporting applications such as switching and voice mail in which calls are routed between endpoints in either fixed or wireless networks.

While Telephone User Part (TUP) is still in use currently, but it is fallen out of favor except certain parts of the world such as China. TUP is replaced by ISUP which adds support for data, advanced ISDN and Intelligent Network.

Protocol Structure

The TUP header structure is as follows:

8	7	6	5	4	3	2	1	Octets
Message Type Code								1

Message Type Code - It basically contains the label, the heading code and one or more signals and/or indications. The service information octet comprises the service indicator and the subservice field. The service indicator is used to associate signalling information with a particular User Part and is only used with message signal units. The information in the subservice field permits a distinction to be made between national and international signalling messages. In national applications when this discrimination is not required possibly for certain national User Parts only, the subservice field can be used independently for different User Parts.

Related protocols

SS7, ASP, BICC, BISUP, ISUP, MTP, SCCP, TCAP, DUP, MAP

Sponsor Source

TUP is defined by the ITU-T Q.723 and ISUP is defined in ITU-T Q.763.

Reference

http://www.itu.int/rec/recommendation.asp?type=folders&lang=e&parent=T-REC-Q.723
SS7 Telephone User Part

http://www.itu.int/rec/recommendation.asp?type=folders&lang=e&parent=T-REC-Q.763
SS7 ISDN User Part

Other Protocols

Protocol Name

Microsoft CIFS: Common Internet File System

Protocol Description

The Common Internet File System (CIFS), an enhanced version of Microsoft Server Message Block (SMB), is the standard way that computer users share files across intranets and the Internet. CIFS enables collaboration on the Internet by defining a remote file-access protocol that is compatible with the way applications already share data on local disks and network file servers. CIFS runs over TCP/IP, utilizes the Internet's global Domain Naming Service (DNS) for scalability and is optimized to support slower speed dial-up connections common on the Internet. CIFS can be sent over a network to remote devices using the redirector packages. The redirector also uses CIFS to make requests to the protocol stack of the local computer.

Key features that CIFS offers are:

- File Access with integrity: CIFS supports the usual set of file operations; open, close, read, write and seek. CIFS also supports file and record lock and unlocking. CIFS allows multiple clients to access and update the same file while preventing conflicts by providing file sharing and file locking.
- Optimization for Slow Links: The CIFS protocol has been tuned to run well over slow-speed dial-up lines. The effect is improved performance for users who access the Internet using a modem.
- Security: CIFS servers support both anonymous transfers and secure, authenticated access to named files. File and directory security policies are easy to administer.
- Performance and Scalability: CIFS servers are highly integrated with the operating system, and are tuned for maximum system performance. CIFS supports all Microsoft platforms after Windows 95. It also supports other popular operation systems such as UNIX, VMS, Macintosh, IBM LAN server etc.
- Unicode File Names: File names can be in any character set, not just character sets designed for English or Western European languages.
- Global File Names: Users do not have to mount remote file systems, but can refer to them directly with globally significant names, instead of ones that have only local significance.
- CIFS complements Hypertext Transfer Protocol (HTTP) while providing more sophisticated file sharing and file transfer than older protocols, such as FTP.

Protocol Structure

The CIFS and SMB defines many client and server type of commands and messages. The commands and messages can be broadly classified as follows:

- Connection establishment messages consist of commands that start and end a redirector connection to a shared resource at the server.
- Namespace and File Manipulation messages are used by the redirector to gain access to files at the server and to read and write them.
- Printer messages are used by the redirector to send data to a print queue at a server and to get status information about the print queue.
- Miscellaneous messages are used by the redirector to write to mailslots and named pipes.

The typical process and architecture of the CIFS message flow is shown as follows:

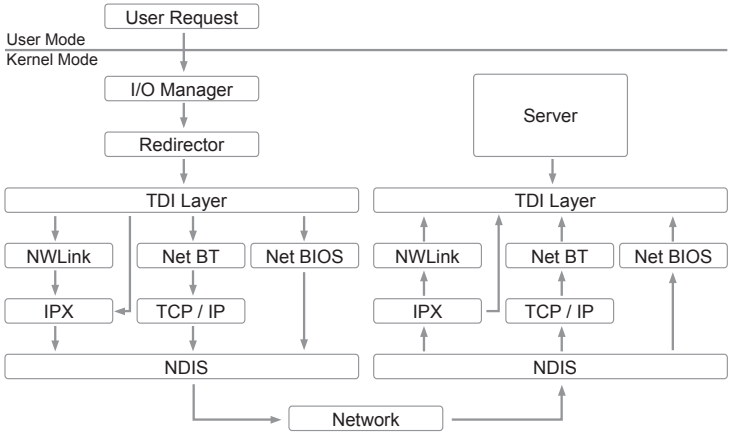


Figure 2-31: Microsoft CIFS Flow Chart

Related protocols

SMB, HTTP, FTP, DNS

Sponsor Source

CIFS is a Microsoft protocol.

Reference

http://www.microsoft.com/mind/1196/cifs.asp
CIFS: A Common Internet File System
http://www.snia.org/tech_activities/CIFS/CIFS-TR-1p00_FINAL.pdf
Common Internet file System (CIFS) Technical Reference

Protocol Name

Microsoft SOAP: Simple Object Access Protocol

Protocol Description

The Simple Object Access Protocol (SOAP) is a lightweight and simple XML-based protocol that is designed to exchange structured and typed information on the Web. SOAP can be used in combination with a variety of existing Internet protocols and formats including Hypertext Transfer Protocol (HTTP), Simple Mail Transfer Protocol (SMTP), and Multipurpose Internet Mail Extensions (MIME), and can support a wide range of applications from messaging systems to remote procedure calls (RPCs).

SOAP consists of three parts:

- The SOAP envelope construct defines an overall framework for expressing what is in a message; who should deal with it and whether it is optional or mandatory.
- The SOAP encoding rules define a serialization mechanism that can be used to exchange instances of application-defined data types.
- The SOAP RPC representation defines a convention that can be used to represent remote procedure calls and responses.

SOAP messages are basically one-way transmissions from a sender to a receiver, but SOAP messages are often combined to implement patterns such as request/response. All SOAP messages are encoded using XML. A SOAP message is an XML document that consists of a mandatory SOAP envelope, an optional SOAP header and a mandatory SOAP body.

Binding SOAP to HTTP provides the advantage of being able to use the formalism and decentralized flexibility of SOAP with the rich feature set of HTTP. Carrying SOAP in HTTP does not mean that SOAP overrides existing semantics of HTTP but rather that the semantics of SOAP over HTTP map naturally to HTTP semantics. In the case of using HTTP as the protocol binding, an RPC call maps naturally to an HTTP request and an RPC response maps to an HTTP response. However, using SOAP for RPC is not limited to the HTTP protocol binding.

Protocol Structure

SOAP message format:
SOAP header

```
<SOAP-ENV: Envelope
Attributes>
<SOAP-ENV:Body
Attributes
</SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

Example 1 SOAP Message Embedded in HTTP Request

```
POST /StockQuote HTTP/1.1
Host: www.stockquoteserver.com
Content-Type: text/xml; charset="utf-8"
Content-Length: nnnn
SOAPAction: "Some-URI"

<SOAP-ENV:Envelope
  xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/
envelope/"
   SOAP-ENV:encodingStyle="http://schemas.xmlsoap.
org/soap/encoding/">
  <SOAP-ENV:Body>
    <m:GetLastTradePrice xmlns:m="Some-URI">
      <symbol>DIS</symbol>
    </m:GetLastTradePrice>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

Following is the response message containing the HTTP message with the SOAP message as the payload:

Example 2 SOAP Message Embedded in HTTP Response

```
HTTP/1.1 200 OK
Content-Type: text/xml; charset="utf-8"
Content-Length: nnnn

<SOAP-ENV:Envelope
  xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/
envelope/"
   SOAP-ENV:encodingStyle="http://schemas.xmlsoap.
org/soap/encoding/"/>
  <SOAP-ENV:Body>
     <m:GetLastTradePriceResponse xmlns:m="Some-
URI">
       <Price>34.5</Price>
    </m:GetLastTradePriceResponse>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

Related protocols

HTTP, XML, RPC, MIME, STMP

Sponsor Source

The Simple Object Access Protocol (SOAP) is proposed by Microsoft.

Reference

http://www.w3.org/TR/2000/NOTE-SOAP-20000508/
Simple Object Access Protocol (SOAP)

Protocol Name

Xerox IDP: Internet Datagram Protocol

Protocol Description

Internet Datagram Protocol (IDP) is a simple, unreliable datagram protocol, which is used to support the SOCK_DGRAM abstraction for the Internet Protocol (IP) family. IDP sockets are connectionless and normally used with the sendto and recvfrom subroutines. The connect subroutine can also be used to fix the destination for future packets, in which case the recv or read subroutine and the send or write subroutine can be used.

Xerox protocols (XNS protocol suite) are built vertically on top of IDP. Thus, IDP address formats are identical to those used by the Sequenced Packet Protocol (SPP). The IDP port space is the same as the SPP port space; that is, an IDP port may be "connected" to an SPP port, with certain options enabled. In addition, broadcast packets may be sent (assuming the underlying network supports this) by using a reserved broadcast address. This address is network interface-dependent.

IDP has been adopted by various other manufacturers. The most popular variant is Novell's IPX.

Protocol Structure

Usage Conventions

The following example illustrates how IDP uses the SOCK_DGRAM mechanism:

```
#include <sys/socket.h>
#include <netns/ns.h>
#include <netns/idp.h>
s = socket(AF_NS, SOCK_DGRAM, 0);
```

Socket Options for IDP

SO_HEADERS_ON_INPUT	When set, the first 30 bytes of any data returned from a read or recvfrom subroutine are the initial 30 bytes of the IDP packet, described as follows:
	struct idp {
	u_short idp_sum;
	u_short idp_len;
	u_char idp_tc;
	u_char idp_pt;
	struct ns_addr idp_dna;
	struct ns_addr idp_sna;
	};
	This allows the user to determine both the packet type and whether the packet was a multicast packet or directed specifically at the local host. When requested by the getsockopt subroutine, the SO_HEADERS_ON_INPUT option gives the current state of the option: NSP_RAWIN or 0.

SO_HEADERS_ON_OUTPUT	When set, the first 30 bytes of any data sent are the initial 30 bytes of the IDP packet. This allows the user to determine both the packet type and whether the packet should be a multicast packet or directed specifically at the local host. You can also misrepresent the sender of the packet. When requested by the getsockopt subroutine, the SO_HEADERS_ON_OUTPUT option gives the current state of the option: NSP_RAWOUT or 0.
SO_DEFAULT_HEADERS	The user provides the kernel an IDP header, from which the kernel determines the packet type. When the SO_DEFAULT_HEADERS option is requested by the getsockopt subroutine, the kernel provides an IDP header, showing the default packet type and the local and foreign addresses, if connected.
SO_ALL_PACKETS	When set, this option disables automatic processing of both Error Protocol packets, and SPP packets.
SO_SEQNO	When requested by the getsockopt subroutine, the S0_SEQNO option returns a sequence number that is not likely to be repeated. It is useful in constructing Packet Exchange Protocol (PEP) packets.

Error Codes

The IDP protocol fails if one or more of the following are true:

EISCONN	The socket already has a connection established on it.
ENOBUFS	The system ran out of memory for an internal data structure.
ENOTCONN	The socket has not been connected or no destination address was specified when the datagram was sent.
EADDRINUSE	An attempt was made to create a socket with a port that has already been allocated.
EADDRNOTAVAIL	An attempt was made to create a socket with a network address for which no network interface exists.

Related protocols

IPX, XNS

Sponsor Source

IDP is defined by Xerox

Protocol Name

Toshiba FANP: Flow Attribute Notification Protocol

Protocol Description

Flow Attribute Notification Protocol(FANP) is a protocol between neighbor modes which manages cut-through packet forwarding functionalities. In cut-through packet forwarding, a router doesn't perform conventional IP packet processing for received packets. FANP indicates mapping information between a datalink connection and a packet flow to the neighbor node. It helps a pair of nodes manage mapping information. By using FANP, routers such as the CSR (Cell Switch Router) can forward incoming packets based on their datalink-level connection identifiers, bypassing usual IP packet processing. FANP has the following characteristics:

- Soft-state, cut-through path (Dedicated-VC) management
- Protocol between neighbor nodes instead of end-to-end
- Applicable to any connection-oriented, datalink platform.

FANP generally runs on ATM networks.

There are 7 FANP control messages. They are encapsulated into IP packets, apart from the PROPOSE message which uses an extended ATM ARP message format. The destination IP address in the IP packet header signifies the neighbor node's IP address. The source IP address is the sender's IP address. The IP protocol ID is 110.

The following message format exists for: Offer, Ready and Error messages. Propose Ack, Remove and Remove Ack messages do not have the flow ID field.

Protocol Structure

8	16	24	32bit
Version	OpCode	Checksum	
VCID type	Flow ID	Reserved/Refresh int./Error code	
VCID			
Flow ID			

- Version - The Version number. This version is version 1.
- OpCode – Operation code, the following OpCode values exist: 1 Propose Ack; 2 Offer; 3 Ready; 4 Error; 5 Remove; 6 Remove ACK.
- Checksum - A 16 bit checksum for the whole message.
- VCID type - The type of VCID. The current value is 1. The VCID uniquely identifies the datalink connection between neighbor nodes.
- Flow ID - If the Flow ID is 0, then the flow ID field is null. If the Flow ID is 1, then the Flow ID field described below is present.
- Reserved - In Offer messages the Refresh Timer field appears here. In error messages, the Error code field appears here.
- Refresh timer - The interval of the Refresh timer, in seconds. (Only appears in Offer messages.) The recommended value is 120.
- Error code - Only appears in Error Messages.
- VCID - Virtual Connection Identification.
- Flow ID - The Flow ID field does not appear in propose ACK, Remove and Remove Ack messages. When there is a flow ID type value of 1, this field contains the source and destination IP addresses of the flow.

Sponsor Source

FANP is a Toshiba protocol circulated by IETF (www.ietf.org) in RFC 2129.

Reference

http://www.javvin.com/protocol/rfc2129.pdf
Toshiba's Flow Attribute Notification Protocol (FANP) Specification

Network Protocols Dictionary: From A to Z and 0 to 9

A B C D E F G H I J K L M N O P Q R S T U V W X Y Z Numbers

Abbreviation	Network Protocol Description	Sponsor Organization
A		
AAL	ATM Adaptation Layer Interface	ITU-T
AAL0	ATM Adaptation Layer Type 0 refers to raw ATM cells.	ITU-T
AAL1	ATM Adaptation Layer Type 1 supports constant bit rate, time-dependent traffic such as voice and video.	ITU-T
AAL2	ATM Adaptation Layer Type 2 reserved for variable bit rate video transfer.	ITU-T
AAL3/4	ATM Adaptation Layer Type 3/4 supports variable bit rate, delay-tolerant data traffic requiring some sequencing and/or error detection support.	ITU-T
AAL5	ATM Adaptation Layer Type 5 supports variable bit rate, delay-tolerant connection-oriented data traffic requiring minimal sequencing or error detection support.	ITU-T
AARP	AppleTalk Address Resolution Protocol	Apple
ABP	AppleTalk Broadcast Protocol	Apple
ACSE	ISO Association Control Service Element	ISO
ADSL	Asynchronous Digital Subscriber Line	ANSI/ITU
ADSL Lite	Universal ADSL and splitterless ADSL; also known as G.lite	ANSI/ITU
ADSP	AppleTalk Data Stream Protocol	Apple
AEP	AppleTalk Echo Protocol	Apple
AFP	AppleTalk Filing Protocol	Apple
AH	Authentication Header in IPSEC suite	IETF
APPC	Advanced Program to Program Communication	IBM
AppleTalk	Apple Computer protocol suite for LAN and internet communications	Apple
APPN	Advanced Peer to Peer Networking	IBM
ARP	Address Resolution Protocol	IETF
ASN.1	Abstract Syntax Notation number 1 (specifically LDAP, Kerberos, SNMP and ITU VoIP protocols)	ITU
ASP	AppleTalk Session Protocol	Apple
ATCP	AppleTalk Control Protocol	Apple
ATIP	AppleTalk Tunneling Through IP	IETF
ATM	Asynchronous Transfer Mode with fixed 53-byte cells to support voice, data and vidio support a minimum rate of 1.544 Mbps (DS1)	ITU-T
ATP	AppleTalk Transaction Protocol	Apple
AURP	AppleTalk Update-based Routing Protocol	Apple

Abbreviation	*Network Protocol Description*	*Sponsor Organization*
B		
BACP	Bandwidth Allocation Control Protocol (PPP suite)	IETF
BAP	Bandwidth Allocation Protocol (PPP suite)	IETF
BCAST	Broadcast Protocol	Novell
BCC	Broadcast Call Control	Novell
BCP	Bridging Control Protocol (PPP suite)	IETF
BGP	Border Gateway Protocol	IETF
BGP-4	Border Gateway Protocol version 4	IETF
BGMP	Border Gateway Multicast Protocol	IETF
B-ICI	BISDN Inter Carrier Interface (ATM PNNI Signaling)	ITU-T
BISDN	Broadband Integrated Services Digital Network	ITU-T
BISUP	Broadband ISDN User Part	ITU-T
Bluetooth	Specifications for wireless communication between personal devices such as PCs, cordless phone, headsets and PDAs within 10 meter.	Bluetooth SIG
BMP (Burst)	Burst Mode Protocol	Novell
BOOTP	Bootstrap Protocol (BOOTP)	IETF
BPDU	Bridge Protocol Data Unit	IETF
BVCP	PPP Banyan VINES Control Protocol (PPP suite)	BANYAN
C		
C7	Signalling System #7 for telephony signalling	ITU-T
CCP	Compression Control Protocol (PPP suite)	IETF
CDMA	Code Division Multiple Access for cellular phone and wireless LAN	Qualcomm
CDMA2000	Code Division Multiple Access 2000 for 3G cellular phone and wireless LAN	Qualcomm
CDP	Cisco Discovery Protocol	Cisco
CGMP	Cisco Group Management Protocol	Cisco
CHAP	Challenge Handshake Authentication Protocol PPP Authentication	IETF
CIF	Cells in Frames (ATM over LAN)	ITU-T
CIFS	Common Internet File System	Microsoft
CMIP	Common Management Information Protocol	ISO
CMIS	Common Management Information Service	ISO
CMOT	CMIP Over TCP/IP	ISO/IETF
CLDAP	Connectionless Lightweighted Directory Access Protocol (RFC 3352, Not active)	IETF
CLNP	OSI Connectionless Network Protocol	ISO
CONP	OSI Connection-Oriented Network Protocol	ISO
COPS	Common Open Policy Service	IETF
CR-LDP	Constraint-based Label Distribution Protocol	IETF

Abbreviation	*Network Protocol Description*	*Sponsor Organization*
CSMA/CD	Ethernet Carrier Sense Multi-Access with Collision Control; Media Access Control as defined by IEEE 802.3	IEEE
CTERM	Command Terminal (DECnet)	DEC/HP
D		
DAP	Data Access Protocol (DECnet)	DEC/HP
DAP	Directory Access Protocol (X.500)	ISO & ITU
DCAP	Data Link Switching Client Access Protocol	IETF
DCP	Data Compression Protocol over Frame Relay	IETF
DCPCP	DCP Control Protocol	IETF
DDP	Datagram Delivery Protocol (AppleTalk)	Apple
DECnet	Digital Equipment Corporation protocols suite	DEC/HP
DHCP	Dynamic Host Configuration Protocol	IETF
DIAG	Diagnostic Responder protocol	Novell
DiffServ	Differenciated Service in IP network	IETF
Diagnostic Protocol	Diagnostic Protocol	DEC/HP
DISL	Dynamic Inter-Switch Link Protocol	Cisco
DLSw	Data Link Switching	IBM
DNA	Digital Network Architecture	DEC/HP
DNCP	PPP DECnet Phase IV Control Protocol (PPP suite)	DEC/HP
DNS	Domain Name System (Service)	IETF
DOCSIS	Data Over Cable Service Interface Specification	CableLabs
DQDB	Distributed Queue Dual Bus Defined in IEEE 802.6	IEEE
DRARP	Dynamic Reverse Address Resolution Protocol	IETF
DRIP	Duplicate Ring Protocol	Cisco
DS0	Digital Signal for voice channel of 64 Kbps	ANSI
DS1/T1	Digital Signal for voice channel of 1.544 Mbps supports 24 DS0	ANSI
DS3/T3	Digital Signal for voice channel of 44.736 Mbps supports 28 DS1/T1	ANSI
DSL	Digital Subscriber Line	ANSI/ITU
DSMCC	Digital Storage Media Command and Control (Audio Visual over ATM)	
DTS	DNA Time Service	DEC/HP
DTP	Dynamic Trunk Protocol	Cisco
DVMRP	Distance Vector Multicast Routing Protocol	IETF
DUP	Data User Part in the SS7 protocol suite	ITU-T

Abbreviation	*Network Protocol Description*	*Sponsor Organization*
E		
E1	Digital Signal for voice channel of 2.048 Mbps supports 30 DS0; a standard for European and some other international countries	ITU-T
E3	Digital Signal for voice channel of 34.368 Mbps supports 16 E1; a standard for European and some other international counties	ITU-T
EAP	PPP Extensible Authentication Protocol	IETF
EAPOL	EAP over LAN as defined in IEEE 802.1X	IEEE
ECHO	Echo Protocol	IETF
ECP	Encryption Control Protocol (PPP suite)	IETF
EGP	Exterior Gateway Protocol	IETF
EIGRP	Enhanced Interior Gateway Routing Protocol	Cisco
ES-IS	End System to Intermediate System Routing Exchange protocol	ISO
ESP	Encapsulating Security Payload	IETF
Ethernet	LAN protocol synonymous with IEEE 802.3 standard	IEEE
F		
FANP	Flow Attribute Notification Protocol	Toshiba
Fast Ethernet	100Mbps Ethernet LAN as defined by IEEE 802.3u	IEEE
FDDI	Fiber Distributed Data Interface for optical LAN	ANSI
Fibre Channel	Protocol to provide fast data transfer between computers with a rate up to 1 Gbps over coaxial cable or fiber	ANSI
FCIP	Fibre Channel over TCP/IP	IETF
FCP	Fibre Channel Protocol	ANSI
Finger	User Information Protocol	IETF
Frame Relay	A WAN protocol to support inter-network communications at layer 2	ITU-T/ANSI
FTAM	File Transfer Access Management Protocol	ISO
FTP	File Transfer Protocol	IETF
FUNI	Frame-based UNI (User Network Interface)	IETF
G		
G.711	Pulse code modulation (PCM) of voice frequencies	ITU
G.721	32 kbit/s adaptive differential pulse code modulation (ADPCM)	ITU
G.722	7 kHz audio-coding within 64 kbit/s	ITU
G.723	Extensions adaptive differential pulse code modulation	ITU
G.726	Audio coding at 40, 32, 24, 16 kbit/s adaptive differential pulse code modulation (ADPCM)	ITU

Abbreviation	*Network Protocol Description*	*Sponsor Organization*
G.727	Audio coding at 5-, 4-, 3- and 2-bit/sample embedded adaptive differential pulse code modulation (ADPCM)	ITU
G.728	Coding of speech at 16 kbit/s using low-delay code excited linear prediction	ITU
G.729	Coding of speech at 8 kbit/s using conjugate-structure algebraic-code-excited linear-prediction	ITU
GARP	Generic Attributes Registration Protocol (IEEE 802.1P)	IEE
Gigabit Ethernet	1000Mbps Ethernet LAN as defined by IEEE 802.3z and 802.3ab	IEEE
GMM	GPRS Mobility Management	ITU-T
GMRP	802.1P GARP Multicast Registration Protocol	IEEE
GRE	Generic Routing Encapsulation	IETF
G.Lite	Splitterless ADSL	ANSI/ITU
GSM	GPRS Session Management	ITU-T
GSMP	General Switch Management Protocol (IP Switching)	ITU-T
GTP	GPRS Tuneling Protocol	ITU-T
GVRP	GARP VLAN Registration protocol	IEEE
H		
H.225	Call signaling and media stream packetization and RAS	ITU
H.235	Security for H.323 based systems	ITU
H.245	Control Protocol for Multimedia communication	ITU
H.248	Media Gateway Control protocol , same as Megaco	ITU
H.261	Video Codec	ITU
H.263	Video Codec	ITU
H.323	Packet-based multimedia communications (VoIP) architecture	ITU
HDLC	High Level Data Link Control protocol developed by ISO, based on pioneering work by IBM on SDLC.	ISO
HDSL	High Date Rate Digital Subscriber Line	ANSI/ITU
HPR-APPN	High Performance Routing for Advanced Peer to Peer Network, IBM network architecture for dynamic routing across arbitrary network topologies.	IBM
HSRP	Hot Standby Router Protocol	Cisco
HTML	Hypertext Markup Language	IETF
HTTP	HyperText Transfer Protocol	IETF
HTTPS	Hypertext Transfer Protocol Secure	IETF

Abbreviation	Network Protocol Description	Sponsor Organization
I		
ICMP	Internet Control Messages Protocol	IETF
ICMPv6	Revision of ICMP for IPv6	IETF
ICP	VINES Internet Control Protocol	Banyan
IDSL	ISDN Digital Subscriber Line	ANSI/ITU
IDP	Internet Datagram Protocol (XNS)	Xerox
IDRP	Inter-Domain Routing Protocol	ISO
iFCP	Internet Fibre Channel Protocol	IETF
IFMP	Ipsilon Flow Management Protocol (IP Switching)	IETF
IGMP v1, v2, v3	Internet Group Management Protocol version 1, version 2, and version 3	IETF
IGRP	Interior Gateway Routing Protocol	Cisco
IISP	Interim Interswitch Signaling Protocol (ATM Signaling)	ETSI & ECSA
IKE	Internet Key Exchange Protocol in IPsec	IETF
ILMI	Interim Local Management Interface, bi-direction exchange of management information between UMEs.	IETF
IMAP	Internet Message Access Protocol	IETF
IMT-2000	3G wireless communication protocol same as CDMA2000	ITU-T
InARP	Inverse Address Resolution Protocol	IETF
IP	Internet Protocol for packet addressing and routing in network (LAN and WAN)	DARPA/IETF
IP NetBIOS Datagram Service	IP NetBIOS Datagram Service	IETF
IPC	InterProcess Communications protocol, datagram and reliable message delivery service for Banyan.	DARPA
IPCP	IP Control Protocol, responsible for configuring the IP parameters on both ends of the PPP link.	IETF
IPHC	IP Header Compression	IETF
IPsec	Security Architecture for IP network and IP Security Protocols	IETF
IPv4	Internet Protocol version 4	DARPA/IETF
IPv6	Internet Protocol version 6	IETF
IPv6CP	IPv6 PPP Control Protocol, responsible for configuring, enabling and disabling the IPv6 protocol modules on both ends of a PPP link.	IETF
IPX	Internetwork Packet Exchange, Novell network layer protocol similar to IP	Novell
IPXCP	IPX PPP Control Protocol, choose and configure the IPX network-layer protocol over PPP.	Novell
IRCP	Internet Relay Chat Protocol	IETF
IRDP	ICMP Router Discovery Protocol	IETF
ISAKMP	Internet Security Association and Key Management Protocol	IETF
iSCSI	Internet Small Computer Systems Interface for fast storage data transfer over TCP/IP	IETF

Abbreviation	*Network Protocol Description*	*Sponsor Organization*
ISDN	Integrated Services Digital Network for a circuit switched network to support access of voice, data and vedio	ITU-T
IS-IS	IS-IS (ISO 10589) - Intermediate System to Intermediate System, exchange of configuration and routing information to facilitate the operation of the routing and relaying functions of the network layer.	ISO
ISL	Inter-Switch Link Protocol	Cisco
iSNS	Internet Storage Name Service Protocols	IETF
ISO-IP	ISO Internetworking Protocol, also named CLNP (Connectionless Network Protocol), is similar to the IP (Internet Protocol) defined by DARPA.	ISO
ISO-PP	OSI Presentation Layer Protocol (ISO 8823 / X.226)	ISO
ISO-SP	OSI Session Layer Protocol (ISO 8327 / X.225)	ISO
ISO-TP	OSI Transport Layer Protocol: TP0, TP1, TP2, TP3, TP4 (ISO 8073)	ISO
ISO VTP	ISO Virtual Terminal Protocol	ISO
ISUP	ISDN User Part of SS7, defines protocol and procedures used to setup, manage and release trunk circuits that carry voice and data calls over the public switched telephone network.	ITU-T
ITOT	ISO Transport Service on top of TCP	ISO/IETF
K		
KERBEROS	Kerberos network authentication protocol	MIT
L		
L2F	Layer 2 Forwarding protocol, permits the tunneling of the link layer of higher layer protocols.	Cisco
L2TP	Layer 2 Tunneling Protocol, used for integrating multi-protocol dial-up services into existing ISP POP.	IETF
LAN	Local Area Network	
LANE-UNI	LAN Emulation - User to Network Interface	ITU
LANE-NNI	LAN Emulation - Network to Network Interface	ITU
LAPB	Link Access Procedure/Protocol Balnced	ITU/CCITT
LAPD	ISDN Link Access Protocol, Channel D.	ITU/CCITT
LAPF	Link Access Procedure F (Frame Relay), modified LAPD standard for Frame Relay.	ITU/CCITT
LAT	Local Area Transport protocol, designed to handle multiplexed terminal traffic to/from timesharing hosts (DECnet).	DEC/HP

Abbreviation	*Network Protocol Description*	*Sponsor Organization*
LAVC	Local Area VAX Cluster protocol, communications between DEC VAX computers in a cluster (DECnet).	DEC/HP
LCP	Link Control Protocol, establishes, configures and tests the data link connection (PPP suite).	IETF
LDAP	Lightweight Directory Access Protocol	IETF
LDP	Label Distribution Protocol	IETF
LLC	Logical Link Control protocol (IEEE 802.2), provides a link mechanism for upper layer protocols.	IEEE
LPP	Lightweight Presentation Protocol	ISO / IETF
LQR	Link Quality Report, specifies the mechanism for link quality monitoring with PPP.	IEEE
LU	SNA Logical Units: LU0, LU1, LU2, LU3, LU6.2, for communication session management	IBM

M

M2UA	Protocol for backhauling of SS7 MTP2-User signaling messages over IP using Stream Control Transmission Protocol (SCTP)	ITU-T
M3UA	Protocol to support transport of SS7 MTP3-User signaling over IP using Stream Control Transmission Protocol (SCTP)	ITU-T
MACIP	Mac Internet Protocol	Apple
MAP	Mobile Application Part, a signaling protocol in the SS7 suite	ITU-T
MAN	Metropolitan Area Network	
MAPOS	Multiple Access Protocol over SONET / SDH	ITU-T
MARS	Multicast Address Resolution Server	IETF
MBGP	Multiprotocol BGP	IETF
MEGACO	Media Gateway Control Protocol, same as H.248	IETF & ITU-T
MGCP	Media Gateway Control Protocol for VOIP	Cisco
MIME	Multipurpose Internet Mail Extensions	IETF
MHS	Message Handling Service	ISO
MLP	Multilink Procedure, added upper sublayer of the LAPB, operating between the packet layer and a multiplicity of single data link protocol functions (SLPs) in the data link layer (X.25).	IETF
MOP	Maintenance Operation Protocol, utility services such as uploading and downloading system software, remote testing and problem diagnosis (DECnet).	DEC/HP
MOSPF	Multicast Extensions to OSPF	IETF
MOUNT	Protocol used to initiate client access to a server supporting NFS.	IETF
MP	MultiLink PPP (MultiPPP)	IETF
MPEG	Motion Pictures Experts Group for data compression of motion video	ISO

Abbreviation	*Network Protocol Description*	*Sponsor Organization*
MPLS	Multi-Protocol Label Switching to support IP services over ATM and Frame Relay networks through set of procedures for augmenting network layer packets with “label stacks”, thereby turning them into labeled packets.	IETF
MPOA	Multi Protocol Over ATM, deals with efficient transfer of inner-subnet unicast data in a LAN emulation environment.	ITU-T
MPPC	Microsoft Point-to-Point Compression Protocol	Microsoft
MSDP	Multicast Source Discovery Protocol	IEEE
MSRAP	Microsoft Remote Administration Protocol	Microsoft
MSRPC	Microsoft Remote Procedure Call	Microsoft
MTP-2	Message Transfer Part, Level 2, signaling link to provides reliable transfer of signaling messages between two directly connected signaling points (SS7).	ITU-T
MTP-3	Message Transfer Part, Level 3, connects Q.SAAL to the users (SS7 suite).	ITU-T
MZAP	Multicast-Scope Zone Announcement Protocol	IETF

N

NARP	NBMA Address Resolution Protocol	IETF/Novell
NAT	Network Address Translation	IETF
NAU	Systems Network Architectre (SNA) Network Accessible Units	IBM
NBFCP	PPP NetBIOS Frames Control Protocol, network control protocol for establishing and configuring the NBF protocol over PPP.	IETF
NBMA	Non-Broadcast, Multi-Access	Novell
NBP	AppleTalk Name Binding Protocol	Apple
NBSS	NetBIOS Session Service over TCP/IP	IETF
NCP	NetWare Core Protocol	Novell
NDS	NetWare Directory Services, globally distributed network database	Novell
NDMP	Network Data Management Protocol	SNIA
NetBEUI	NetBIOS Extended User Interface	IBM
NetBIOS	Network Basic Input/Output System	IBM
NetRPC	NetRemote Procedure Call, used to access VINES applications such as StreetTalk and VINES Mail	Banyan
NetWare	Novell Network Operating System (NOS)	Novell
NFS	Sun Network File System, file sharing application for the Sun protocol suite.	Sun
NHDR	Network Layer Header, begins the frame used by RTP nodes.	Novell
NHRP	NBMA Next Hop Resolution Protocol.	IETF/Novell
NLSP	NetWare Link Services Protocol	Novell
NNTP	Network News Transfer Protocol	IETF
NSP	Network Services Protocol, provides reliable virtual connection	

Abbreviation	*Network Protocol Description*	*Sponsor Organization*
	services with flow control to the network layer Routing Protocol (DECnet).	DEC/HP
NTP	Network Time Protocol, time synchronization system for computer clocks through the Internet network .	DARPA/IETF
O		
OC1	SONET/SDH transmission with rate of 51.84 Mbps	ANSI/ITU-T
OC12	SONET/SDH transmission with rate of 622.08 Mbps	ANSI/ITU-T
OC192	SONET/SDH transmission with rate of 9.953 Gbps	ANSI/ITU-T
OC3	SONET/SDH transmission with rate of 155.52 Mbps	ANSI/ITU-T
OC48	SONET/SDH transmission with rate of 2.488 Gbps	ANSI/ITU-T
OC768	SONET/SDH transmission with rate of 39.812 Gbps	ANSI/ITU-T
OC96	SONET/SDH transmission with rate of 4.976 Gbps	ANSI/ITU-T
OSI Model	Open Systems Interconnection network reference model	ISO
OSI NLCP	PPP OSI Network Layer Control Protocol, responsible for configuring, enabling and disabling the OSI protocol modules on both ends of the PPP link.	IETF
OSPF	Open Shortest Path First, link-state routing protocol used for routing IP.	IETF
P		
PAP	Password Authentication Protocol, provides a simple method for the peer to establish its identity using a 2-way handshake.	IETF
PAP	Printer Access Protocol, manages the virtual connection to printers and other servers (AppleTalk).	Apple
PEP	Packet Exchange Protocol, provides a semi-reliable packet delivery service that orients towards single-packet exchanges.	Xerox
PIM	Protocol Independent Multicast	IETF
PIM-DM	Protocol Indelendent Multicast - Dense Mode	IETF
PIM-SM	Protocol Independent Multicast-Sparse Mode	IETF
PGM	Pragmatic General Multicast Protocol	IETF
PMAP	Port Mapper protocol, manages the allocation of transport layer ports to network server applications (Sun).	Sun
RMON	Remote Monitoring MIBs in SNMP (RMON1 & RMON2)	IETF
PNNI	Private Network-to-Network Interface, hierarchical, dynamic link-state routing protocol (ATM).	ETSI & ECSA
POP	Post Office Protocol	IETF

Abbreviation	*Network Protocol Description*	*Sponsor Organization*
POP3	Post Office Protocol version 3, permits workstations to dynamically access a maildrop on a server host (TCP/IP).	IETF
PP	ISO Presentation Protocol, performs context negotiation and management between open systems.	ISO
PPP	Point-Point Protocol: a data link layer protocol for two peer devices to transport packets over the link	IETF
PPP Multilink	Multilink Point-to-Point Protocol, permits a system to indicate to its peer that it is capable of combining multiple physical links into a “bundle”.	IETF
PPP-BPDU	PPP Bridge Protocol Data Unit, used to connect remote bridges.	IETF
PPPoA	PPP over ATM	IETF
PPPoE	PPP over Ethernet	IETF
PPTP	Point-to-Point Tunneling Protocol, allows PPP to be channeled through an IP network.	Microsoft
Q		
Q.2931	Signaling standard for ATM to support Switched Virtual Connections. This is the signaling standard for ISDN.	ETSI & ECSA
Q.850	Usage of cause and location	ITU
Q.922	Link Access Procedure/Protocol (LAPF) as defined in the ITU Q.922	ITU
Q.931	ISDN user-network interface layer 3 for Basic Call Control	ITU
Q.932	Digital subscriber signaling system No. 1	ITU
Q.952	Stage 3 description for call offering supplementary services using DSS 1 - Diversion supplementary services	ITU
Q.953	Call waiting/Call hold/Completion of Calls to Busy Subscribers (CCBS)/Terminal Portability (TP)/Call Completion on No Reply (CCNR)	ITU
Q.955	Closed user group/Multi-level precedence and preemption (MLPP)	ITU
Q.956	Advice of charge/Reverse charging	ITU
Q.957	User-to-User Signaling (UUS)	ITU
QLLC	Qualified Logical Link Control protocol, transfers IBM SNA data over an X.25 network.	IBM
R		
RADIUS	Remote Authentication Dial-in User Service for user authentication and accounting	IETF
RANAP	Radio Access Network Application Part is the Radio Network Layer signaling protocol	

Abbreviation	*Network Protocol Description*	*Sponsor Organization*
RARP	Reverse Address Resolution Protocol	IETF
RAS	Registration, Admission and Status (H.225)	ITU
RDP	Reliable Data Protocol	IETF
RFC	Request for Comments, a series of notes about the Internet, started in 1969 (when the Internet was the ARPANET). The notes discuss many aspects of computing and computer communication focusing in networking protocols, procedures, programs, and concepts, but also including meeting notes, opinion, and sometimes humor. The specification documents of the Internet protocol suite, as defined by the Internet Engineering Task Force (IETF) and its steering group (the IESG), are published as RFCs. Many of the TCP/IP protocols and PPP protocols are defined by rfc's.	IETF
RGMP	Cisco Router Port Group Management Protocol	Cisco
RIP	Routing Information Protocol	IETF
RIP2	Routing Information Protocol version 2	IETF
RIPng	Routing Information Protocol for IPv6	IETF
RIPX	Novell Routing Information Protocol, used to collect, maintain and exchange correct routing information among gateways within the Internet	Novell
rlogin	Remote Login, allows UNIX users of one machine to connect to other UNIX systems across the Internet and interact as if their terminals are directly connected to the machines (TCP/IP).	IETF
RLP	Radio Link Protocol	IEEE
RM Cells	Rate Management cells (ATM cells).	ETSI & ECSA
RMON	Remote Monitoring MIBS - belong to the SNMP protocol family	IETF
ROSE	ISO Remote Operation Service Element protocol	ISO
RP	DECnet Routing Protocol, distributes routing information among DECnet hosts.	DEC/HP
RPC	Remote Procedure Call protocol, activates a function on a remote station and retrieves the result	Sun
RPC Mount Procedures	RPC Mount Procedures	Sun
RPC NFS Procedures	RPC NFS Procedures	Sun
RPC-PMP	RPC Port Mapper Procedures	Sun
RSH/RCMD	Remote Shell/Remote Command	IETF
RSVP	Resource ReSerVation Protocol, designed for an integrated services Internet.	IETF
RSVP-TE	Resource Reservation Protocol: Traffic Extension	IETF
RTCP	Real-time Transport Control Protocol	IETF
RTMP	Routing Table Maintenance Protocol, manages routing information for AppleTalk networks.	IETF
RTP	VINES Routing Update Protocol, used to distribute network topology (Banyan).	Banyan

Abbreviation	*Network Protocol Description*	*Sponsor Organization*
RTP	Real-time Transport Protocol	IETF
RTSE	Reliable Transfer Service Element	ISO
RTSP	Real Time Streaming Protocol	IETF
RUDP	Reliable UDP	IETF

S

SAN	Storage Area Network	
SAP	Novell's Netware Service Advertising Protocol, provides information about what servers are available on the network.	Novell
SAP	Session Announcement Protocol	IETF
SAS	Serial Attached SCSI	ANSI
SCCP	Signaling Connection Control Part, offers enhancements to MTP level 3 to provide connectionless and connection-oriented network services, as well as to address translation capabilities (SS7).	ITU-T
SCCP	Skinny Client Control Protocol	Cisco
SCP	Session Control Protocol, manages logical links for DECnet connections.	DEC
SCSI	Small Computer Systems Interface for fast data transfer in Storage Area Network	ANSI
SCSP	Server Catche Synchronization Protocol	IETF
SCTP	Stream Control Transmission Protocol	IETF
SDCP	PPP Serial Data Control Protocol, responsible for configuring, enabling and disabling the SDTP modules on both ends of the point-to-point link.	IETF
SDH	Synchronous Data Link Hierarchy, a European standard for data transmission over fiber equivalent to SONET	ITU-T
SDLC	Synchronous Data Link Control protocol, developed by IBM to be used as the layer 2 of the SNA hierarchical network.	IBM
SDSL	Single Line Digital Subscriber Line	ANSI/ITU
SDP	Session Description Protocol	IETF
SER	Serialization packet, ensures that a single version of NetWare is not being loaded on multiple servers.	Novell
SGCP	Simple Gateway Control Protocol for VOIP	Cisco & Telcordia
S-HTTP	Secure HyperText Transfer Protocol	IETF
SIP	SMDS Interface Protocol, three-level protocol that controls access to the network.	Bellcore
SIP	Session Initiation Protocol	IETF
SLIP	SLIP: Serial Line IP	IETF
SLP	Service Location Protocol	IETF

Abbreviation	*Network Protocol Description*	*Sponsor Organization*
SMB	Server Message Block protocol providing file and print sharing functions for LAN Manager, Banyan VINES and other networking operating systems.	IBM
SMDS	Switched Multi-Megabit Data Service for public network service for LAN-to-LAN interconnection with a bandwidth up to 45 Mbps (T3)	Bellcore
S/MIME	Secure Multipurpose Internet Mail Extensions	IETF
SMPP	Short Message Peer to Peer: a protocol for exchange short messages between SMS peer entities such as short message service centers.	Aldiscon/Logica
SMS	Short Message Service: a mechanism of delivery of short messages over the mobile networks	3GPP
SMTP	Simple Mail Transfer Protocol, mail service modeled on the FTP file transfer service.	IETF
SNA	Systems Network Architecture, introduced by IBM to provide a framework for joining together many mutually incompatible IBM products for distributed processing.	IBM
SNACP	SNA PPP Control Protocol, responsible for configuring, enabling and disabling SNA on both ends of the point-point link (PPP).	IBM
SNAP	Subnetwork Access Protocol for LAN logical link control frame	IEEE/IETF
SNI	Subscriber Network Interface such as DS0, DS1/T1, DS3/T3, E1, E3 to access to the SMDS network	Bell Labs
SNMP	Simple Network Management Protocol: allow diverse network objects to be managed in a global network management architecture.	IETF
SNMPv1	SNMP version 1	IETF
SNMPv2	SNMP version 2	IETF
SNMPv3	SNMP version 3	
SNMP OID	SNMP Object Identifiers	IETF
SNMP RMON	SNMP Remote Network Monitoring	IETF
SNMP SMI	SNMP Structure of Management Information (RFC 1155)	IETF
SNTP	Simple Network Time Protocol	IETF
SOAP	Simple Object Access Protocol	Microsoft
SOCKS	A circuit level security technology supports multiple means of authentication, negotiation between client and server over a virtual circuit	IETF
SONET	Synchronous Optical Network, a North America standard for data transmission over fiber from 51.84 Mbps to 39.812 Gbps	ANSI
SPANS	Simple Protocol for ATM Network Signaling, developed by FORE Systems for use in ATM networks.	FORE
SPP	Sequenced Packet Protocol, provides a reliable virtual connection service for private connections	Banyan
SPP	Sequenced Packet Protocol	Xerox

Abbreviation	*Network Protocol Description*	*Sponsor Organization*
SPX	Sequenced Packet Exchange, Novell's transport layer protocol providing a packet delivery service for third party applications.	Novell
SRB	Source Routing Bridging, proprietary header of Bay Networks which passes Token Ring information over WAN lines.	BAY Networks
SRP	Spatial Reuse Protocol.	
SS7	Signaling System 7, a common channel signaling system.	ITU-T
SSH	Secure Shell	SSH
STP	Spanning Tree Protocol, prevents the formation of logical looping in the network (IEEE 802.1D)	IEEE
T		
T.120	Multipoint Data Conferencing and Real Time Communication Protocols (T.121, T.122, T.123, T.124, T.125, T.126, T.127)	ITU
TACACS+	Terminal Access Controller Access Control System (version 3)	Cisco
TALI	Transport Adapter Layer Interface	IETF
TARP	TID Address Resolution Protocol	ISO/OSI
TCAP	Transaction Capabilities Application Part, enables the deployment of advanced intelligent network services using the SCCP connectionless service (SS7).	ITU-T
TCP	Transmission Control Protocol for reliable, sequnced, and unduplicated delivery of bytes to end user, as part of the IP suite at layer 4	DARPA/IETF
TCP/IP	TCP over IP protocols which are the core protocols for Internet communications	DARPA/IETF
TDP	Tag Distribution Protocol, a two party protocol that runs over a connection oriented transport layer with guaranteed sequential delivery.	Cisco
TELNET	Network Virtual Terminal protocol for terminal emulation	IETF
TFTP	Trivial File Transfer Protocol, supports file writing and reading	IETF
THDR	Transport Layer Header, used by RTP endpoints to provide correct processing of the packet (SNA).	IBM
Timeplex (BRE2)	Bridge Relay Encapsulation, proprietary Ascom Timeplex protocol that extends bridging across WAN links by means of encapsulation (Frame Relay).	IETF
TLS	Transport Layer Security Protocol	IETF
Token Ring	LAN protocol where all stations are connected in a ring and each station can directly hear transmissions only from its immediate neighbor. (IEEE 802.5)	IEEE
TP	ISO Transport Protocol class 0, 1, 2, 3, 4 (TP0, TP1, TP2, TP3, TP4)	ISO
TUP	Telephone User Part in the SS7 protocol suite	ITU-T

Abbreviation	*Network Protocol Description*	*Sponsor Organization*
U		
UDP	User Datagram Protocol to provide layer 4 connectionless without acknowledgement delivery of packets to end users, as part of the IP suite	DARPA/IETF
UMTS	Universal Mobile Telecommunication System for next generation of GSM	ETSI
UNI	User Network Interface, an interface point between ATM end users and a private ATM switch, or between a private ATM switch and the public carrier ATM network.	ITU
URL	Uniform Resource Locator	IETF
V		
Van Jacobson	Compressed TCP protocol which improves the TCP/IP performance over low speed serial links.	IETF
VARP	VINES Address Resolution Protocol, used for finding the node Data Link Control address from the node IP address (Banyan).	Banyan
VCI	Virtual Channel Identifier.	IETF
VDSL	Very high date rate Digital Subscriber Line	ANSI/ITU
VINES	Virtual Integrated Network Service	Banyan
VINES IP	VINES Internet Protocol	Banyan
VINES IPC	VINES Interprocess Communication Protocol	Banyan
VLAN	Virtual LAN supported by IEEE 802.1Q etc.	IEEE
VOIP	Voice over IP - transmit voice over IP packet network - supported by protocols such as H.323, SIP, MGCP, MEGACO etc.	
VPI	Virtual Path Identifier	IETF
VRRP	Virtual Router Redundancy Protocol (RFC 2338)	IETF
VTP	VLAN Trunking Protocol	Cisco
W		
WAE	Wireless Application Environment	IEEE
WAN	Wide Area Network	
WAP	Wireless Application Protocol	IEEE
WCCP	Web Cache Communication Protocol	IETF/Cisco
WDOG	Watchdog protocol, provides constant validation of active workstation connections.	Novell
Wellfleet BOFL	Wellfleet Breath of Life, used as a line sensing protocol.	IEEE
Wellfleet SRB	Source Routing Bridging, proprietary header of Bay Networks	

Abbreviation	*Network Protocol Description*	*Sponsor Organization*
	which passes Token Ring information over WAN lines.	BAY Networks
Whois	Whois directory access protocol	IETF
WiFi	Wireless Fidelity - Nick name for 802.11b for wireless LAN with bandwidth up to 11 Mbps	IEEE
WINS	Windows Internet Name Service	Microsoft
WiMAX	IEEE 802.16: Broadband Wireless MAN Standard	IEEE
WLAN	Wireless Lan supported by IEEE 802.11, 802.11a, 802.11b, 802.11g	IEEE
WML	Wireless Markup Language for WAP related browser	IEEE
WSP	Wireless Session Protocol	IEEE
WTLS	Wireless Transport Layer Security	IEEE
WTP	Wireless Transaction Protocol	IEEE
X		
X Window	X Window/X Protocol: X Window System Protocol for Unix and Linux Graphic Display	X.Org
X.25	A WAN communication protocol stack	ITU-T
X.75	Signaling system used to connect X.25 packet switched networks.	ITU-T
X.400	ITU/ISO Message Handling Service (email transmission service) protocols	ISO/ITU
X.500	Directory Access Service Protocols (DAP)	ISO & ITU
xDSL	Collective Digital Subscriber Line Technologies (DSL, IDSL, ADSL, HDSL, SDSL, VDSL, G.Lite)	ANSI/ITU
XNS	Xerox Network System protocols, provide routing capability and support for both sequenced and connectionless packet delivery.	Xerox
XOT	Cisco Systems' X.25 over TCP protocol.	Cisco
Y		
YP (NIS)	Sun Yellow Pages protocol, known now as Network Information Service, is a directory service used for name look-up and general table enumeration.	Sun
Z		
ZIP	AppleTalk Zone Information Protocol, manages the relationship between network numbers and zone names.	Apple

Abbreviation	*Network Protocol Description*	*Sponsor Organization*
Numbers		
1000Base-CX	Gigabit over 150 ohm coaxial cable up to 200 meter for Ethernet	IEEE
1000Base-LX	Gigabit over fiber with long wave laser up to 3 kilometers for Ethernet	IEEE
1000Base-SX	Gigabit over fiber with short wave laser up to 550 meters for Ethernet	IEEE
1000BaseT	Gigabit over twisted pair for Ethernet	IEEE
1000BaseX	Gigabit over multiple media for Ethernet	IEEE
100BaseT	100 Mbps over twisted pair for Ethernet	IEEE
100BaseX	100 Mbps for Ethernet for multiple media: FX: Fiber	IEEE
10Base2 Thin	10 Mbps over thin coaxial cable	IEEE
10Base5 Thick	10 Mbps over 50 ohm thick coaxial cable for Ethernet	IEEE
10BaseF	10 Mbps over Fiber for Ethernet	IEEE
10BaseT	10 Mbps over twisted pair for Ethernet	IEEE
10Broad36	10 Mbps over coaxial cable up to 3600 meters with Frequency Division Multiplexing	IEEE
1Base5	1 Mbps over unshielded twisted pair for Ethernet	IEEE
802.1	IEEE protocols suite for internetworking of LAN, MAN and WAN.	IEEE
802.12	100 VG-Any LAN standard	IEEE
802.11	Wireless LAN standard suite	IEEE
802.11a	Wireless LAN standard with speed up to 54 Mbps	IEEE
802.11b	Wireless LAN standard with speed up to 11 Mbps	IEEE
802.11g	Wireless LAN standard with speed up to 54 Mbps	IEEE
802.11i	Wireless LAN security specification	IEEE
802.15	Wireless personal communication specification (Bluetooth)	IEEE
802.16	Wireless MAN specification	IEEE
802.1D	Spanning Tree Protocol	IEEE
802.1P	LAN Layer 2 traffic prioritization (QoS) specification	IEEE
802.1Q	Virtual LAN (VLAN) Switching protocol	IEEE
802.1X	LAN/WLAN Authentication and Key Management(EAPOL)	IEEE
802.2	Logical Link Control protocol	IEEE
802.3	Ethernet LAN protocol suite	IEEE
802.5	Token-passing access on ring topology using unshielded twisted pair	IEEE
802.6	Metropolitan Area Network (MAN) layer 2 standard (DQDB)	IEEE
802.3ab	Gigabit Ethernet over twisted pair (1000BaseT)	IEEE
802.3ae	10 Gigabit Ethernet standard	IEEE
802.3u	Fast Ethernet - 100 Mbps LAN	IEEE
802.3z	Gigabit Ethernet over fiber standard (1000BaseX)	IEEE

Major Networking and Telecom Standard Organizations

ANSI American National Standards Institute

25 West 43rd Street, 4th FL
New York NY 10036 USA
Tel: 212-642-4900
www.ansi.org

ISOC: Internet Society

www.isoc.org

ETSI European Telecommunications Standards Institute

650, Route des Lucioles
F-06921 Sophia Antipolis Cedex, France
Tel: 33 (0)4 92 94 42 00
www.etsi.org

IETF: Internet Engineering Task Force

1775 Wiehle Ave. Suite 102
Reston, VA 20190 USA
Tel: 703-326-9880
www.ietf.org

FCC Federal Communications Commission

455 12th Street NW
Washington DC 20554 USA
Tel: 888-225-5300
www.fcc.gov

ITU International Telecommunications Union

Place des Nations
CH-1211 Geneva 20, Switzerland
Tel: 41 22 99 51 11
www.itu.ch

IEEE Institute of Electrical and Electronics Engineers, Inc.

445 Hoes Lane P.O. Box 1331
Piscataway, NJ 08855-1331 USA
Tel: 732-981-0060
www.ieee.org

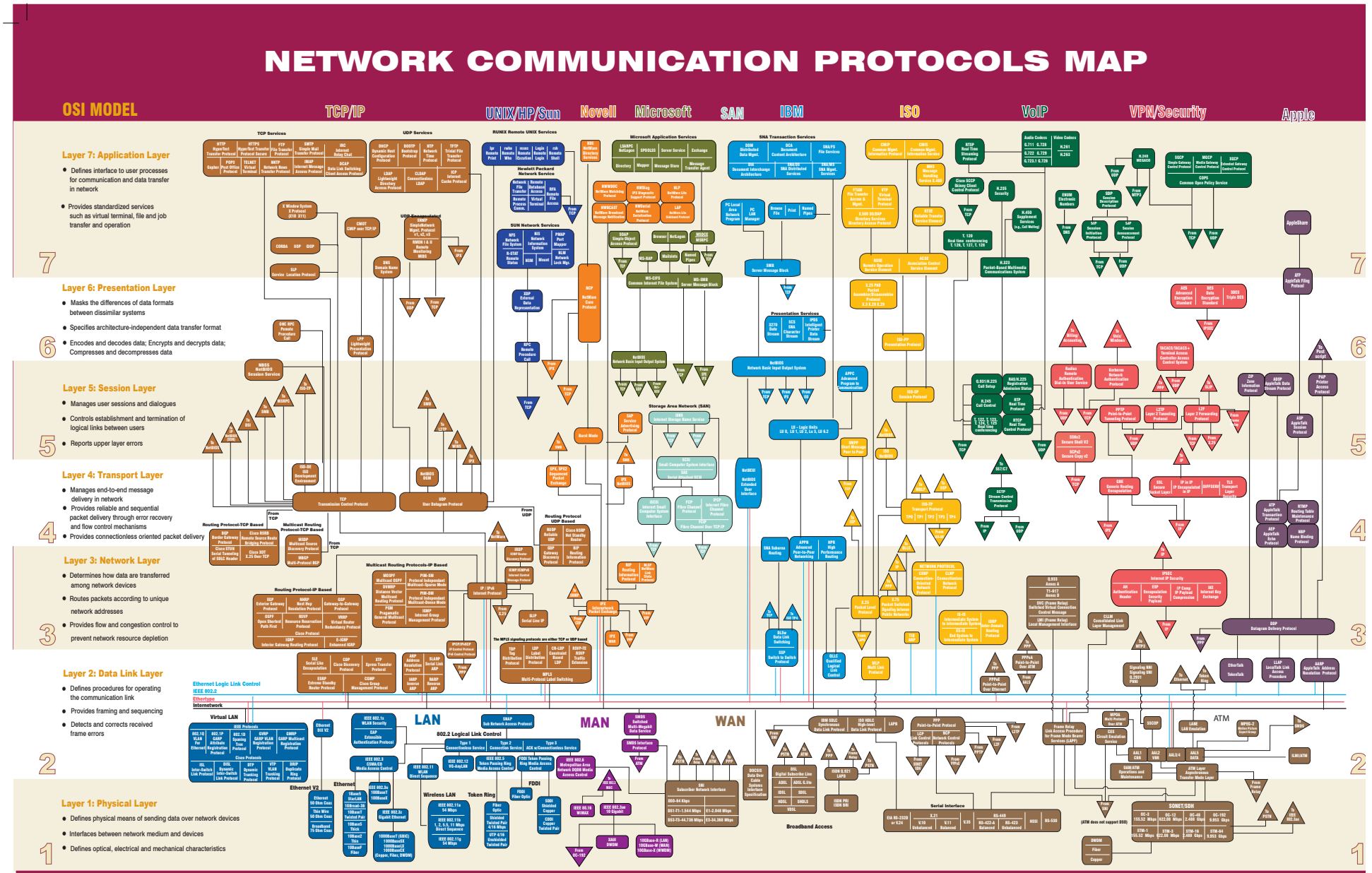
ISO International Organization for Standardization

One rue de Varembe CH-1211
Case Postale 56 Geneva 20 Switzerland
Tel: 41-22-749-0111
www.iso.ch

IEC International Electrotechnical Commission

3, rue de Varembe P.B. Box 131
1211 Geneva 20, Switzerland
Tel: 41-22-919-02-11
www.iec.ch

Network Communication Protocols Map



ANSI
American National Standards Institute
25 west 43rd Street, 4th FL
New York NY 10036 USA
Tel: 212-642-4900
www.ansi.org

ETSI
European Telecommunications Standards Institute
650, Route des Lucioles
06921 Sophia Antipolis Cedex, France
Tel: 33 (0)4 92 94 42 00
www.etsi.org

FCC
Federal Communications Commission
445 12th Street SW
Washington DC 20554 USA
Tel: 888-225-5322
www.fcc.gov

IEEE
Institute of Electrical and Electronics Engineers, Inc.
445 Hoes Lane
Piscataway, NJ 08855-1331 USA
Tel: 732-981-0060
www.ieee.org

ISO
International Organization for Standardization
One rue de Varembe CH-1211
Case Postale 56
Geneva 20, Switzerland
Tel: 41 22 749 0111
www.iso.ch

ITU
International Telecommunications Union
ITU - Place des Nations
CH-1211 Geneva 20, Switzerland
Tel: 41 22 99 51 11
www.itu.ch

ISOC: Internet Society
www.isoc.org
IETF: Internet Engineering Task Force
www.ietf.org
1775 Wiehle Ave. Suite 102
Reston VA 20190 USA
Tel: 703-326-9880

IEC
International Electrotechnical Commission
3, rue de Varembe
P.B. Box 131
CH-1211 Geneva 20, Switzerland
Tel: 41 22 919 02 11
www.iec.ch

Javvin

Second Edition

Network Protocols Handbook

"This book is an excellent reference for Internet programmers, network professionals and college students who are majoring IT and networking technologies. It is also useful for any individuals who want to know more details about Internet technologies. I highly recommend this book to our readers."

Dr. Ke Yan
Chief Architect of Juniper Networks
Founder of NetScreen Technologies

Fully explains and illustrates all commonly used network communication protocols, including TCP/IP, WAN, LAN technologies

Covers the latest and emerging technologies such as VOIP, SAN, MAN, VPN/Security,WLAN, VLAN and more

Addresses vendor specific technologies: Cisco, IBM, Novell, Sun, HP, Microsoft, Apple,etc.

Reviews the ISO networking architecture and protocols

Covers SS7 protocols

Hundreds of illustrations of protocol formats and header structures

Hundreds of references for further reading and studies

"Must-Have" for IT/Networking professionals and students

ISBN: 0-9740945-2-8

EAN: 978-0-9740945-2-6

Javvin

Javvin Technologies, Inc.

13485 Old Oak Way
Saratoga CA 95070 USA

www.javvin.com