Excerpts taken from:

Network Troubleshooting
By Othmar Kyas

An Agilent Technologies Publication

Section IV

Troubleshooting High-Layer Protocols

Chapter 16

Internet Protocols

16.6 Internet Protocol – The Next Generation (IPv6)

At the beginning of the 1990s, due to the rapid growth of the Internet and a multitude of new multimedia and security-sensitive applications, it became clear that the Internet Protocol needed to be substantially revised. Since 1992,

many IETF working groups have been dealing with suggested improvements, drafting 25 requests for comments (RFCs) on this subject. Finally in 1994, the new version of the Internet protocols called IPv6 (or IPng, for "the next generation") was adopted. The main changes in IPv6 as compared with its predecessor, IPv4, include:

- Internet addresses with a length of 128 bits (rather than 32 bits)
- A simplified header format (fields such as Time to Live and Type of Service are eliminated)
- Optional authentication and encryption
- Traffic flow components for improved multimedia capability
- Compatibility with the existing IP version to ensure smooth migration

One of the most fundamental and most urgently needed changes was the expansion of the IP address space. The move from 32-bit to 128-bit addresses increases the number of possible addresses to 2^{128} (or three times 10^{38}), and will solve the problem of address scarcity for the foreseeable future.

The number of 128-bit Internet addresses is large enough to provide every Internet host with a subaddress space that is as big as the entire IPv4 address space used in the public Internet before the adoption of IPv6.

The 128-bit IPv6 addresses are written as decimal byte values separated by colons rather than periods. For example:

145:23:45:62:47:234:567:234:678:5:2:123:23:33:4:128

16.6.1 The IPv6 Address Format

IPv6 uses three distinct types of addresses: unicast, anycast, and multicast. Anycast addresses designate a group of hosts, but packets addressed to an anycast only need to be delivered to the nearest member of the group. Multicast packets must be delivered to each of the addressees. Unicast addresses refer to individual hosts and are assigned with the following formats according to a hierarchical structure:

- Provider-based addresses
- Geographic-based addresses
- (ISO) NSAP (Network Service Access Point) addresses
- Hierarchical IPX addresses
- Local-use addresses
- IPv4-capable host addresses

The introduction of provider addresses allows companies to switch from one Internet provider to another without changing the complete addresses of their computers. The first 3 bits (010) indicate the provider-based address type, and are followed by the registry ID and the provider ID. The subscriber ID and intra-subscriber address are appended to these. When a company changes its Internet provider, the subnetwork and host addresses can remain unchanged. Only the new provider's ID and a new subscriber ID need to be inserted. This principle also benefits networks that are first created with no connection to the Internet, but later want Internet access without completely reconfiguring the network. If the private network uses local addresses, which contain only the subscriber ID and intra-subscriber address, companies have the option of converting these to public Internet addresses simply by adding the appropriate registry ID and provider ID (see Figure 16.15).

Figure 16.15 IPv6 address formats

In order to provide for the coexistence of IPv4 and IPv6 in a transitional phase, a way to integrate IPv4 addresses in the IPv6 address format also has been defined. IPv4

addressing is indicated by zeroes in the first 80 bits of an IPv6 address, followed by 16 bits with the value 0000 (or FFFF if the IPv4 host is not IPv6-capable). The original IPv4 address is then contained in the remaining 32 bits.

16.6.2 The IPv6 Data Packet Format

The format of IP data packets has also been improved in IPv6. Although IPv6 uses addresses that are four times as long, the IPv6 header is only half as long as in Version 4. Optional extension headers can follow the IPv6 header. Most of the routers that forward the IPv6 data packet on its way to its destination address will only need to analyze the IPv6 header so that routing is faster than with IPv4 headers. The IPv6 header contains the following fields:

- Version number (4 bits)
- Flow label (20 bits; used to support real-time sensitive multimedia applications)
- Payload length (16 bits)
- Next header (8 bits; indicates the type of the extension header immediately following the IPv6 header)
- Hop limit (8 bits; decremented each time the packet is forwarded. The packet is discarded when this counter reaches zero.)
- Source address (128 bits)
- Destination address (128 bits)

Other header extensions for special functions can be inserted between the IPv6 addresses and the transport layer header of the data packet. The length of these options can be any integer multiple of 8 bytes, so that even complex functions, such as encryption or authentication mechanisms, can be implemented. The following optional header fields are defined:

- Hop-by-hop
- Routing
- Fragment
- Destination
- Authentication
- Encapsulating security payload

Routing protocols, such as OSPF, RIP, and IGRP, can be used with IPv6 in the same way as with IPv4. The source station also has the option of specifying an explicit address sequence as a route, which can be used in reverse order by the recipient for replies. This mechanism is intended mainly to support the connection of mobile computer systems to IP networks.

16.6.3 Authentication and Encryption in IPv6

Authentication and security encapsulation options permit authentication of communicating systems and privacy of data contents at the IP level. The optional authentication header can be used to guarantee IPv6 data packets against corruption. The sender applies a special hash function to the data packet (MD5, Message Digest 5, is recommended) and inserts the resulting value in the authentication header. Hash functions are characterized by the fact that the corresponding inverse function is very difficult to perform. On the average, 2^{64} attempts are necessary to find a second input value that results in the same output value of a hash function. Hash functions for use as "message digests" are specified in RFCs 1319, 1320, and 1321 (MD2, MD4, and MD5).

The authentication header offers protection against manipulation of data packets, but does not guarantee the privacy of the data sent. This is the purpose of the Encapsulation Security header. It can be used to transport parts of the data packet or the entire payload in encrypted form.

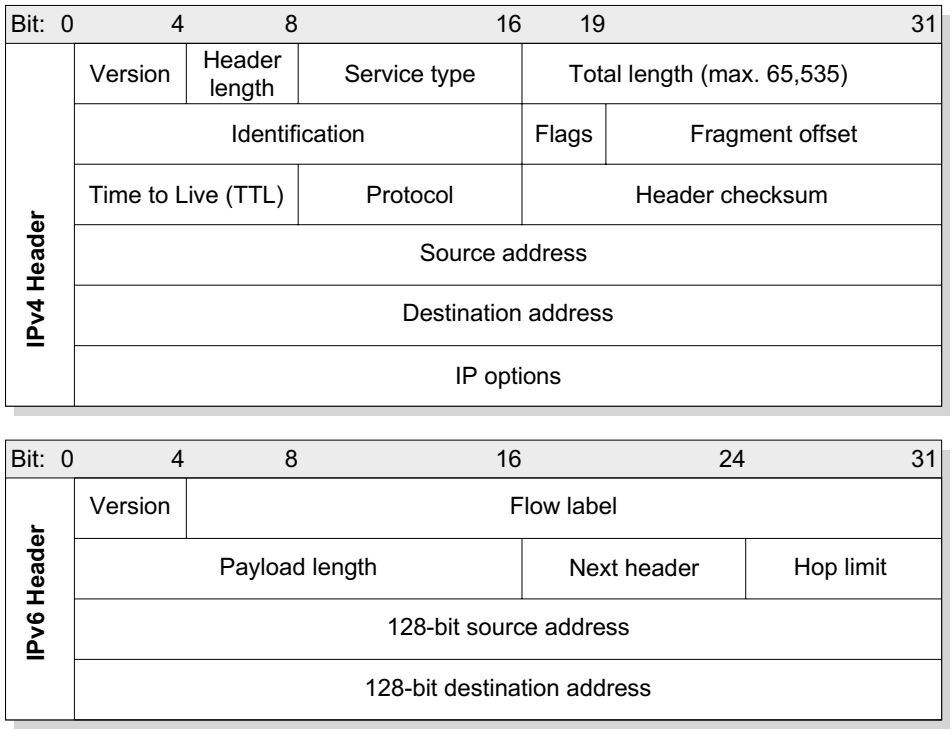

Figure 16.16 The IPv6 and IPv4 data packet format

16.7 The NetBIOS Protocol

The NetBIOS protocol, as the name implies, was originally implemented as BIOS code on a ROM chip for the IBM PC Network Broadband LAN. Today there are three variants of NetBIOS in use as session layer protocols, building on the LLC protocol (NetBEUI), the Novell IPX protocol, and TCP/IP. NetBIOS is described in RFC 1001 (Protocol Standard for a NetBIOS Service on a TCP/UDP Transport: Concepts and Methods) and RFC 1002 (Protocol Standard for a NetBIOS Service on a TCP/UDP Transport: Detailed Specifications).

The most significant difference between NetBIOS for IPX and UDP/TCP, on the one hand, and NetBEUI for the LLC protocol, on the other, is that the latter has no network layer and therefore cannot be routed. The NetBIOS protocol is primarily used for the

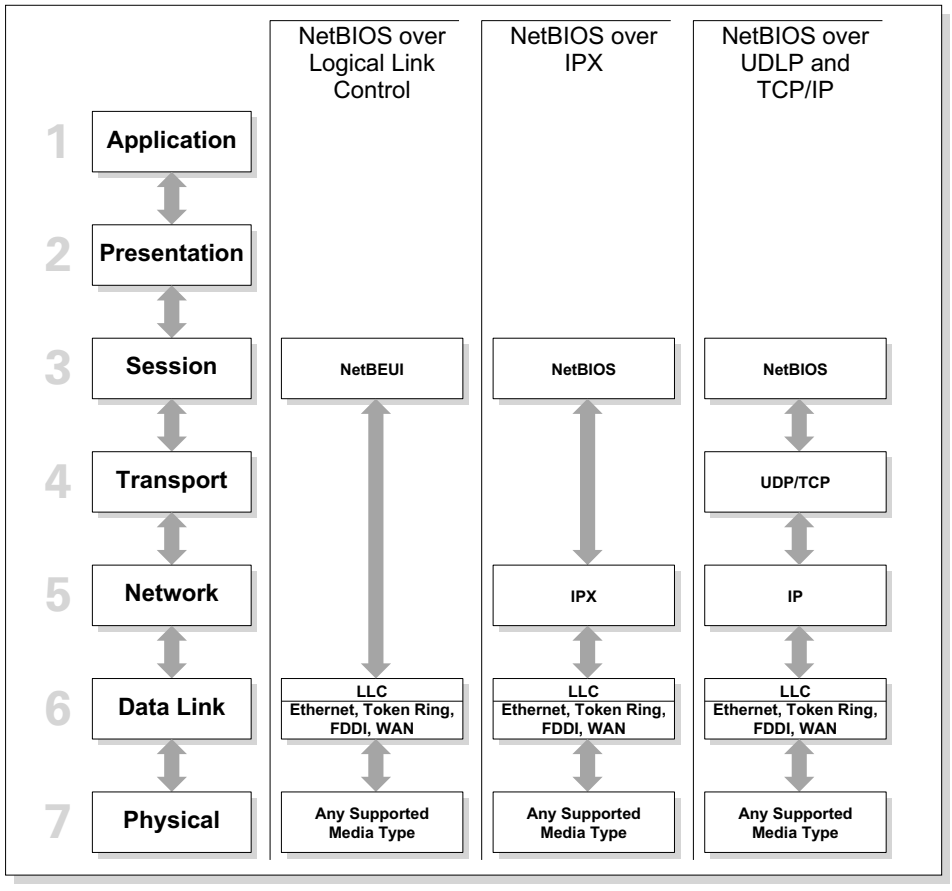

Figure 16.17 The NetBIOS protocols

location of resources in the network (similar to DNS) and the establishment of session layer connections.

The NetBIOS variant known as NetBIOS Extended User Interface (NetBEUI) is mainly used in Microsoft Windows environments. It is used to register new stations in the network and to log clients into server systems. NetBEUI is initialized using MAC multicast frames (MAC frame ID 030000000001 in Ethernet networks, C000000000080 in Token-Ring networks). Services implemented over NetBEUI include peer-to-peer file sharing in Windows 95/98 or server connections. A local session number (LSN) and a remote session number (RSN) identify every NetBIOS session over IPX.

In "Windows for Workgroups" environments that use NetBIOS over IPX as the session layer protocol, NetBIOS is mainly used for name services. The main differences from NetBEUI are the use of ordinary MAC broadcasts (FFFFFFFFFFFF) rather than NetBEUI's multicasts for initialization and the fact that the protocol is routable. Microsoft's extended implementation of NetBIOS over IPX is called Microsoft Name Management Protocol (NMPI).

NetBIOS over TCP/IP uses both TCP (ports 137–139) and UDP (ports 137–139). UDP is used for connectionless services, such as sending and receiving data packets, and

Unique (U):	The name may have only one IP address assigned to it. On a network device, multiple occurrences of a single name may appear to be registered. The suffix may be the only unique character in the name.
Group (G):	A normal group; the single name may exist with many IP addresses. The NetBIOS name server (WINS) responds to a name query on a group name with the limited broadcast address (255.255.255.255). Because routers block the transmission of these addresses, the Internet Group was designed to service communications between subnets.
Multihomed (M):	The name is unique, but due to multiple network interfaces on the same computer this configuration is necessary to permit the registration. The maximum number of addresses is 25.
Internet Group (I):	This is a special configuration of the group name used to manage Windows NT domain names.
Domain Name (D):	New in Windows NT 4.0.

Figure 16.18a NetBIOS name suffixes

Name	Number (h)	Type	Usage
<computername>	00	U	Workstation Service
<computername>	01	U	Messenger Service
<\\—__MSBROWSE__>	01	G	Master Browser
<computername>	03	U	Messenger Service
<computername>	06	U	RAS Server Service
<computername>	1F	U	NetDDE Service
<computername>	20	U	File Server Service
<computername>	21	U	RAS Client Service
<computername>	22	U	Microsoft Exchange
<computername>	23	U	Microsoft Exchange Store
<computername>	24	U	Microsoft Exchange
<computername>	30	U	Modem Sharing Server
<computername>	31	U	Modem Sharing Client
<computername>	43	U	SMS Clients Remote
<computername>	44	U	SMS Administrators
<computername>	45	U	SMS Clients Remote Chat
<computername>	46	U	SMS Clients Remote
<computername>	4C	U	DEC Pathworks TCPIP Service on Windows NT
<computername>	52	U	DEC Pathworks TCPIP
<computername>	87	U	Microsoft Exchange MTA
<computername>	6A	U	Microsoft Exchange IMC
<computername>	BE	U	Network Monitor Agent
<computername>	BF	U	Network Monitor
<username>	03	U	Messenger Service
<domain>	00	G	Domain Name
<domain>	1B	U	Domain Master Browser
<domain>	1C	G	Domain Controllers
<domain>	1D	U	Master Browser
<domain>	1E	G	Browser Service
<INet~Services>	1C	G	IIS
<IS~computer name>	00	U	IIS
<computername>	[2B]	U	Lotus Notes Server
IRISMULTICAST	[2F]	G	Lotus Notes
IRISNAMESERVER	[33]	G	Lotus Notes
Forte_$ND800ZA	[20]	U	DCA IrmaLan Gateway

Figure 16.18b NetBIOS name suffixes

for name services when no WINS (NetBIOS name server) is present. TCP is used to communicate with the NetBIOS name server and to set up session connections.

Microsoft restricts the length of NetBIOS names to a maximum of 15 characters, using a sixteenth character as a node type suffix (see Figure 16.18).

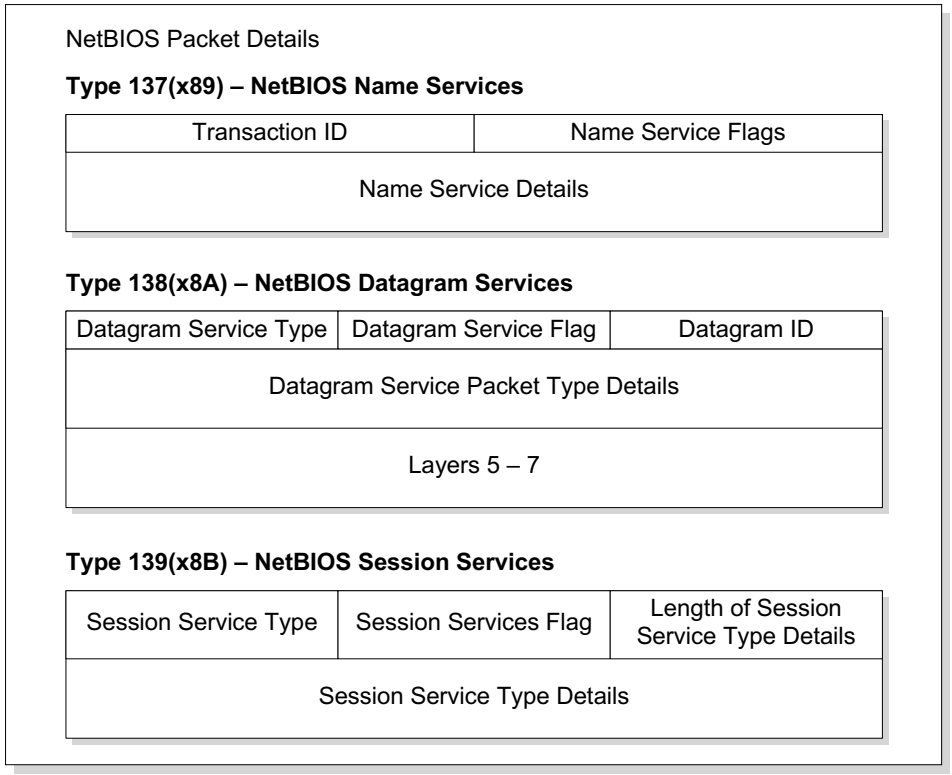

Figure 16.19 NetBIOS packet formats

16.8 The Dynamic Host Control Protocol (DHCP)

DHCP, RFC 2131, is used for dynamic assignment of IP addresses and subnet masks to new clients logging into the network. The key advantage over static, permanently assigned addresses and subnet masks is that the client can connect to any segment of an IP network without requiring manual configuration changes. DHCP automatically assigns the client an IP address and subnet mask, which are valid for the local

subnetwork. This is especially advantageous for diskless workstations and mobile clients (notebooks), as well as in the case of changes in network design. As recently as a few years ago, diskless workstations used the Reverse Address Resolution Protocol (RARP) to obtain an IP address. As with ARP, the main disadvantage of RARP is that it uses MAC layer broadcasts, which are not forwarded by routers. The use of RARP therefore requires a RARP server in every network segment. Moreover, RARP servers only hand out IP addresses, not subnet masks. In order to overcome the limitations of RARP, the Bootstrap Protocol (BOOTP) was developed. BOOTP distributes both IP addresses and subnet masks. However, BOOTP requires a static network environment in which a client with a given MAC address is always assigned the same IP address and subnet mask. DHCP, by contrast, permits true dynamic IP address and subnet mask assignments, and also provides a significantly larger vendor options field (312 bytes, compared with 64 bytes under BOOTP). The DHCP packet format is practically identical to the BOOTP format and is illustrated in Figure 16.20. UDP is used as the transport protocol with the reserved ports 67 (DHCP server) and 68 (DHCP client). A protocol analyzer can thus help in troubleshooting DHCP problems by filtering for UDP ports 67 and 68.

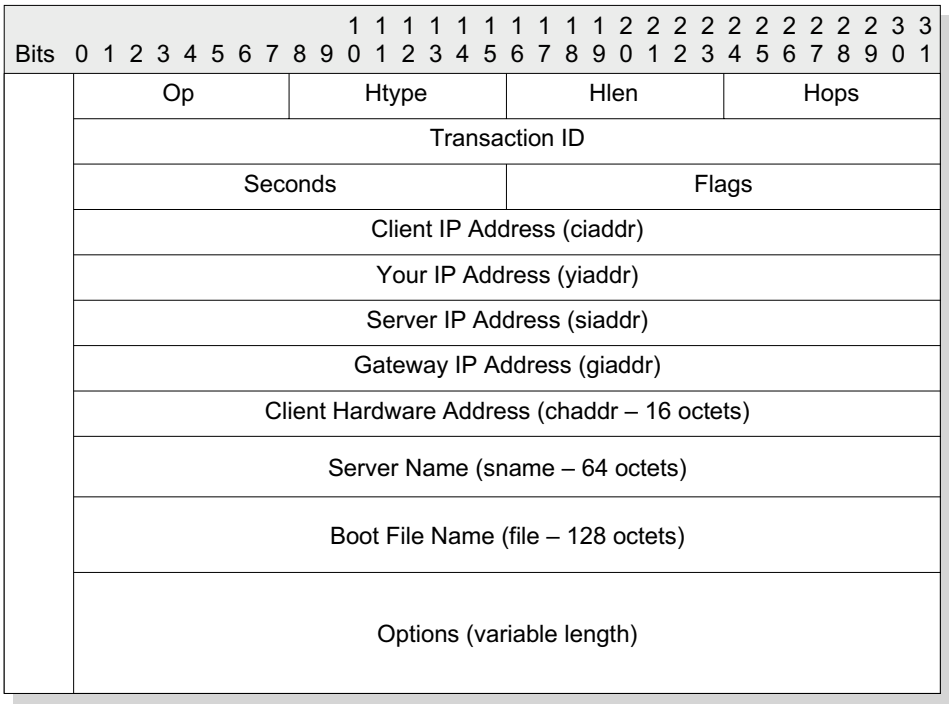

Figure 16.20 The DHCP packet format

16.8.1 Diagnosing DHCP Errors

The most common problems with DHCP arise through the mixture of dynamic and fixed IP addresses. For this reason it is extremely important that no fixed IP addresses be released to the DHCP server for distribution. Another type of problem occurs when DHCP servers are used to manage fixed addresses—that is, when a given IP address is always assigned to the same MAC address: if a station (a MAC address) is moved to a different segment, the DHCP server may assign it its old IP address even though the address is not valid in the new subnet.

16.9 Internet Protocols: Standards

The working groups of the Internet Engineering Task Force (IETF) develop the specifications for the Internet protocol family. The standardization documents, known as Requests for Comments (RFCs), are available from several repositories, including the InterNIC documentation archive at:

http://www.internic.net/.

The most important Internet standards are listed here:

IPoA MIB:	Classical IP and ARP Over ATM MIB, RFC 2320 (Proposed standard)
NETBIOS:	Protocol Standard for a NetBIOS Service on a TCP/UDP Transport, RFC 1001, 1002 (STD 19)
DNS:	Domain Names—Concepts and Facilities [port 53], RFC 1034 (STD 13)
DNS:	Domain Names—Implementation and Specification, RFC 1035 (STD 13)
IP-IEEE:	Transmission of IP over IEEE 802 Networks, RFC 1042 (STD 43)
RPC:	Remote Procedure Call Protocol Version 2 (Sun), RFC 1057
SLIP:	Transmission of IP over Serial Lines, RFC 1055 (STD 47)
RIP:	Routing Information Protocol, RFC 1058 (STD 34)
IP-DVMRP:	Distance Vector Multicast Routing Protocol, RFC 1075
IP-NETBIOS:	Transmission of IP over NetBIOS, RFC 1088 (STD 48)
NFS:	Network File System Protocol (Sun), RFC 1094
IGMP	(Internet Group Multicast Protocol): Host Extensions for IP Multicasting, RFC 1112 (STD 5)

IP-IPX:	Transmission of 802.2 Packets over IPX, RFC 1132 (STD 49)
IP-CMPRS:	Compressing TCP/IP Headers, RFC 1144
SNMP:	Simple Network Management Protocol [port: 161], RFC 1157 (STD 15)
BGP:	Border Gateway Protocol [port 179], RFC 1163
IP–X.121:	IP to X.121 Address Mapping for DDN, RFC 1236
ICMP-ROUT:	ICMP Router Discovery Messages, RFC 1256
TFTP:	Trivial File Transfer Protocol, RFC 1350
IP-X.25:	Multiprotocol Interconnect on X.25 and ISDN in the Packet Mode, RFC 1356
IP-FDDI:	Transmission of IP and ARP over FDDI Networks, RFC 1390
SNMPv2:	Version 2 of the Internet-standard Network Management Framework, RFC 1441ff.
IRCP:	Internet Relay Chat Protocol, RFC 1459
IP-TR-MC:	IP Multicast over Token Ring LANs, RFC 1469 (Proposed standard)
DHCP-BOOTP:	Interoperation Between DHCP and BOOTP, RFC 1534 (Draft standard)
PPP:	The Point-to-Point Protocol (PPP), RFC 1661 (STD 51)
RIP2-APP:	RIP Version 2 Protocol Applicability Statement, RFC 1722 (STD 57)
IPNG:	The Recommendation for the IP Next Generation Protocol, RFC 1752
RMON MIB:	Remote Network Monitoring Management Information Base, RFC 1757, 2021, 2074
ICMP-DM:	ICMP Domain Name Messages, RFC 1788
NFSV3:	NFS Version 3 Protocol Specification, RFC 1813
IPV6-AH:	IP Authentication Header, RFC 1826, 2402
RPC:	Remote Procedure Call Protocol Specification v2, RFC 1831

WHOIS++:	Architecture of the WHOIS++ Service, RFC 1835
IPV6:	Internet Protocol, Version 6 Specification, RFC 1883, 2460
IPV6-Addr:	IPv6 Addressing Architecture, RFC 1884
DNS-IPV6:	DNS Extensions to Support IP Version 6, RFC 1886
RTP:	A Transport Protocol for Real Time Applications, RFC 1889
POP3:	Post Office Protocol—Version 3 [port: 110], RFC 1939
HTTP-1.0:	Hypertext Transfer Protocol—HTTP/1.0, RFC 1945
IPV6-ETHER:	Transmission of IPv6 Packets Over Ethernet, RFC 1972, 2464
IPIP:	IP Encapsulation within IP, RFC 2003
IPV6-FDDI:	Transmission of IPv6 Packets Over FDDI, RFC 2019
IPV6-PPP:	IP Version 6 over PPP, RFC 2023, 2472
IMAPV4:	Internet Message Access Protocol—Version 4rev1, RFC 2060
DNS-SEC:	Domain Name System Security Extensions, RFC 2065
IP-HIPPI:	IP over HIPPI, RFC 2067
HTTP-1.1:	Hypertext Transfer Protocol—HTTP/1.1, RFC 2068
DHCP:	DHCP Options and BOOTP Vendor Extensions, RFC 2132
RADIUS:	Remote Authentication Dial In User Service [port: 1812], RFC 2138
RSVP:	Resource ReSerVation Protocol—Version 1, RFC 2205
IP-ATM:	Classical IP and ARP over ATM, RFC 2225
FTP-SECEXT:	FTP Security Extensions, RFC 2228
LDAP3:	Lightweight Directory Access Protocol (v3), RFC 2251
OSPF2:	Open Shortest Path First Version 2, RFC 2328 (STD 54)
NHRP:	NBMA Next Hop Resolution Protocol, RFC 2332
IARP:	Inverse Address Resolution Protocol, RFC 2390
IP-FR:	Multiprotocol Interconnect over Frame Relay, RFC 2427
IPV6-FDDI:	Transmission of IPv6 Packets Over FDDI, RFC 2467
UDP:	User Datagram Protocol, RFC 768 (STD 6)
IP:	Internet Protocol, RFC 791 (STD 5)

ICMP:	Internet Control Message Protocol, RFC 792 (STD 5)
TCP:	Transmission Control Protocol, RFC 793 (STD 7)
SMTP:	Simple Mail Transfer Protocol [port: 25], RFC 821 (STD 10)
ARP:	Ethernet Address Resolution Protocol, RFC 826 (STD 37)
TELNET:	Telnet Protocol Specification, RFC 854 (STD 8)
ECHO:	Echo Protocol [port 7], RFC 862 (STD 20)
DAYTIME:	Daytime Protocol [port 13], RFC 867 (STD 25)
TIME:	Time Server Protocol, RFC 868 (STD 26)
ICMPv6:	Internet Control Message Protocol for IPv6, RFC 1885
BOOTP:	Bootstrap Protocol, RFC 951
FTP:	File Transfer Protocol [port 21], RFC 959 (STD 9)
NNTP:	Network News Transfer Protocol [port: 119], RFC 977

16.10 Troubleshooting Internet Protocols

16.10.1 Gathering Information on Symptoms and Recent Changes

The first step in any troubleshooting process is to gather information. The more information you have about the symptoms and characteristics of a problem—including *when* it first occurred—the better your chances of solving the problem quickly and efficiently. Typical questions to ask at this stage include:

- Has any hardware or software network component been modified?
- Do the symptoms occur regularly or intermittently?
- When was the first occurrence of the symptom?
- Are the symptoms related to certain applications or connections, or do they affect all network operations?
- Has anyone connected or disconnected a PC (laptop or desktop) or any other component to or from the network?
- Has application software been installed or updated?
- Have system files been cloned from one network node to another?
- Has anyone installed an adapter card in a computer?
- Has any maintenance work been performed in the building recently (by a telephone company or building maintenance personnel, for example)?
- Has anyone (including cleaning personnel) moved any equipment or furniture?

16.10.2 Error Symptoms in IP Networks

As in every networking process, significant operating data for IP networks should be recorded before trouble arises. Such information includes a description of all IP network components (network stations, routers, switches) with their configurations, IP addresses, host and domain names, physical interfaces, and the protocols and applications that communicate through them. Furthermore, vital statistics such as capacity use (average and peak loads) should also be available. When errors arise, a comparison between current test data and the data recorded during normal operation often provides a first indication of the cause of trouble. If an unusually high number of ICMP packets, broadcasts, rerouting packets, or TCP timeouts is observed this can be considered a clue as to the source of problems. Modern protocol analyzers generate such protocol performance statistics automatically, thus simplifying diagnostic work significantly (see Figure 16.21).

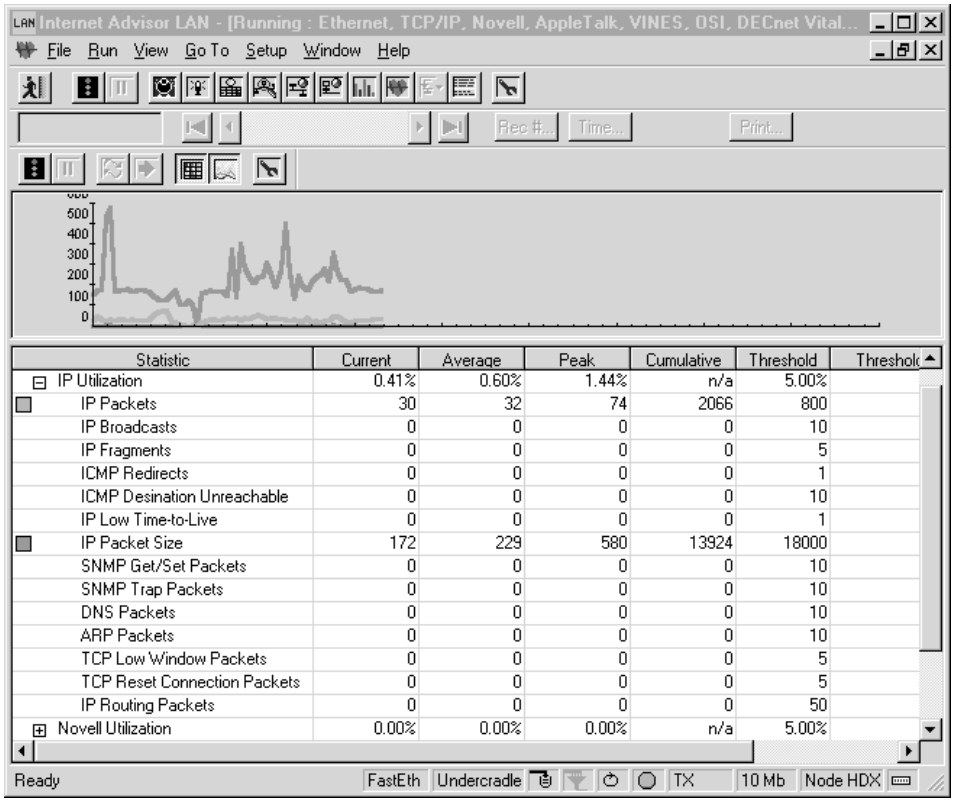

Figure 16.21 Testing IP performance with a protocol analyzer

The first step in diagnosing a problem is to determine the symptoms as exactly as possible. Is it still possible to initiate or accept connections? Do existing connections get interrupted? Are the response times higher or throughput lower for certain services or for all services?

If no connection can be established to one or more hosts in the network—even though standard hardware checks show that the station is connected to the network and its network adapter is correctly configured and operational—then the first step is to determine the extent of the error domain by sending pings. The ping command uses ICMP Echo Request and Echo Reply packets to test the availability and status of IP hosts. Sending pings in the following order can narrow down the error domain:

1) Ping the problem host's loopback address, 127.0.0.1.
2) Ping the problem host from another host in the same subnetwork.
3) Ping another host in the same subnetwork from the problem host.
4) Ping the default router from the problem host.
5) Ping another host in a different subnetwork from the problem host.
6) Ping the problem host by name.
7) Ping another host in the same subnetwork by name.
8) Ping another host in a different subnetwork by name.
9) Telnet to the problem host; telnet to the port number of the problem application on the problem host.

If the ping to the loopback address yields no response, then either the network adapter is defective or the TCP/IP driver is not correctly installed. Note, however, that in certain operating systems, such as UNIX, the loopback interface may function even when the physical interface to the network is defective! If the problem host does not answer a ping from another station in the same subnetwork, the sending host itself must first be tested for a functioning network connection. Then the IP address of the problem host should be verified.

If communication is successful within the subnetwork, the next step is to test the host's availability across a router. Begin by sending a ping from the problem host to the router. If the router does not answer, then the IP address and subnet mask of the problem host must be verified and the router's IP address and configuration must be checked. In order to forward an IP packet, the router must be aware of the destination network (Telnet to the router and retrieve IP route). If no specific route is configured for the network in question, then at least a default route (also called "default gateway") must be configured that leads to the destination network. Note: the router may also be configured to ignore pings (ICMP Type 0 or Echo Request packets) in order to prevent extra traffic due to such tests. If this is the case, the ping will yield no response even if the router is operational and its routing tables are correct.

If the ping test is successful between the problem host and the router, the next step is to test the connection from the problem host to hosts in neighboring networks that are connected through the router. Test destinations are systematically chosen in successively more remote networks until a network or router is identified to which a connection is no longer possible. The traceroute command (tracert in Windows 95 or NT) can be used to trace the packet's path and identify where the route breaks down. The traceroute program uses the ICMP Echo command to send a packet to the destination and back again. In addition to the response time, this command also counts the number of intermediate stations (hops) through which the packet is routed. Each "hop" reduces the packet's Time to Live (TTL) counter by one. Traceroute starts with sending an ICMP echo command using a TTL value of 1. Because the TTL is decremented by every router along the transmission path, this packet only reaches the first router, which decrements the TTL to zero, discards the packet, and sends back the ICMP message "Timeout Exceeded". The traceroute program notes the source address of this message and the delay time, then sends another packet, this time with a TTL of two. The first router reduces the TTL to one as it forwards the packet, and this time the second router sets the TTL to zero and returns a Timeout Exceeded message. This process continues, allowing the traceroute program to record a list of all routers along the path to the packet's destination address. Finally, when a packet with a sufficient TTL reaches the destination, the host responds with an "Unreachable Port" message, and the traceroute procedure is finished. Note, however, that there are routers that do not respond in this way to traceroute packets! In rare cases traceroute may display only a partial path. In addition, depending on the route the ICMP Echo commands are taking, results may differ. Because the traceroute function cannot detect route changes, the traceroute command should be used at least three times in a row to obtain reliable results.

If the traceroute test is successful, then the address resolution must be tested by sending a ping to the problem host using its host and domain name rather than its numeric IP address. If the host does not respond, its configuration and the DNS server's entries must be inspected for correct name/IP address mapping. (A ping to the name server itself indicates whether it is running and connected to the network.) Then a ping by name can be performed in the opposite direction, from the problem host to other hosts within the local subnetwork and in other subnets.

If the description of the symptoms indicates intermittent or periodical interruptions in availability, then the cause is often found in an overloaded network or destination host. Here again the traceroute command can be used to determine whether the destination itself is unavailable or whether the data packets are being lost in an overloaded transit network. Another frequent cause of intermittent availability is

another station using the same IP address as the destination host. So long as the other station is active, no IP connection to the actual owner of the IP address can be established. Such duplicate IP addresses can be quickly identified using a protocol analyzer, which sends ARP (Address Resolution Protocol) messages and monitors the resulting ARP responses. If an ARP request evokes two ARP reply packets with different hardware addresses, then the two stations with these addresses are using the same IP address. Finally, if the intermittent connection problems are observed in conjunction with frequent TCP connection timeouts, this may indicate a defective network component such as a bridge or router. In normal operation, less than one percent of network connections should be lost due to TCP timeouts. A protocol analyzer can be used to generate exact statistics on the number of TCP connection timeouts. If an IP connection between two hosts is possible, but certain applications are still unable to communicate over it, then the cause can usually be found in the router access tables. These are lists in which the network administrator specifies the permitted services (by port number). TCP and UDP packets addressed to other ports are blocked. Thus Telnet sessions may be possible while all other network services are blocked between two hosts. For this reason, the next troubleshooting step, if no problems are found at the IP level, is to test the TCP layer connection and the availability of the port addresses that correspond to the desired applications. The simplest way to determine whether a TCP connection can be established is to attempt a Telnet session with the problem host. Even if the Telnet program reports an error such as:

telnet: Unable to connect to remote host

telnet: Connection refused

this is sufficient indication that the TCP/IP stack is working. If a Telnet connection is possible, then at least the corresponding TCP port on the host is available. This is a more reliable indication than the ping command because ping packets can be filtered out along the path between the hosts, and in any case do not indicate the availability of specific applications.

Another common problem in IP networks is "local routing". This phenomenon occurs when a switch is connected to a router. In order to reach a segment that is connected to the same switch, a packet first has to be switched to the router, is routed within the router, and then transferred through the same switch again to the destination segment. Obviously this process doubles the number of packets passed through the switch and consumes buffer and CPU resources in the router. If a protocol analyzer is placed between the switch and the router, the local routing phenomenon can easily be detected by filtering on a station pair and looking for packets with identical TCP sequence numbers. If one of those packets contains the router address as destination address and the second one the router address as source address, and if the TTL value of packet two is decremented by one, a local routed communication path is found.

Another scenario that can cause local routing is when a local segment has two different active subnets at the same time. In this case the router will forward the frames to the same segment, again causing unnecessary resource consumption within the router.

16.10.3 Symptoms and Causes: Internet Protocols

This section lists the typical causes of trouble in IP networks, grouped by the symptoms observed.

Symptom: Ping from Problem Host to Loopback Address 127.0.0.1 Fails

Cause (1): TCP/IP driver or interface card software is not installed properly.
Cause (2): Faulty network interface.

Symptom: Ping to Problem Host from a Node in the Same Subnet Fails

Cause (1): Problem host not connected to the network.
Cause (2): Incorrectly configured interface card.
Cause (3): Wrong IP address; incorrect subnet mask.

Symptom: Ping from Problem Host to a Host in the Same Subnet Fails

Cause (1): Problem host not connected to the network.
Cause (2): Destination node not active.
Cause (3): Incorrect IP destination address used in the ping command.

Symptom: Ping from Problem Host to Default Router Fails

Cause (1): Problem host not connected to the network.
Cause (2): IP address or subnet mask of the problem host is incorrect.
Cause (3): Incorrect IP address of the router used in the ping command.
Cause (4): Incorrectly configured default route.
Cause (5): Inactive router port.
Cause (6): DHCP problem. DHCP server used to manage fixed addresses (a given IP address is always assigned to the same MAC address) and station moved to a different segment. The DHCP server assigns old IP address even though the address is not valid in the new subnet.

Symptom: Ping from Problem Host to Host Outside the Local Subnet Fails

Cause (1): Incorrectly configured default gateway on the router.
Cause (2): No default gateway configured on the router.
Cause (3): Remote host inactive.
Cause (4): Router port to destination subnet inactive.
Cause (5): Router access list incomplete; does not reflect route updates.
Cause (6): Incorrecmt router filter activated.

Symptom: Ping to Problem Host by Name Fails

Cause (1): Problem host not connected to the network.
Cause (2): The host name « IP address mapping at the problem host is incorrect.
Cause (3): The IP address of the DNS server is incorrectly configured.
Cause (4): DNS server is inactive.
Cause (5): The table of the host sending the ping contains incorrect host name « IP address mapping.

Symptom: Ping by Name from Problem Host to Host in the Same Subnet Fails

Cause (1): Incorrect host/domain name entered in the ping command.
Cause (2): Remote host not active.
Cause (3): IP address of the DNS server incorrectly configured on the problem host.
Cause (4): DNS server inactive.
Cause (5): The table of the problem host contains incorrect host name « IP address mapping for the host sending the ping.

Symptom: Ping by Name to a Host Outside the Local Subnet from the Problem Host Fails

Cause (1): Incorrect host name entered in the ping command.
Cause (2): Remote host not active.
Cause (3): IP address of the DNS server incorrectly configured on the problem host.
Cause (4): DNS server inactive.
Cause (5): The table of the problem host contains incorrect host name « IP address mapping for the remote host.
Cause (6): Incorrect IP address of default route in the router table.

Symptom: Intermittent Loss of Connection

Cause (1): Overloaded destination network or host.

Cause (2): Duplicate IP address: another node is using the destination's IP address.
Cause (3): TCP connections timing out (due to bridge or router problems, for example).
Cause (4): Mixture of dynamic (DHCP based) and static assigned IP addresses.

Symptom: IP Connection Working but Application not Available

Cause (1): Router access list configured to block the application port.
Cause (2): Application not active.

Symptom: Diskless Stations Unable to Boot

Cause (1): Router blocking UDP broadcast forwarding: BOOTP broadcasts not transmitted.
Cause (2): Incorrectly configured router filter blocking BOOTP packets.

Symptom: Generally Low TCP/IP Performance

Cause (1): Poorly configured DNS client: client waits until the first DNS request times out before trying a second DNS server.
Cause (2): DNS server not configured for reverse lookup; reverse lookup requests not being handled.
Cause (3): DNS table incomplete: does not contain IP address <> name mappings for all hosts.
Cause (4): Local routing. A switch is connected to a router. In order to reach a segment that is connected to the same switch, a packet first has to be switched to the router, is routed within the router, and then transferred through the same switch again to the destination segment. The number of packets that are being passed through the switch are doubled and additional buffer and CPU resources in the router are consumed.
Cause (5): Overloaded networks in the path the TCP connection is using.

Symptom: Excessive TCP Retransmissions

Cause (1): Sent TCP packet or a returning acknowledgement packet lost by an overloaded switch or router.
Cause (2): Packet corrupted during the transmission contains a CRC error.
Cause (3): TCP data corrupted, shows a TCP checksum error.
Cause (4): Fragment of a fragmented TCP packet lost or corrupted.
Cause (5): Buffer overflow at the receiver.
Cause (6): Acknowledgement for a TCP packet sequence is too slow and sender retransmits.

Symptom: TCP Window Size too Small

Cause (1): Default IP MTU (Maximum Transfer Unit) changed by user.
Cause (2): IP MTU and receive TCP buffer size changed by Internet optimizing application.

Symptom: TCP Window Size Drops Below Maximum

Cause: Application not loading data off the TCP stack fast enough. Likely causes are performance problems on the server or the client. (If the node is a printer, however, this is a normal scenario.)

Symptom: TCP Window Size Constant, Slow Acknowledgement Packets

Cause (1): Overloaded network, routers, or switches.
Cause (2): Dial-up or WAN connection overloaded.

Symptom: TCP Window Size too Large

Cause: In some cases a large TCP window size in combination with a small amount of available interface memory and a slow hard drive can cause performance problems. Communication then has to stop until the TCP segment (for example, a large 64 Kbyte TCP segment) is stored and the data transfer process can resume.

Symptom: OSPF Routing Problems

Cause (1): The Hello and Dead timers of the various routers in the network not coordinated.
Cause (2): IGRP or RIP route distribution incorrectly configured in OSPF.
Cause (3): Incorrectly configured virtual link.
Cause (4): No router port assigned an IP address (OSPF uses IP address as ID).

Symptom: RIP Problems

Cause (1): Incorrect routing tables because of routing information retrieved from wrong interface or protocol.
Cause (2): Incorrect router filter configuration.
Cause (3): Subnet masks of router and host do not match.

The following listing summarizes the most frequent error sources for problems in TCP/IP environments (in alphabetical order):

- Application is not active
- Destination host is not active
- DNS server is not active
- DNS server not configured for reverse lookup. Reverse lookup request cannot be handled
- DNS table incomplete, does not contain IP domain name mappings for all hosts
- Duplicate IP address: a second node is using the destination host's IP address
- Faulty network interface
- Host name or IP address configuration of problem host is incorrect
- Inactive router port
- Incorrect destination IP address in the ping command
- Incorrect IP address or subnet address mask on problem host
- Incorrect router filter activated
- Incorrect routing tables due to routing information retrieved from wrong interface or protocol
- Incorrect terminal settings
- Incorrect user ID or password
- Incorrectly configured default gateway on the router
- Incorrectly configured interface card
- Incorrectly configured router filter blocking BOOTP frames
- Incorrectly configured virtual links
- IP address of the default route in the router table is incorrect
- IP address of the DNS server is incorrectly configured on the problem host
- IP address or subnet mask of the problem host is incorrect
- No default gateway configured on the router
- No router port has an IP address assigned (OSPF uses IP address as ID)
- Overloaded destination network or host
- Poorly configured DNS client: client waits until the first DNS request times out before trying a second DNS server
- Problem host not connected to the network
- Remote host not reachable via network
- Route distribution of IGRP or RIP is incorrectly configured in OSPF

Figure 16.22a The most common error causes in IP networks

- Router is blocking UDP broadcast forwarding: BOOTP broadcasts cannot be transmitted
- Router port to destination subnet inactive
- Subnet masks of router and host do not match
- TCP connections timing out (due to bridge or router problems, for example)
- TCP/IP software or interface card driver is not installed properly
- The Hello and Dead timers of the various routers in the network are not coordinated
- The hosts table on the host sending the ping contains an incorrect entry for the problem host (incorrect host name IP address mapping)
- The hosts table on the problem host contains an incorrect entry for the host sending the ping (incorrect host name IP address mapping)
- Router access list incomplete: does not contain route updates
- Router access list is configured to block the application port

Figure 16.22b The most common error causes in IP networks