Chapter 6

6.1 Public and Private IP Addresses

The stability of the Internet depends directly on the uniqueness of publicly used network addresses. In Figure 1 Required Unique Addresses, there is an issue with the network addressing scheme. In looking at the networks, both have a network address of 198.150.11.0. The router in this illustration will not be able to forward the data packets correctly. Duplicate network IP addresses prevent the router from performing its job of best path selection. Unique addresses are required for each device on a network.

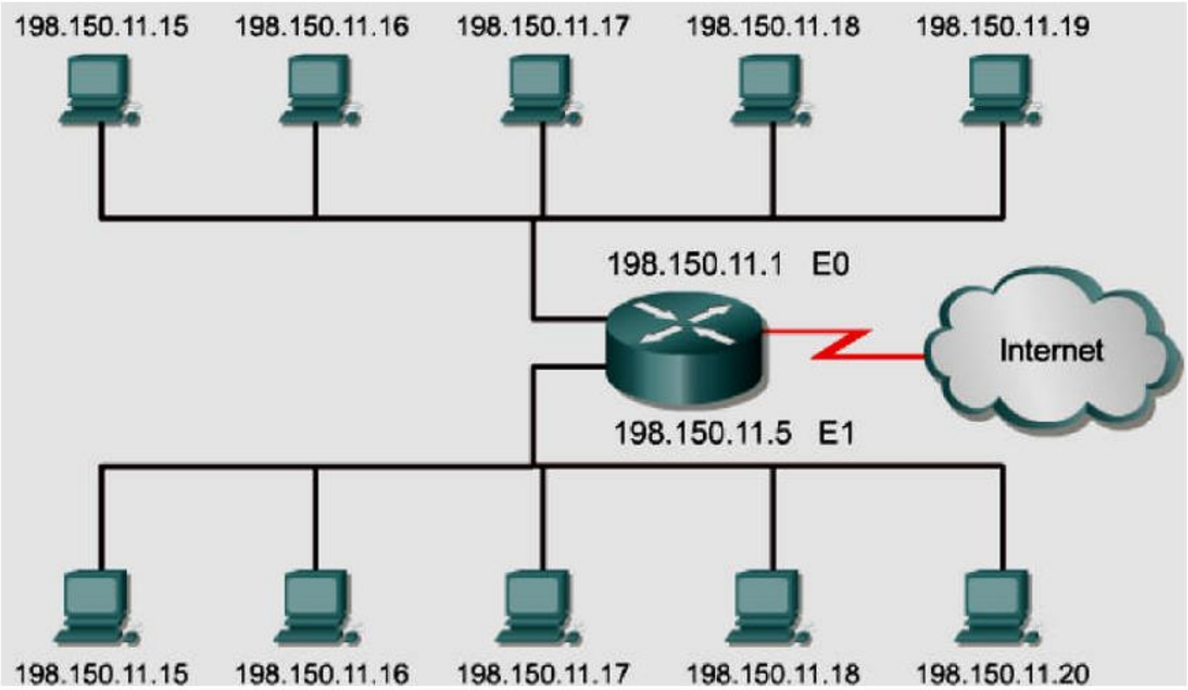


Figure 1 Required Unique Addresses

A procedure was needed to make sure that addresses were in fact unique. Originally, an organization known as the Internet Network Information Center (InterNIC) handled this procedure. InterNIC no longer exists and has been succeeded by the

Internet Assigned Numbers Authority (IANA). IANA carefully manages the remaining supply of IP addresses to ensure that duplication of publicly used addresses does not occur. Duplication would cause instability in the Internet and compromise its ability to deliver datagram's to networks. Public IP addresses are unique. No two machines that connect to a public network can have the same IP address because public IP addresses are global and standardized. All machines connected to the Internet agree to conform to the system. Public IP addresses must be obtained from an Internet service provider (ISP) or a registry at some expense. With the rapid growth of the Internet, public IP addresses were beginning to run out.

Table 1 Private IP Addresses

Class	Invisible Ranges (non routed addresses)	abbreviated
A	**10.0.0.1 – 10.255.255.254 ≅ 16000000**	**10.0.0.0 / 8**
B	**172.16.0.1 – 172.31.255.254 ≅ 10000000**	**172.16.0.0 / 12**
C	**192.168.0.1 – 192.168.255.254 ≅ 65000**	**192.168.0.0 / 16**

Private IP addresses are another solution to the problem of the impending exhaustion of public IP addresses. As mentioned, public networks require hosts to have unique IP addresses. However, private networks that are not connected to the Internet may use any host addresses, as long as each host within the private network is unique. Many private networks exist alongside public networks. However, a private network using just any address is strongly discouraged because that network might eventually be connected to the Internet. RFC 1918 sets aside three blocks of IP addresses for private, internal use.

These three blocks consist of one Class A, a range of Class B addresses, and a range of Class C addresses. Addresses that fall within these ranges are not routed on the Internet backbone. Internet routers immediately discard private addresses. If addressing a nonpublic intranet, a test lab, or a home network, these private addresses can be used instead of globally unique addresses. Private IP addresses can be intermixed, as shown in the graphic, with public IP addresses. This will conserve the number of addresses used for internal connections.

6.2 IPv4 Versus IPv6

The TCP/IP is sustaining a global network of information, commerce, and entertainment. IP Version 4 (IPv4) offered an addressing strategy that, although scalable for a time, resulted in an inefficient allocation of addresses. Unfortunately, Class C addresses are limited to 254 usable hosts. This does not meet the needs of larger organizations that cannot acquire a Class A or B address. Even if there were more Class A, B, and C addresses, too many network addresses would cause Internet routers to come to a stop under the burden of the enormous size of routing tables required to store the routes to reach each of the network Over the past two decades, numerous extensions to IPv4 have been developed. These extensions are specifically designed to improve the efficiency

with which the 32-bit address space can be used. Two of the more important of these are subnet masks and classless interdomain routing (CIDR). Meanwhile, an even more extendible and scalable version of IP, IP Version 6 (IPv6), has been defined and developed. IPv6 uses 128 bits rather than the 32 bits currently used in IPv4. IPv6 uses hexadecimal numbers to represent the 128 bits. IPv6 provides 640 six trillion addresses. This version of IP should provide enough addresses for future communication needs.

Internet Protocol Version 4 (Ipv4) 4 octets

11010001	10011100	11001001	01110001
209.	156.	201.	113

2^{32}=4,294,967,295 IP addresses (Approx.):4.3 billion

4,300,000,000

Internet Protocol Version 6 (Ipv4) 16 octets

10100101.00100100	01110010.11010011	00101100.10000000	11011101.00000010
A524:	72D3:	2C80:	DD02:
00000000.00101001	11101100.01111010	00000000.00101011	11101010.01110011
0029:	EC7A:	002B:	EA73

2^{128}=3.4x1038 IP addresses (approx.):340 Undecillion

340,000,000,000,000,000,000,000,000,000,000,000,000

Figure 2 Ipv4 and Ipv6 Addresses

IPv4 addresses are 32 bits long, written in decimal form, and separated by periods. IPv6 addresses are 128-bits long and are identifiers for individual interfaces and sets of interfaces. IPv6 addresses are assigned to interfaces, not nodes. Since each interface belongs to a single node, any of the unicast addresses assigned to the interfaces of the node may be used as an identifier for the node. IPv6 addresses are written in hexadecimal, and separated by colons. IPv6 fields are 16 bits long. To make the addresses easier to read, leading zeros can be omitted from each field. The field: 0003: is written: 3:. I Pv6 shorthand representation of the 128 bits use eight 16-bit numbers, shown as four hexadecimal digits.

6.3 Obtaining an Internet Address

A network host needs to obtain a globally unique address in order to function on the Internet. The physical or MAC address that a host has is only locally significant, identifying the host within the local area network. Since this is a Layer 2 address, the router does not use it to forward outside the LAN. IP addresses are the most commonly used addresses for Internet communications. This protocol is a hierarchical addressing scheme that allows individual addresses to be associated together and treated as groups. These groups of addresses allow efficient transfer of data across the Internet.

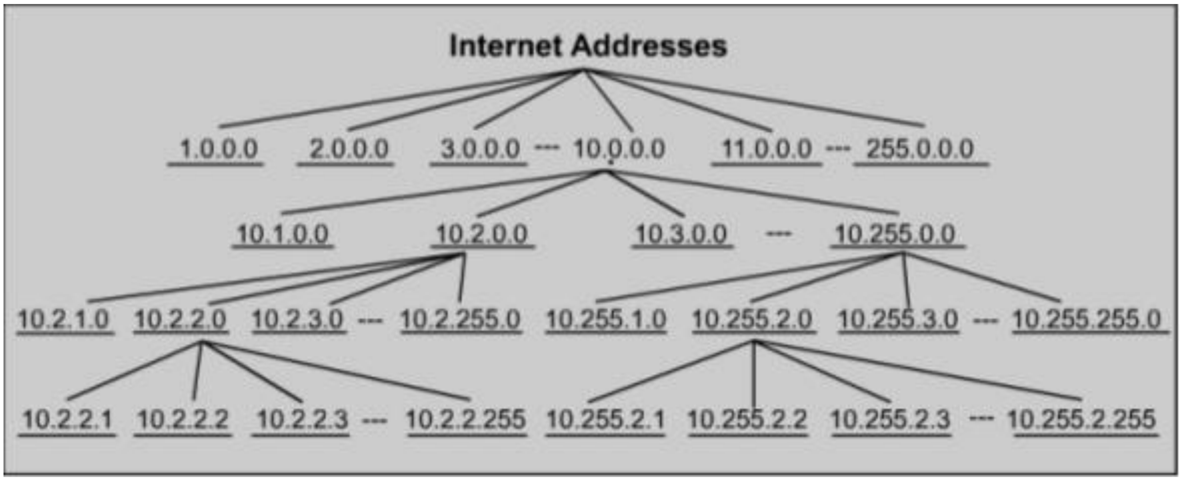


Figure 3 Internet Addresses

Network administrators use two methods to assign IP addresses. These methods are static and dynamic. Later in this lesson, static addressing and three variations of dynamic addressing will be covered. Regardless of which addressing scheme is chosen, no two interfaces can have the same IP address. Two hosts that have the same IP address could create a conflict that might cause both of the hosts involved not to operate properly. As shown in Figure 4 Assigning IP Addresses, the hosts have a physical address by having a network interface card that allows connection to the physical medium. The figure will focus on static IP address assignments.

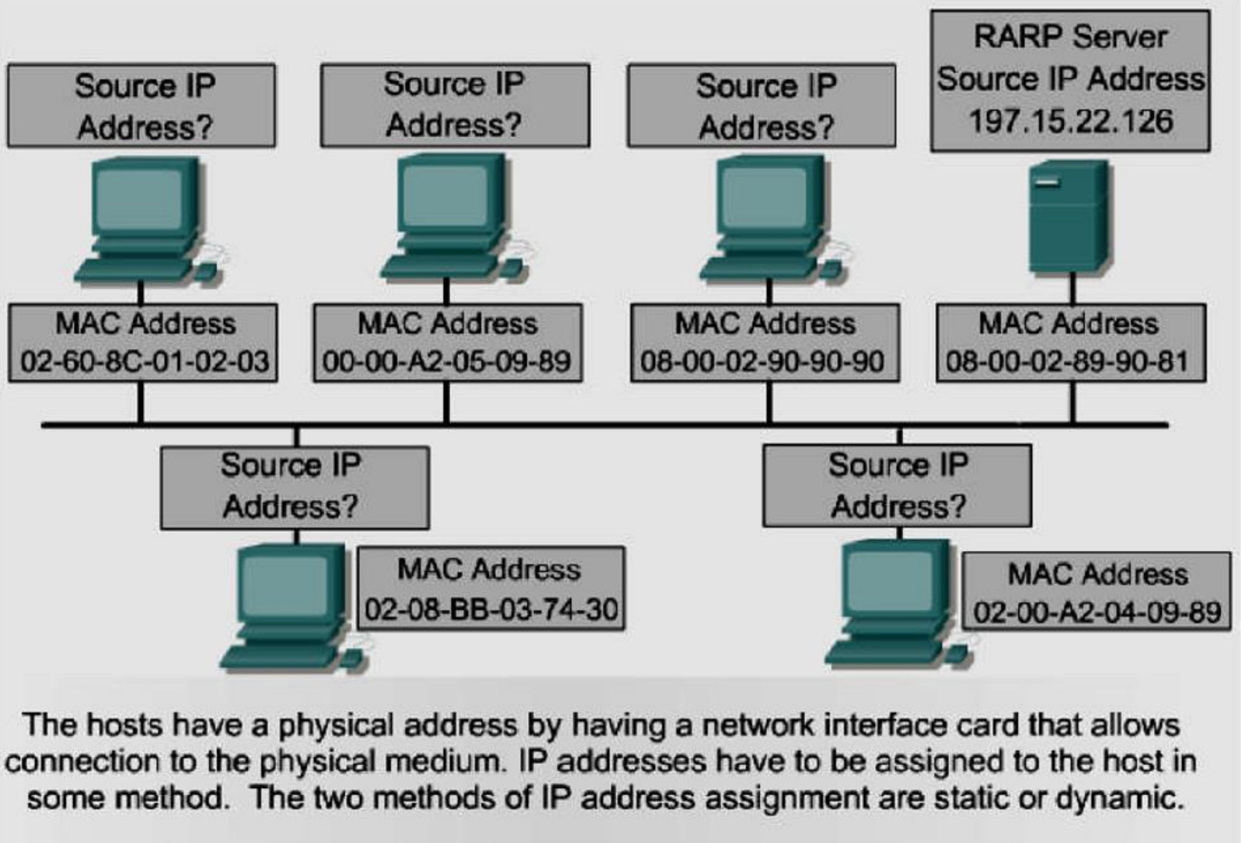
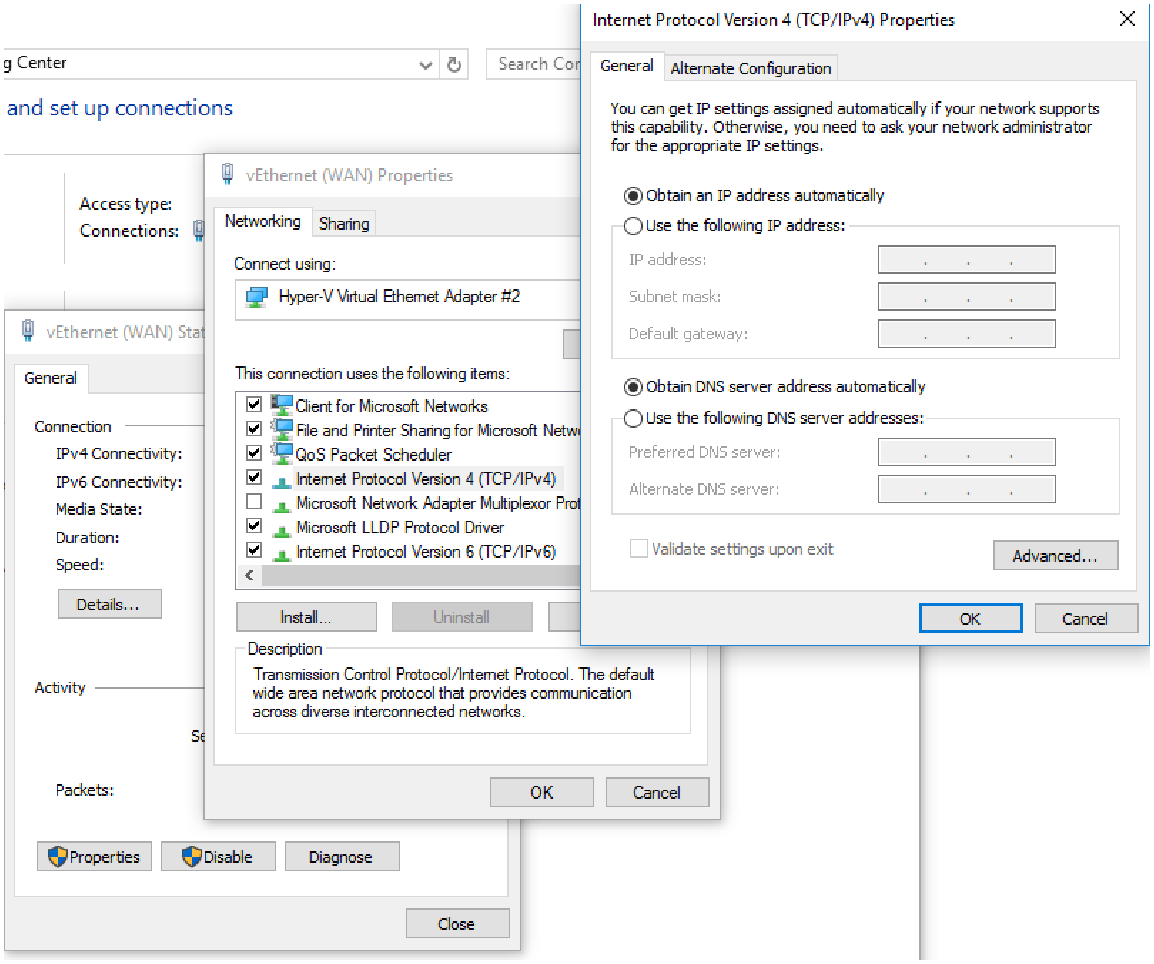

Source IP Address?
Source IP Address?
Source IP Address?
RARP Server
Source IP Address
197.15.22.126
MAC Address 02-60-8C-01-02-03
MAC Address 00-00-A2-05-09-89
MAC Address 08-00-02-90-90-90
MAC Address 08-00-02-89-90-81
Source IP Address?
Source IP Address?
MAC Address 02-08-BB-03-74-30
MAC Address 02-00-A2-04-09-89
The hosts have a physical address by having a network interface card that allows connection to the physical medium. IP addresses have to be assigned to the host in some method. The two methods of IP address assignment are static or dynamic.

Figure 4 Assigning IP Addresses

6.4 Static Assignment of an IP

Address Static assignment works best on small, infrequently changing networks. The system administrator manually assigns and tracks IP addresses for each computer, printer, or server on the intranet. Good recordkeeping is critical to prevent problems which occur with duplicate IP addresses. This is possible only when there are a small number of devices to track.



vEthernet (WAN) Properties

Connect using:

Hyper-V Virtual Ethernet Adapter #2

This connection uses the following items:

☑ Client for Microsoft Networks
☑ File and Printer Sharing for Microsoft Netw
☑ QoS Packet Scheduler
☑ Internet Protocol Version 4 (TCP/IPv4)
□ Microsoft Network Adapter Multiplexor Prot
☑ Microsoft LLDP Protocol Driver
☑ Internet Protocol Version 6 (TCP/IPv6)

Description

Transmission Control Protocol/Internet Protocol. The default wide area network protocol that provides communication across diverse interconnected networks.

Internet Protocol Version 4 (TCP/IPv4) Properties

You can get IP settings assigned automatically if your network supports this capability. Otherwise, you need to ask your network administrator for the appropriate IP settings.

● Obtain an IP address automatically
○ Use the following IP address:
IP address: . . .
Subnet mask: . . .
Default gateway: . . .

● Obtain DNS server address automatically
○ Use the following DNS server addresses:
Preferred DNS server: . . .
Alternate DNS server: . . .

□ Validate settings upon exit

Figure 5 TCP/IP Configuration for Windows

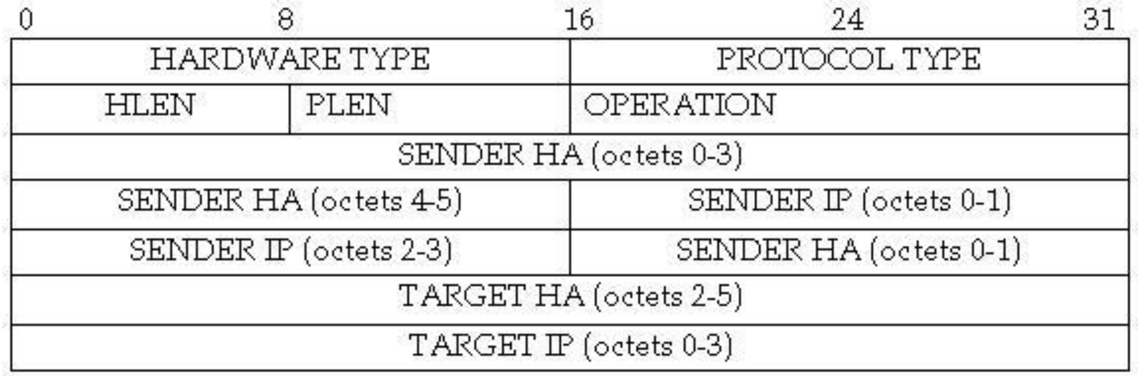
Servers should be assigned a static IP address so workstations and other devices will always know how to access needed services. Consider how difficult it would be to phone a business that changed its phone number every day. Other devices that should be assigned static IP addresses are network printers, application servers, and routers.

6.5 RARP IP Address Assignment

Reverse Address Resolution Protocol (RARP) associates a known MAC addresses with an IP addresses. This association allows network devices to encapsulate data before sending the data out on the network. A network device, such as a diskless workstation, might know its MAC address but not its IP address. RARP allows the device to make a request to learn its IP address. Devices using RARP require that a RARP server be present on the network to answer RARP requests.

Consider an example where a source device wants to send data to another device. In this example, the source device knows its own MAC address but is unable to locate its own IP address in the ARP table. The source device must include both its MAC address and IP address in order for the destination device to retrieve data, pass it to higher layers of the OSI model, and respond to the originating device. Therefore, the source initiates a process called a RARP request. This request helps the source device detect its own IP address.

RARP requests are broadcast onto the LAN and are responded to by the RARP server which is usually a router. RARP uses the same packet format as ARP. However, in a RARP request, the MAC headers and operation code are different from an ARP request.



<table>
<tr><th>0</th><th>8</th><th>16</th><th>24 31</th></tr>
<tr><td colspan="2">HARDWARE TYPE</td><td colspan="2">PROTOCOL TYPE</td></tr>
<tr><td>HLEN</td><td>PLEN</td><td colspan="2">OPERATION</td></tr>
<tr><td colspan="4">SENDER HA (octets 0-3)</td></tr>
<tr><td colspan="2">SENDER HA (octets 4-5)</td><td colspan="2">SENDER IP (octets 0-1)</td></tr>
<tr><td colspan="2">SENDER IP (octets 2-3)</td><td colspan="2">SENDER HA (octets 0-1)</td></tr>
<tr><td colspan="4">TARGET HA (octets 2-5)</td></tr>
<tr><td colspan="4">TARGET IP (octets 0-3)</td></tr>
</table>

Figure 6 ARP/RARP Message Structure

The RARP packet format contains places for MAC addresses of both the destination and source devices. The source IP address field is empty. The broadcast goes to all devices on the network. Figure 8 Figure 9, and depict the destination

MAC address as FF:FF:FF:FF:FF:FF. Workstations running RARP have codes in ROM that direct them to start the RARP process. A step-by-step layout of the RARP process is illustrated in Figure 7 through Figure 14.

- *Hardware Type*: (*16-bits*) - the type of interface the sender seeks an answer for.
- *Protocol Type*: (*16-bits*) - the high-level software address type provided.
- *HLEN*: (*8-bits*) – Hardware address length.
- *PLEN*: (*8-bits*) - Protocol address length.
- *OPERATION*: (*16-bits*) - the specific type of operation requested.
    - 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 HA*: (*6-octets*) - the sender’s actual hardware address, scalable up to six bytes.
- *SENDER IP*: (*4-octets*) - the sender’s IP address, always 32-bits.
- *TARGET HA*: (*6-octets*) - the destination node’s hardware address, scalable up to six bytes.
- *TARGET IP*: (*4-octets*) - the destination node’s IP address, always 32-bits.

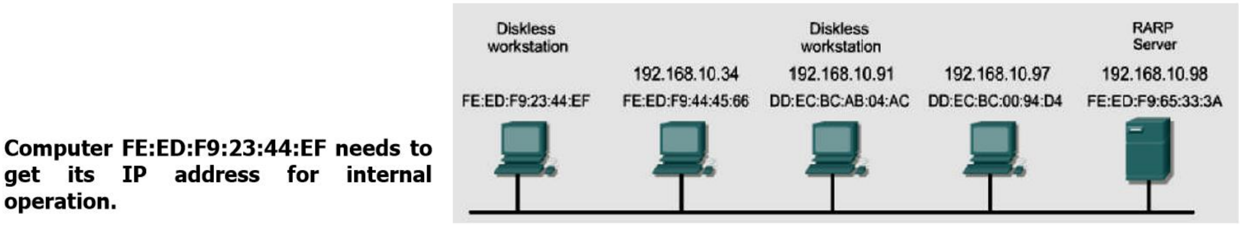
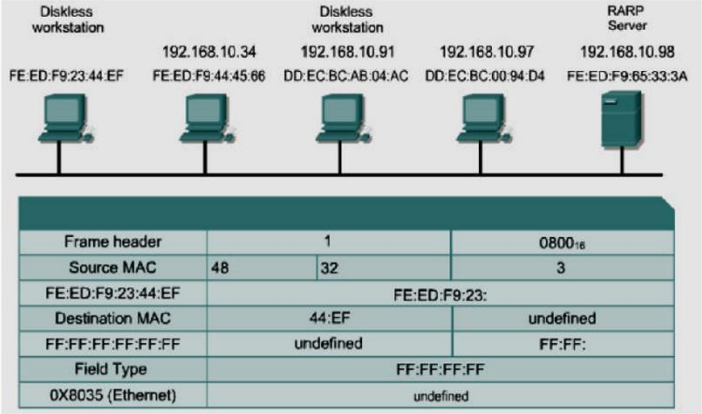

Computer FE:ED:F9:23:44:EF needs to get its IP address for internal operation.

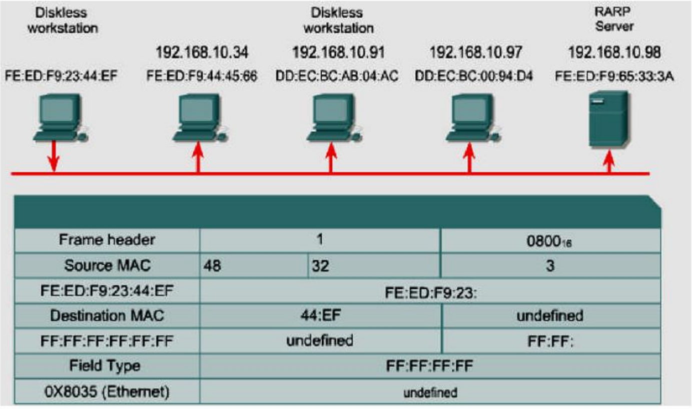
Figure 7 RARP: Network Segment



Frame header	1		0800_{16}
Source MAC	48	32	3
FE:ED:F9:23:44:EF	FE:ED:F9:23:		
Destination MAC	44:EF		undefined
FF:FF:FF:FF:FF:FF	undefined		FF:FF:
Field Type	FF:FF:FF:FF		
0X8035 (Ethernet)	undefined		

Computer FE:ED:F9:23:44:EF generates a RARP request.

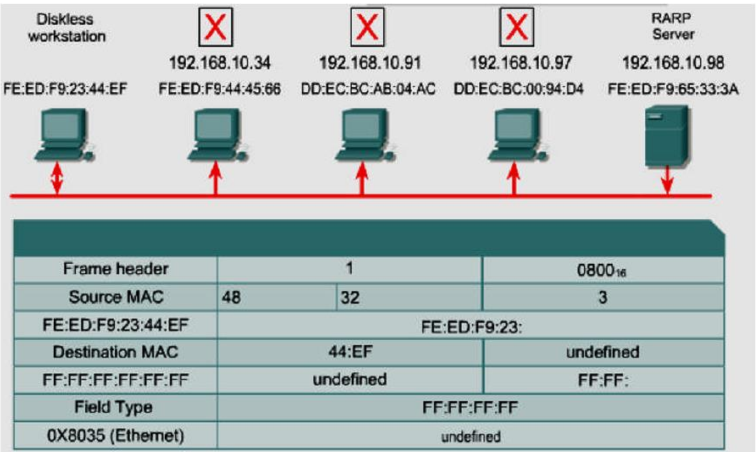
Figure 8 RARP: Request Generation



Frame header	1		0800_{16}
Source MAC	48	32	3
FE:ED:F9:23:44:EF	FE:ED:F9:23:		
Destination MAC	44:EF		undefined
FF:FF:FF:FF:FF:FF	undefined		FF:FF:
Field Type	FF:FF:FF:FF		
0X8035 (Ethernet)	undefined		

Computer FE:ED:F9:23:44:EF transmits RARP request.

Figure 9 RARP: Request Transmission



Frame header	1		0800_{16}
Source MAC	48	32	3
FE:ED:F9:23:44:EF	FE:ED:F9:23:		
Destination MAC	44:EF		undefined
FF:FF:FF:FF:FF:FF	undefined		FF:FF:
Field Type	FF:FF:FF:FF		
0X8035 (Ethernet)	undefined		

All computers pass the packet up to the network layer. If IP numbers do not match, the packet is discarded except for the RARP server, which detects the RARP request field.

Figure 10 RARP: Request Verification

The RARP server creates a RARP reply message for the requesting client.

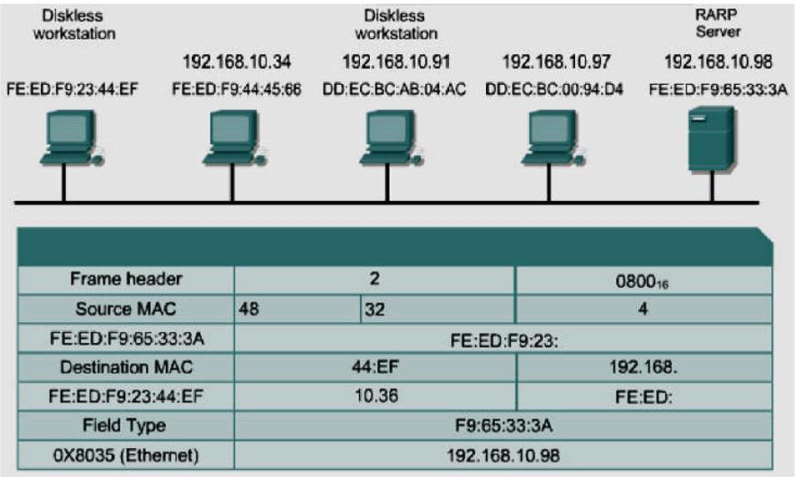


Figure 11 RARP: Reply Generation

All computers copy the frame and examine it.

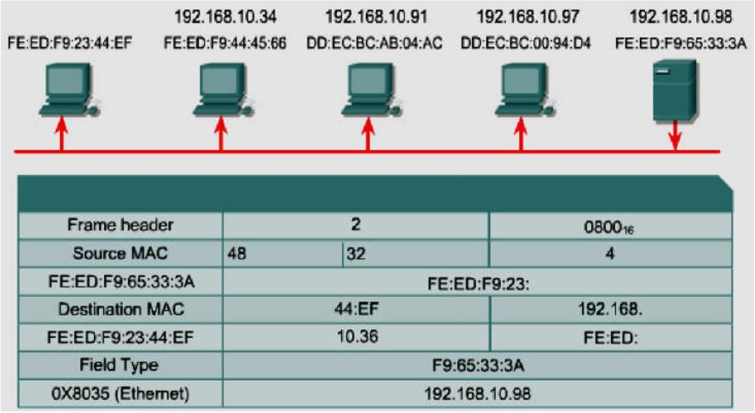


Figure 12 RARP: Reply Transmission

If MAC addresses do not match, the packet is discarded.

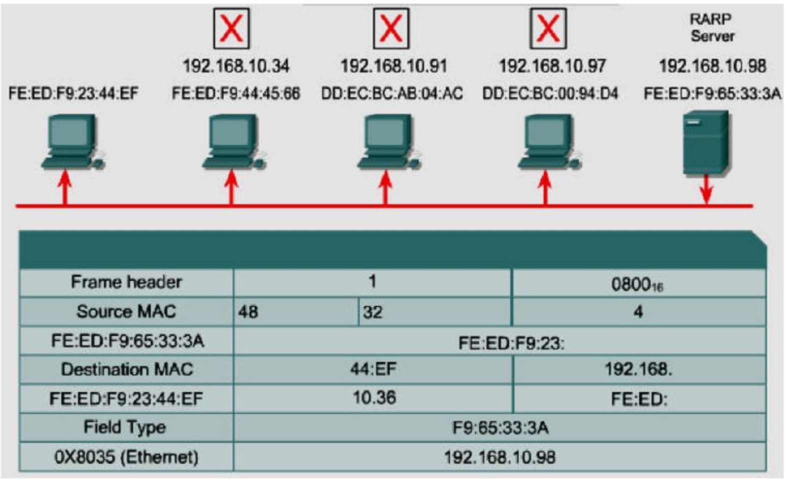
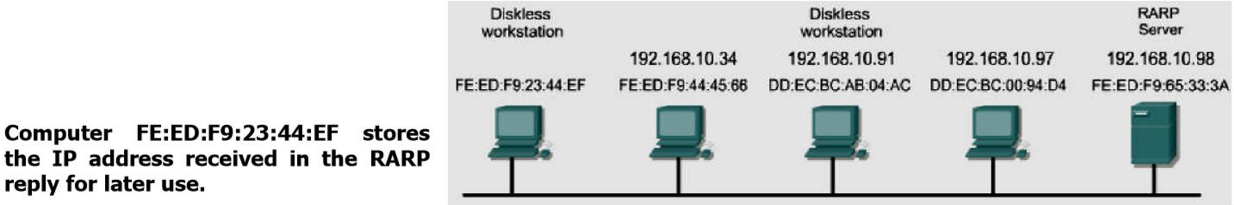


Figure 13 RARP: Reply Evaluation

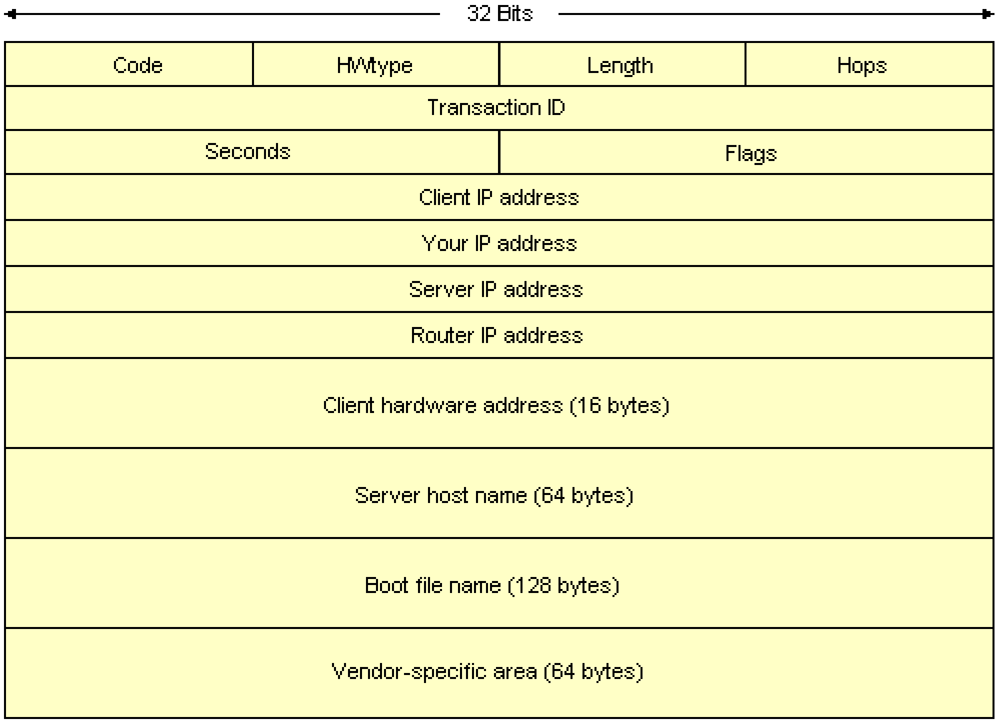


Computer FE:ED:F9:23:44:EF stores the IP address received in the RARP reply for later use.

Figure 14 RARP: Data Storage

6.6 BOOTP IP Address

Assignment The bootstrap protocol (BOOTP) operates in a client-server environment and only requires a single packet exchange to obtain IP information. However, unlike RARP, BOOTP packets can include the IP address, as well as the address of a router, the address of a server, and vendor-specific information.



32 Bits

Code	HWtype	Length	Hops
Transaction ID			
Seconds		Flags	
Client IP address			
Your IP address			
Server IP address			
Router IP address			
Client hardware address (16 bytes)			
Server host name (64 bytes)			
Boot file name (128 bytes)			
Vendor-specific area (64 bytes)			

Figure 15 BOOTP Message Structure

One problem with BOOTP, however, is that it was not designed to provide dynamic address assignment. With BOOTP, a network administrator creates a configuration file that specifies the parameters for each device. The administrator must add hosts and maintain the BOOTP database. Even though the addresses are dynamically assigned, there is still a one to one relationship between the number of IP addresses and the number of hosts. This means that for every host on the network there must be a BOOTP profile with an IP address assignment in it. No two profiles can have the same IP address. Those profiles might be used at the same time and that would mean that two hosts have the same IP address.

A description of the BOOTP message fields is given below.

Code - an operation code that specifies the message type (1 = BOOTREQUEST, 2 = BOOTREPLY)

HWtype - the type of hardware (for example, 1 = Ethernet)

Length - specifies the length of the hardware address in bytes

Hops - set to 0 by the client, and incremented by each router which relays the

Transaction ID - a 32-bit randomly generated number used to match the boot request with the response generated

Seconds - set by the client - the time elapsed in seconds since the client started its boot process

Flags - the first bit of the flags field is used as a broadcast flag - all other bits are reserved for future use and must be set to zero

Client IP address - set by the client (either its known IP address or 0.0.0.0)

Your IP address - set by the server if the Client IP address field was 0.0.0.0

Server IP address - the IP address of the BOOTP server sending a BOOTREPLY message

Router IP address - set by the forwarding router if BOOTP forwarding is used

Client hardware address - set by the client and used by the server to identify which registered client is booting

Server host name - optional server host name (a null-terminated string)

Boot file name - the client leaves this null or specifies a generic name indicating the type of boot file to be used - the server returns the fully qualified filename of a suitable boot file (a null-terminated string)

Vendor-specific Area - optional hardware or vendor-specific configuration information

A device uses BOOTP to obtain an IP address when starting up. BOOTP uses UDP to carry messages. The UDP message is encapsulated in an IP packet. A computer uses BOOTP to send a broadcast IP packet using a destination IP address of all 1s, 255.255.255.255 in dotted decimal notation. A BOOTP server receives the broadcast and then sends back a broadcast. The client receives a frame and checks the MAC address. If the client finds its own MAC address in the destination address field and a broadcast in the IP destination field, it takes and stores the IP address and other information supplied in the BOOTP reply message. A step-by-step description of the process is shown in Figures (16) through (23).

Computer FE:ED:F9:23:44:EF needs to obtain its IP address for Internet and Internet operation.

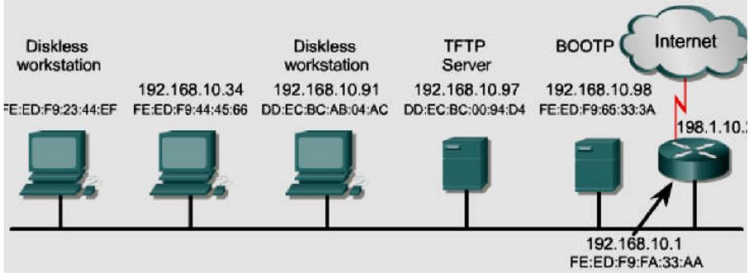
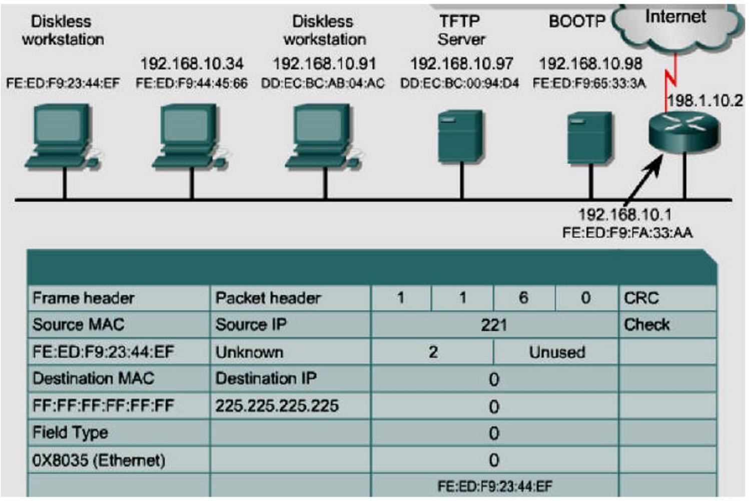


Figure 16 BOOTP: Network Segment

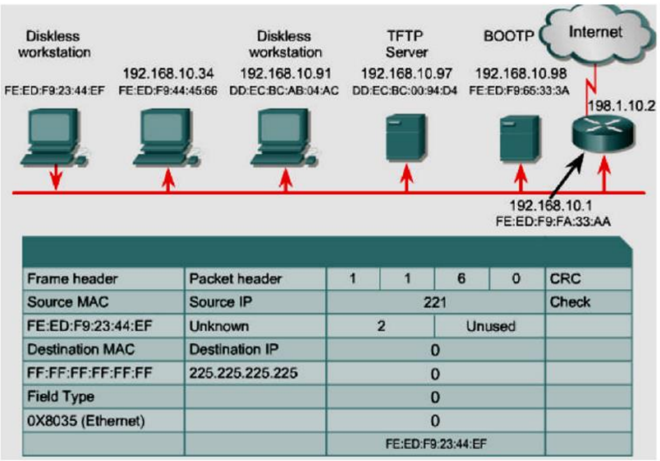
Workstation FE:ED:F9:23:44:EF generates a BOOTP request.



Frame header	Packet header	1	1	6	0	CRC
Source MAC	Source IP	221				Check
FE:ED:F9:23:44:EF	Unknown	2		Unused		
Destination MAC	Destination IP	0				
FF:FF:FF:FF:FF:FF	225.225.225.225	0				
Field Type		0				
0X8035 (Ethernet)		0				
		FE:ED:F9:23:44:EF				

Figure 17 BOOTP: Request Creation

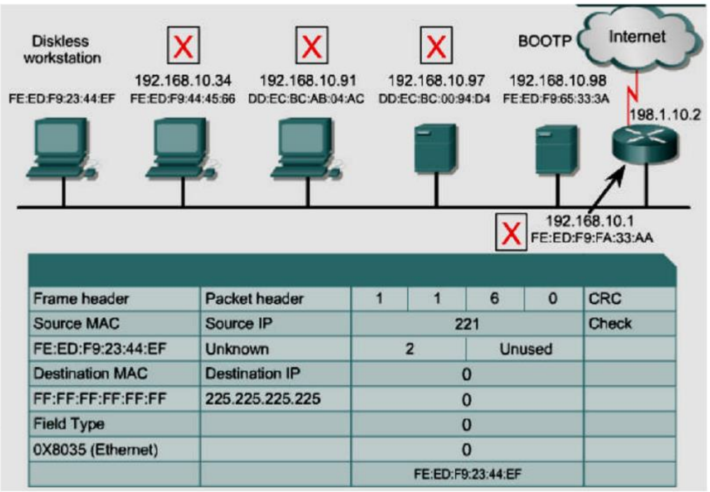
Workstation FE:ED:F9:23:44:EF encapsulates the request in a packet header. The header contains an unknown source IP address and a broadcast destination IP address. For the frame header the workstation uses its MAC address as the source and a broadcast for the destination as it does not know the address of the BOOTP server. The workstation then transmits a BOOTP request frame.



Frame header	Packet header	1	1	6	0	CRC
Source MAC	Source IP	221				Check
FE:ED:F9:23:44:EF	Unknown	2		Unused		
Destination MAC	Destination IP	0				
FF:FF:FF:FF:FF:FF	225.225.225.225	0				
Field Type		0				
0X8035 (Ethernet)		0				
		FE:ED:F9:23:44:EF				

Figure 18 BOOTP: Request Transmission

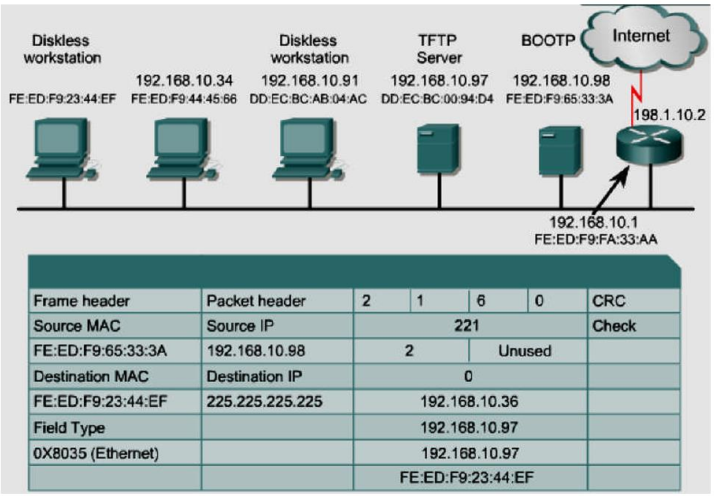
All devices pick up a copy of the frame, detect a broadcast MAC destination, strip off the frame header, and pass the packet up to the network layer. The devices detect that the IP destination is a broadcast IP address, strip off the packet header, and pass the reply data to the transport layer. All of the devices detect the BOOTP request field as being a BOOTP request. All devices except for the BOOTP server discard it.



Frame header	Packet header	1	1	6	0	CRC
Source MAC	Source IP	221				Check
FE:ED:F9:23:44:EF	Unknown	2		Unused		
Destination MAC	Destination IP	0				
FF:FF:FF:FF:FF:FF	225.225.225.225	0				
Field Type		0				
0X8035 (Ethernet)		0				
		FE:ED:F9:23:44:EF				

Figure 19 BOOTP: Request Verification

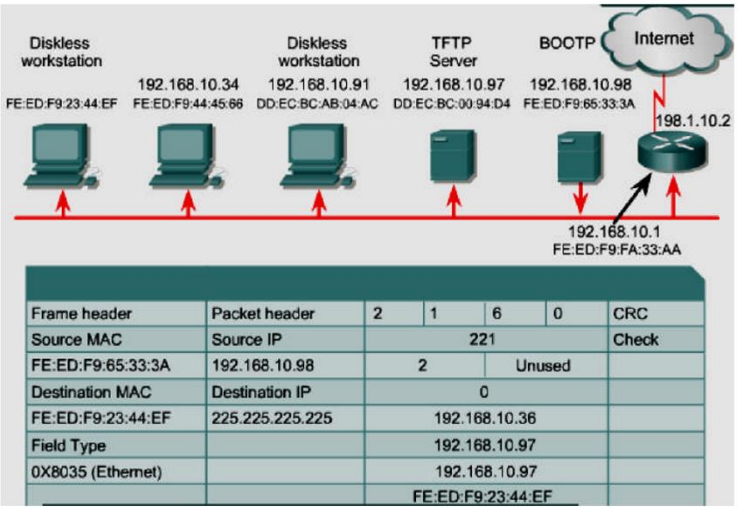
The server prepares a BOOTP response from its database to send back to the requesting device. This includes client IP address. TFTP server address, and default Gateway address (other fields are omitted for this example). In the frame header, source and destination addresses are reversed. In the packet header, the BOOTP server places its IP address in the source field and a broadcast address in the destination field. This is done to get the BOOTP response packet back up to the transport layer to be processed. Only a broadcast will be passed since the client still does not know its IP address.



Frame header	Packet header	2	1	6	0	CRC
Source MAC	Source IP	221				Check
FE:ED:F9:65:33:3A	192.168.10.98	2		Unused		
Destination MAC	Destination IP	0				
FE:ED:F9:23:44:EF	225.225.225.225	192.168.10.36				
Field Type		192.168.10.97				
0X8035 (Ethernet)		192.168.10.97				
		FE:ED:F9:23:44:EF				

Figure 20 BOOTP: Reply Creation

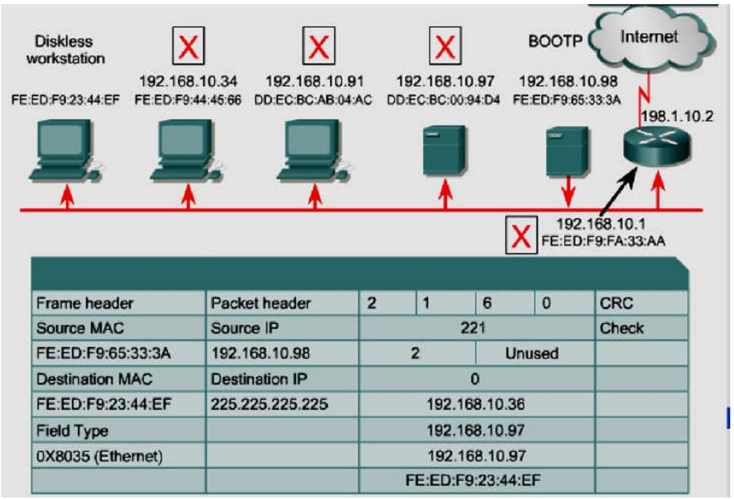
The BOOTP server then sends the BOOTP reply frame back to the requesting device. All devices pick up the packet and examine it.



Frame header	Packet header	2	1	6	0	CRC
Source MAC	Source IP	221				Check
FE:ED:F9:65:33:3A	192.168.10.98	2		Unused		
Destination MAC	Destination IP	0				
FE:ED:F9:23:44:EF	225.225.225.225	192.168.10.36				
Field Type		192.168.10.97				
0X8035 (Ethernet)		192.168.10.97				
		FE:ED:F9:23:44:EF				

Figure 21 BOOTP: Reply Transmission

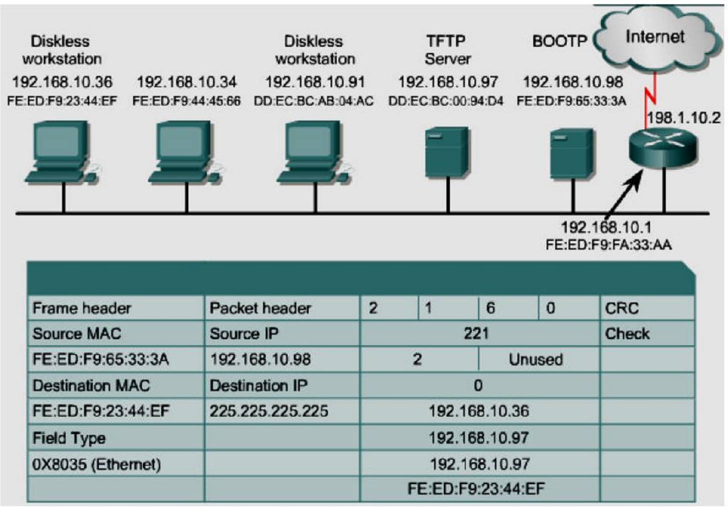
The destination MAC address is not theirs and not a broadcast, so they discard the packet. The MAC address is matched on the requesting client device, so the source IP and MAC address of the BOOTP server are stored in the ARP table of the diskless workstation. The frame header is stripped off and discarded.



Frame header	Packet header	2	1	6	0	CRC
Source MAC	Source IP	221				Check
FE:ED:F9:65:33:3A	192.168.10.98	2		Unused		
Destination MAC	Destination IP	0				
FE:ED:F9:23:44:EF	225.225.225.225	192.168.10.36				
Field Type		192.168.10.97				
0X8035 (Ethernet)		192.168.10.97				
		FE:ED:F9:23:44:EF				

Figure 22 BOOTP: Reply Verified

The packet destination IP is a broadcast, so the packet header is stripped off and the BOOTP reply data is passed up to the transport layer, where the OP field data says that this is a BOOTP reply. The reply data is stored in the appropriate memory locations in the workstation. The workstation now has access to the TFTP server for further operating system downloads and to the default Gateway as well as having its own IP address. It can now fully function on the network and the Internet.

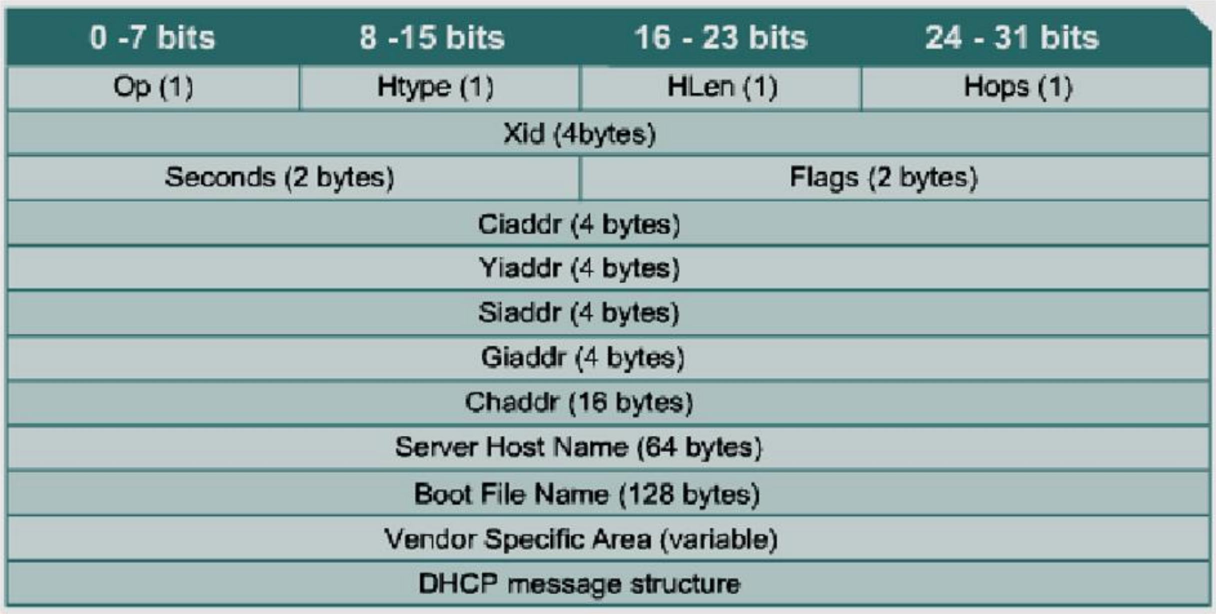


Frame header	Packet header	2	1	6	0	CRC
Source MAC	Source IP	221				Check
FE:ED:F9:65:33:3A	192.168.10.98	2		Unused		
Destination MAC	Destination IP	0				
FE:ED:F9:23:44:EF	225.225.225.225	192.168.10.36				
Field Type		192.168.10.97				
0X8035 (Ethernet)		192.168.10.97				
		FE:ED:F9:23:44:EF				

Figure 23 BOOTP: Data Storage

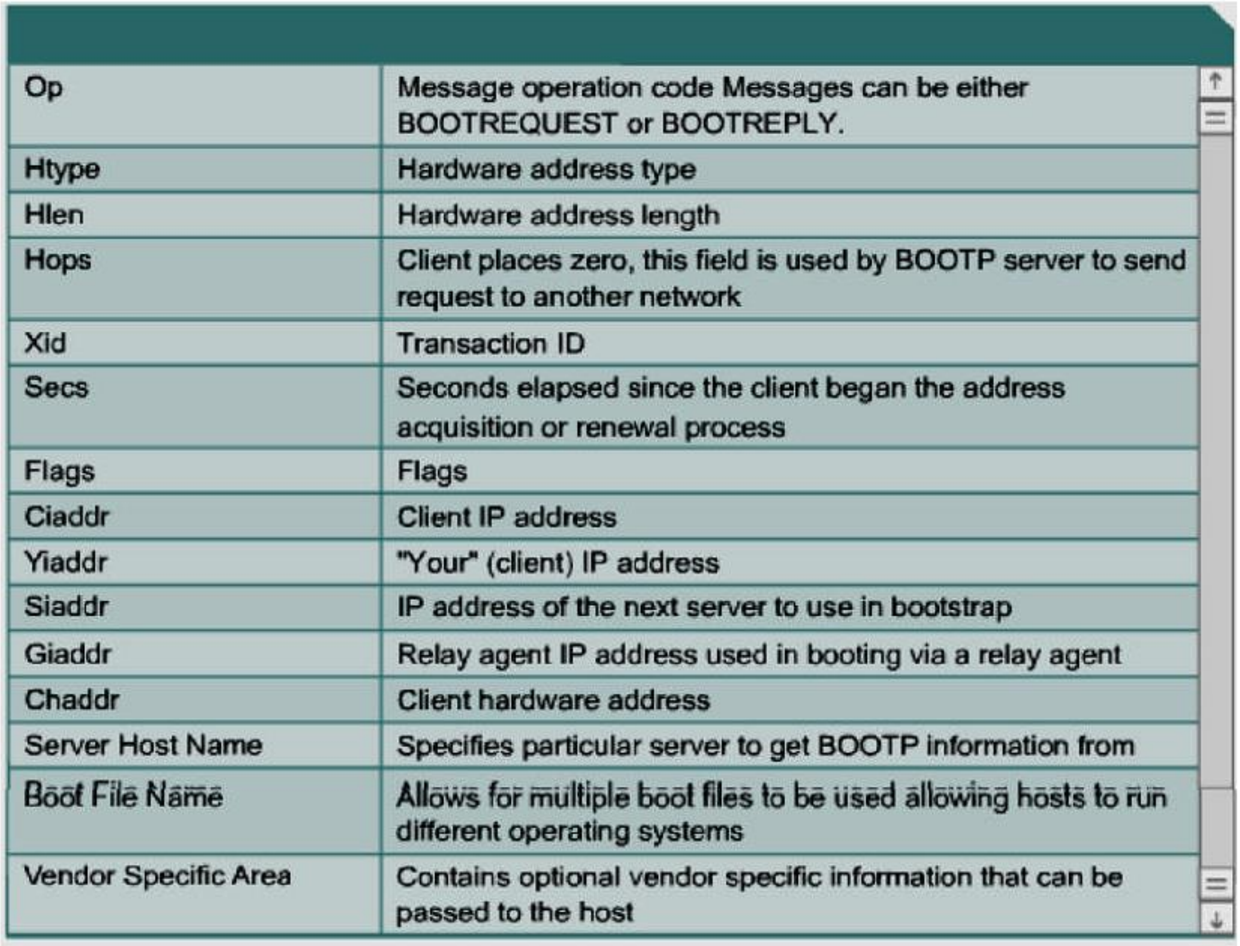
6.7 DHCP IP Address Management

Dynamic host configuration protocol (DHCP) is the successor to BOOTP. Unlike BOOTP, DHCP allows a host to obtain an IP address dynamically without the network administrator having to set up an individual profile for each device. All that is required when using DHCP is a defined range of IP addresses on a DHCP server. As hosts come online, they contact the DHCP server and request an address. The DHCP server chooses an address and leases it to that host. With DHCP, the entire network configuration of a computer can be obtained in one message. This includes all of the data supplied by the BOOTP message, plus a leased IP address and a subnet mask.



0 -7 bits	8 -15 bits	16 - 23 bits	24 - 31 bits
Op (1)	Htype (1)	HLen (1)	Hops (1)
Xid (4bytes)			
Seconds (2 bytes)		Flags (2 bytes)	
Ciaddr (4 bytes)			
Yiaddr (4 bytes)			
Siaddr (4 bytes)			
Giaddr (4 bytes)			
Chaddr (16 bytes)			
Server Host Name (64 bytes)			
Boot File Name (128 bytes)			
Vendor Specific Area (variable)			
DHCP message structure			

Figure 24 DHCP Message Structure



Op	Message operation code Messages can be either BOOTREQUEST or BOOTREPLY.
Htype	Hardware address type
Hlen	Hardware address length
Hops	Client places zero, this field is used by BOOTP server to send request to another network
Xid	Transaction ID
Secs	Seconds elapsed since the client began the address acquisition or renewal process
Flags	Flags
Ciaddr	Client IP address
Yiaddr	"Your" (client) IP address
Siaddr	IP address of the next server to use in bootstrap
Giaddr	Relay agent IP address used in booting via a relay agent
Chaddr	Client hardware address
Server Host Name	Specifies particular server to get BOOTP information from
Boot File Name	Allows for multiple boot files to be used allowing hosts to run different operating systems
Vendor Specific Area	Contains optional vendor specific information that can be passed to the host

Figure 25 DHCP Message Structure Field Descriptions

The major advantage that DHCP has over BOOTP is that it allows users to be mobile. This mobility allows the users to freely change network connections from location to location. It is no longer required to keep a fixed profile for every device attached to the network as was required with the BOOTP system. The importance to this DHCP advancement is its ability to lease an IP address to a device and then reclaim that IP address for another user after the first user releases it. This means that DHCP offers a one to many ratio of IP addresses and that an address is available to anyone who connects to the network. A step-by-step description of the process is shown in Figures (26) through (40).

Laptop AA:EC:F9:23:44:19 is connected to the network.

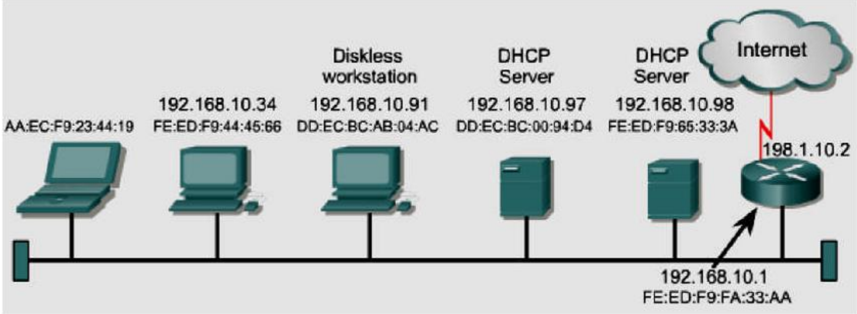
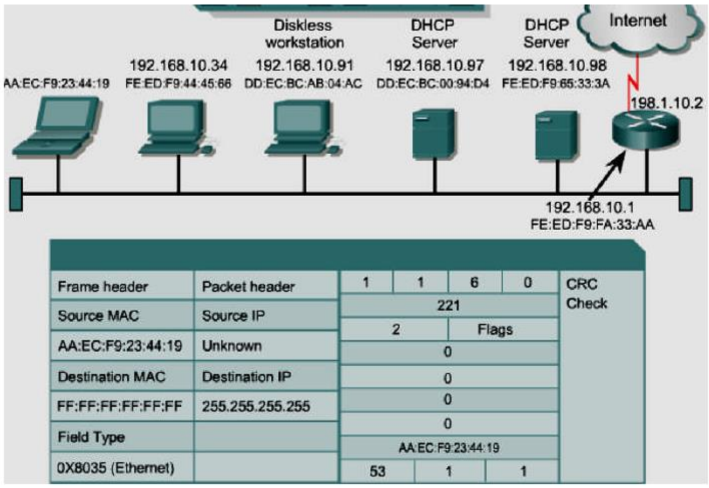


Figure 26 DHCP: Host Boots

Laptop AA:EC:F9:23:44:19 generates a DHCP request.

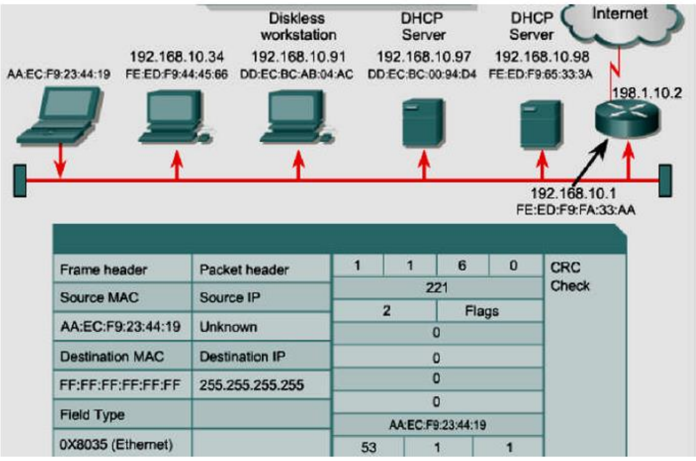


Frame header	Packet header
Source MAC	Source IP
AA:EC:F9:23:44:19	Unknown
Destination MAC	Destination IP
FF:FF:FF:FF:FF:FF	255.255.255.255
Field Type	
0X8035 (Ethernet)	

1	1	6	0	CRC Check
221				
2		Flags		
0				
0				
0				
0				
AA:EC:F9:23:44:19				
53	1	1		

Figure 27 DHCP: Message Structure Field Descriptions

The DHCP request is transmitted by the laptop computer.

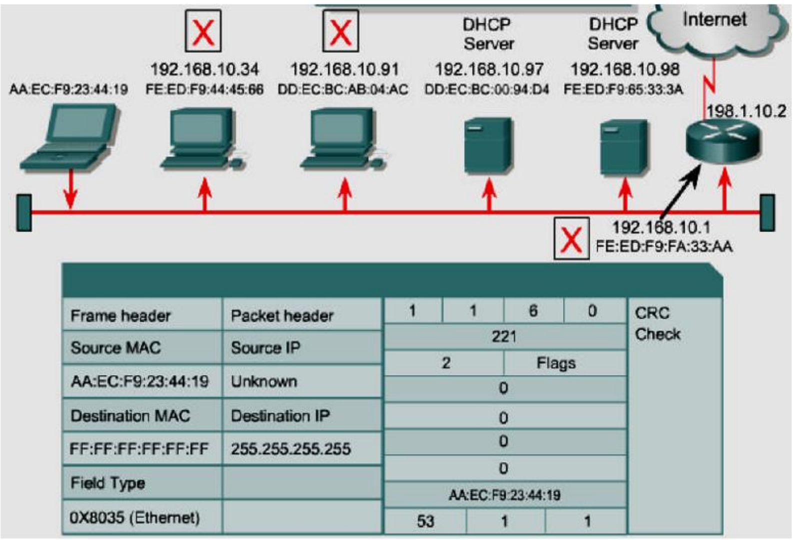


Frame header	Packet header
Source MAC	Source IP
AA:EC:F9:23:44:19	Unknown
Destination MAC	Destination IP
FF:FF:FF:FF:FF:FF	255.255.255.255
Field Type	
0X8035 (Ethernet)	

1	1	6	0	CRC Check
221				
2		Flags		
0				
0				
0				
0				
AA:EC:F9:23:44:19				
53	1	1		

Figure 28 DHCP: Request Transmitted

All devices pick up a copy of the frame, detect a broadcast MAC destination, strip off the frame header, and pass the packet up to the network layer. The devices detect that the IP destination is a broadcast IP address, strip off the packet header, and pass the reply data to the transport layer. All of the devices detect the DHCP request field as being a DHCP request. All devices except for the DHCP servers discard the request.



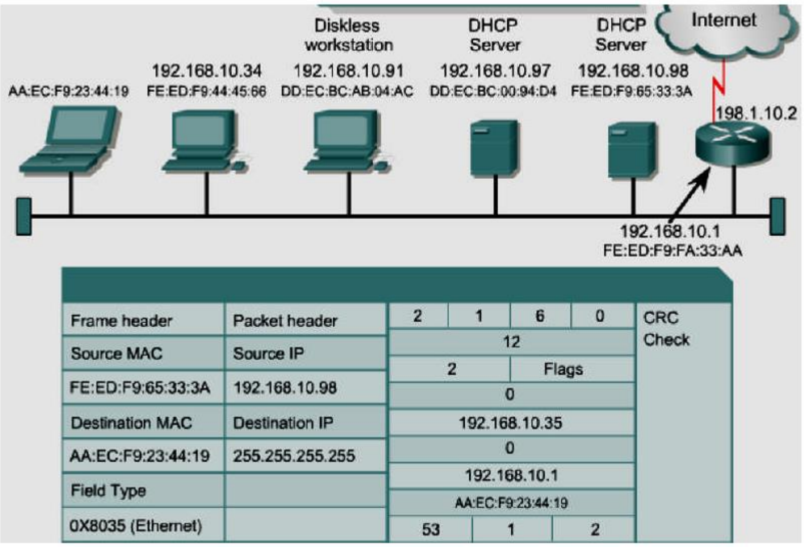
Frame header	Packet header
Source MAC	Source IP
AA:EC:F9:23:44:19	Unknown
Destination MAC	Destination IP
FF:FF:FF:FF:FF:FF	255.255.255.255
Field Type	
0X8035 (Ethernet)	

1	1	6	0
221			
2		Flags	
0			
0			
0			
0			
AA:EC:F9:23:44:19			
53	1	1	

CRC Check

Figure 29 DHCP: Request Evaluated

The server prepares a DHCP offer to send back to the requesting device. This includes client IP address. DHCP server address, and default Gateway address. In the frame header, source and destination addresses are reversed. In the packet header, the DHCP server places its IP address in the source field and a broadcast address in the destination field. This is done to get the DHCP response packet back up to the transport layer to be processed. Only a broadcast will be passed since the client still does not know its IP address.



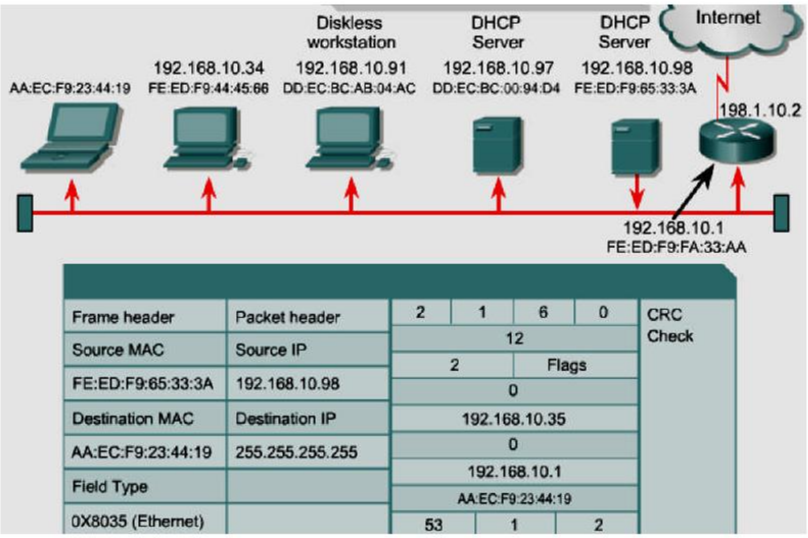
Frame header	Packet header
Source MAC	Source IP
FE:ED:F9:65:33:3A	192.168.10.98
Destination MAC	Destination IP
AA:EC:F9:23:44:19	255.255.255.255
Field Type	
0X8035 (Ethernet)	

2	1	6	0
12			
2		Flags	
0			
192.168.10.35			
0			
192.168.10.1			
AA:EC:F9:23:44:19			
53	1	2	

CRC Check

Figure 30 DHCP: Offer Prepared

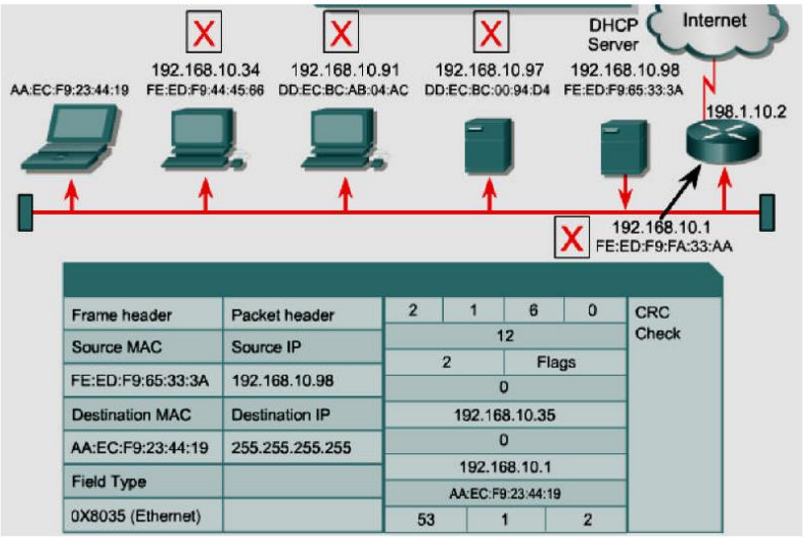
The DHCP server sends the DHCP reply frame back to the requesting device. All devices pick up the packet and examine it.



Frame header	Packet header	2 1 6 0	CRC Check
Source MAC	Source IP	12	
FE:ED:F9:65:33:3A	192.168.10.98	2 Flags 0	
Destination MAC	Destination IP	192.168.10.35	
AA:EC:F9:23:44:19	255.255.255.255	0	
Field Type		192.168.10.1 AA:EC:F9:23:44:19	
0X8035 (Ethernet)		53 1 2	

Figure 31 DHCP: Offer Transmitted

The destination MAC address is not theirs and not a broadcast, so they discard the packet. The MAC address is matched on the requesting client device, and so the source IP and MAC address of the DHCP server are stored in the ARP table of the laptop. The frame header is stripped off and discarded.



Frame header	Packet header	2 1 6 0	CRC Check
Source MAC	Source IP	12	
FE:ED:F9:65:33:3A	192.168.10.98	2 Flags 0	
Destination MAC	Destination IP	192.168.10.35	
AA:EC:F9:23:44:19	255.255.255.255	0	
Field Type		192.168.10.1 AA:EC:F9:23:44:19	
0X8035 (Ethernet)		53 1 2	

Figure 32 DHCP: Offer Evaluated

The second DHCP server sends the DHCP reply frame back to the requesting device. All devices pick up the packet and examine it.

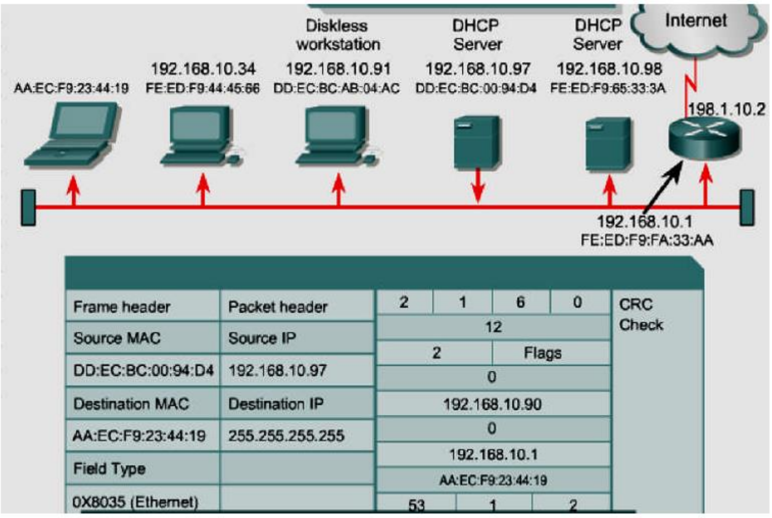


Figure 33 DHCP: Offer Transmitted

The destination MAC address is not theirs and not a broadcast, so they discard the packet. The MAC address is matched on the requesting client device, and so the source IP and MAC address of the DHCP server are stored in the ARP table of the laptop. The frame header is stripped off and discarded. Since the laptop has already received a DHCP offer from another server, this offer is discarded.

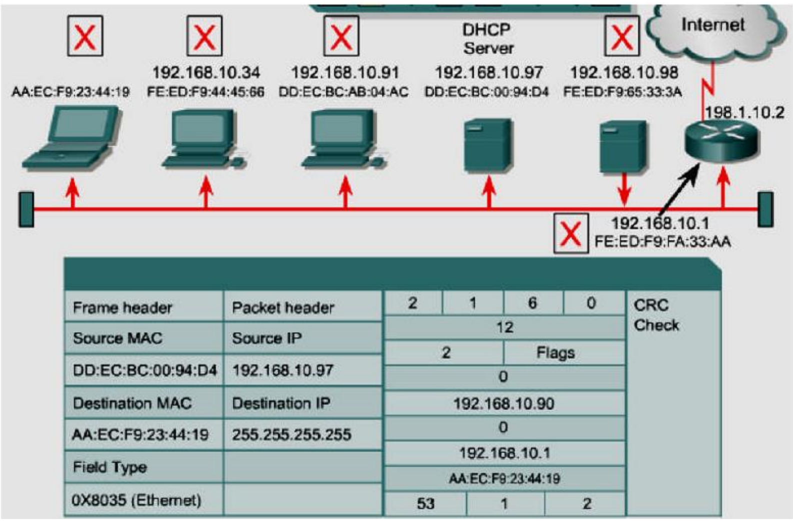
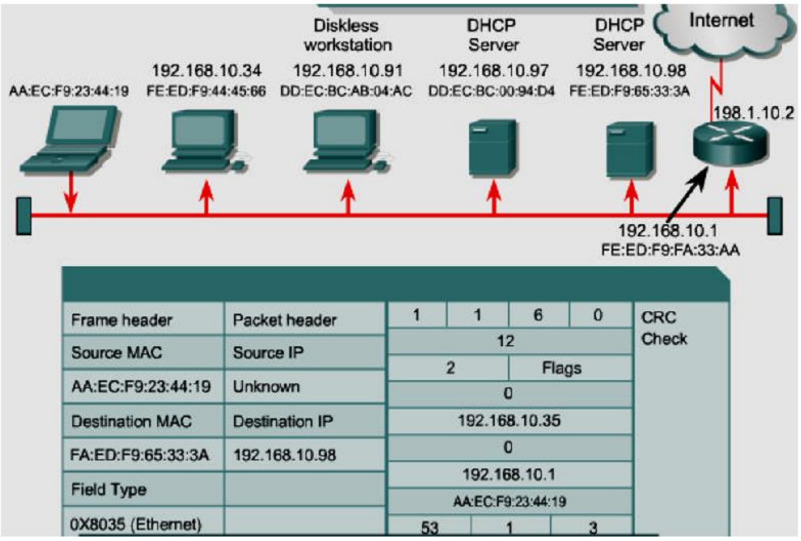


Figure 34 DHCP: Offer Evaluated

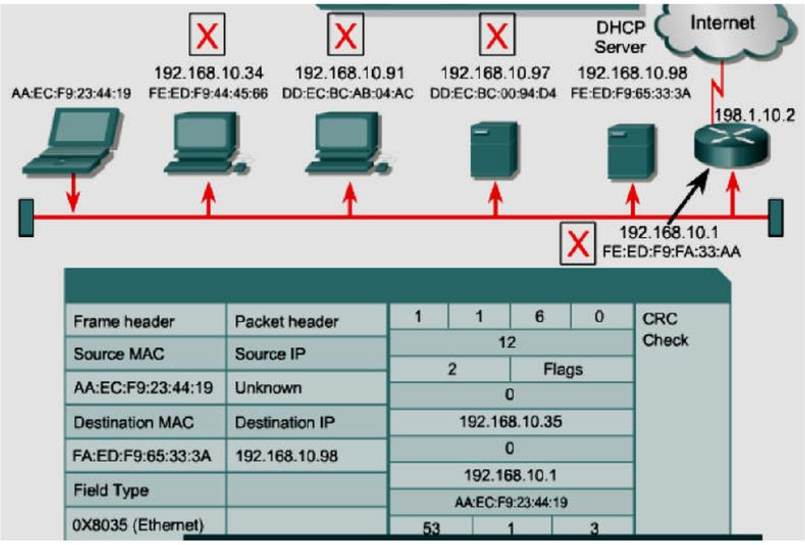
The laptop computer now sends a DHCP request addressed to the specific DHCP server that sent the accepted offer.



Frame header	Packet header	1	1	6	0	CRC Check
Source MAC	Source IP	12				
AA:EC:F9:23:44:19	Unknown	2		Flags		
Destination MAC	Destination IP	0				
FA:ED:F9:65:33:3A	192.168.10.98	192.168.10.35				
Field Type		0				
0X8035 (Ethernet)		192.168.10.1				
		AA:EC:F9:23:44:19				
		53	1	3		

Figure 35 DHCP: Request Generated

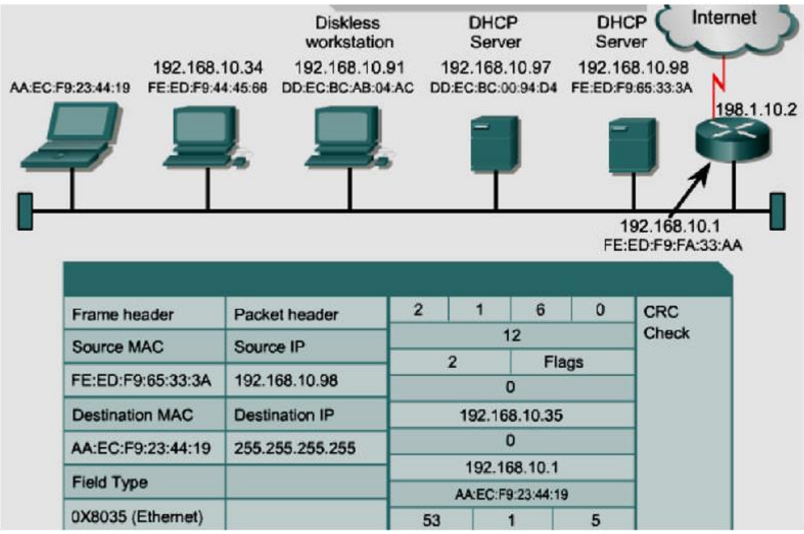
All devices pick up a copy of the frame and compare the MAC destination to their own. If there is no match, the devices discard the frame.



Frame header	Packet header	1	1	6	0	CRC Check
Source MAC	Source IP	12				
AA:EC:F9:23:44:19	Unknown	2		Flags		
Destination MAC	Destination IP	0				
FA:ED:F9:65:33:3A	192.168.10.98	192.168.10.35				
Field Type		0				
0X8035 (Ethernet)		192.168.10.1				
		AA:EC:F9:23:44:19				
		53	1	3		

Figure 36 DHCP: Request Transmitted

The DHCP selected server creates a DHCPACK.



Frame header	Packet header
Source MAC	Source IP
FE:ED:F9:65:33:3A	192.168.10.98
Destination MAC	Destination IP
AA:EC:F9:23:44:19	255.255.255.255
Field Type	
0X8035 (Ethernet)	

2 1 6 0

12

2 Flags

0

192.168.10.35

0

192.168.10.1

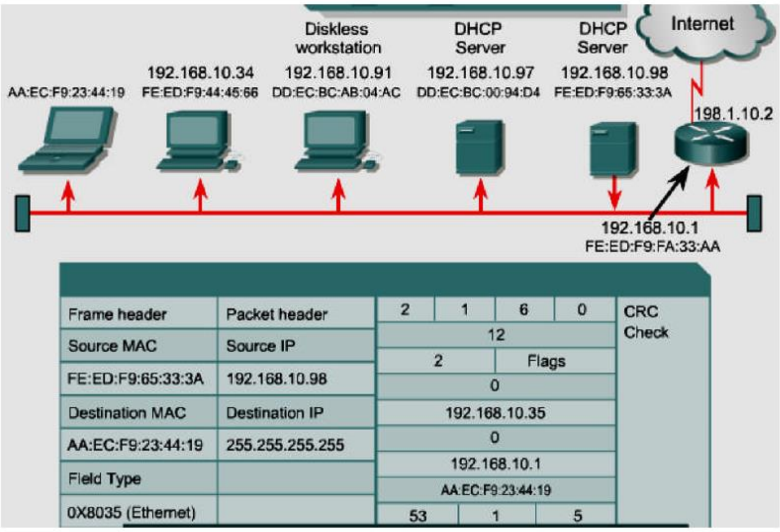
AA:EC:F9:23:44:19

53 1 5

CRC Check

Figure 37 DHCP: DHCPACK Created

The DHCP server sends the DHCPACK frame back to the requesting device. All devices pick up the packet and examine it.



Frame header	Packet header
Source MAC	Source IP
FE:ED:F9:65:33:3A	192.168.10.98
Destination MAC	Destination IP
AA:EC:F9:23:44:19	255.255.255.255
Field Type	
0X8035 (Ethernet)	

2 1 6 0

12

2 Flags

0

192.168.10.35

0

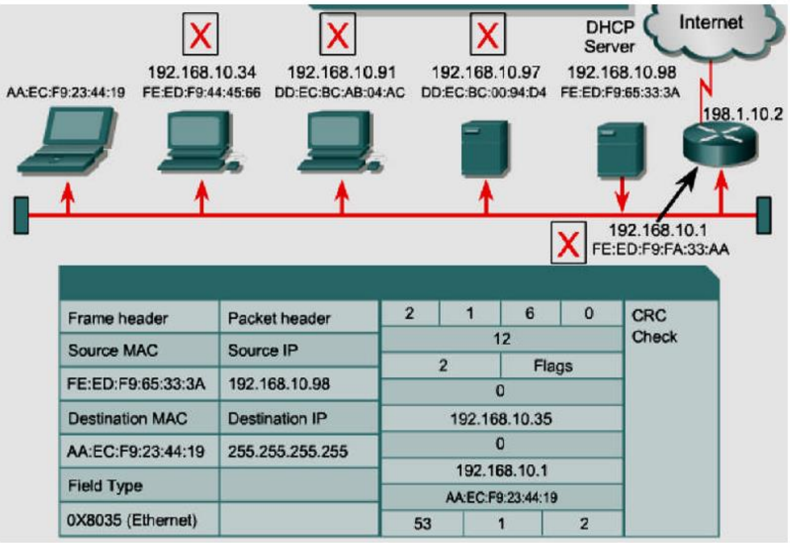
192.168.10.1

AA:EC:F9:23:44:19

53 1 5

CRC Check

Figure 38 DHCP: DHCPACK Transmitted

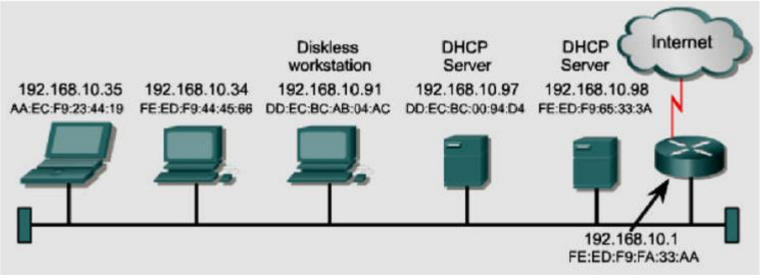


Frame header	Packet header
Source MAC	Source IP
FE:ED:F9:65:33:3A	192.168.10.98
Destination MAC	Destination IP
AA:EC:F9:23:44:19	255.255.255.255
Field Type	
0X8035 (Ethernet)	

<table>
<tr><td>2</td><td>1</td><td>6</td><td>0</td><td rowspan="9">CRC Check</td></tr>
<tr><td colspan="4">12</td></tr>
<tr><td colspan="2">2</td><td colspan="2">Flags</td></tr>
<tr><td colspan="4">0</td></tr>
<tr><td colspan="4">192.168.10.35</td></tr>
<tr><td colspan="4">0</td></tr>
<tr><td colspan="4">192.168.10.1</td></tr>
<tr><td colspan="4">AA:EC:F9:23:44:19</td></tr>
<tr><td>53</td><td colspan="2">1</td><td>2</td></tr>
</table>

The destination MAC address is not theirs and not a broadcast, so they discard the packet. The MAC address is matched on the requesting client device, and so the source IP and MAC address of the DHCP server are stored in the ARP table of the laptop. The frame header is stripped off and discarded.

Figure 39 DHCP: DHCPACK Evaluated

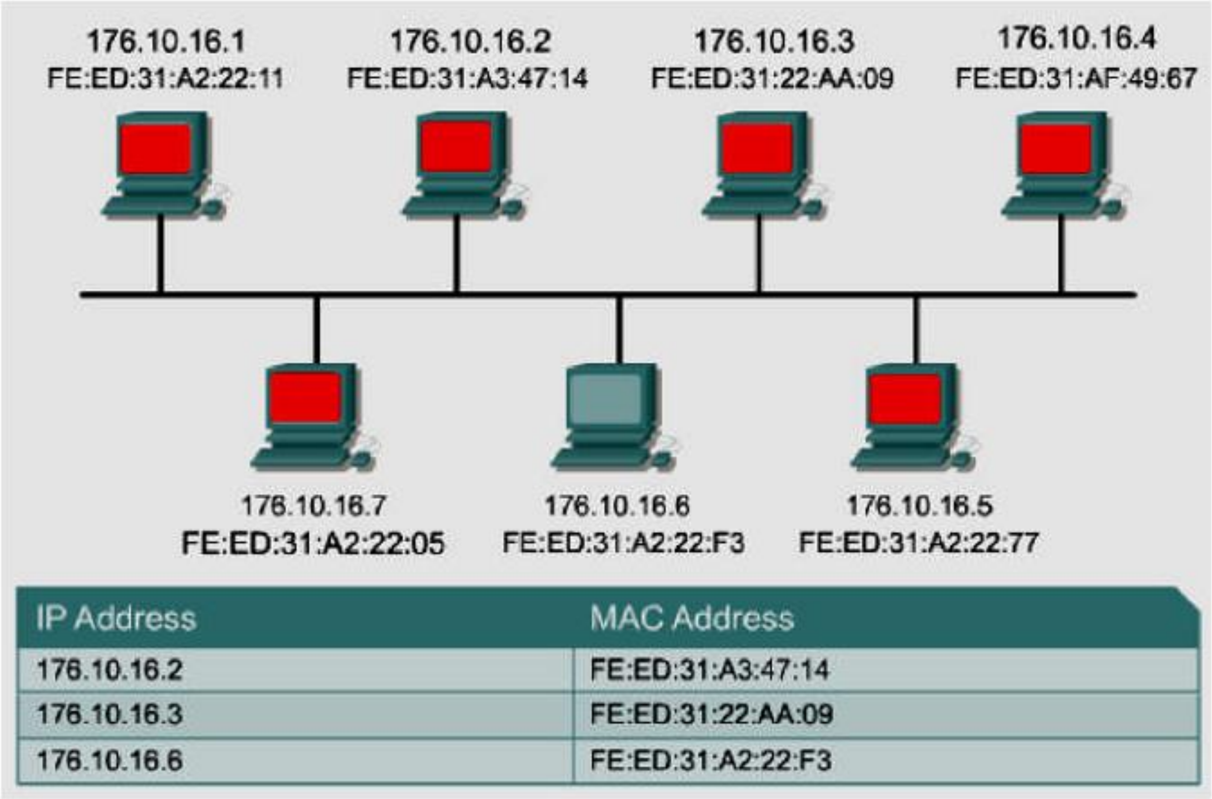


The laptop computer now goes into the bound mode and starts to use the assigned IP address and other data passed with the DHCP offer message.

Figure 40 DHCP: DHCPACK Created

6.8 Problems in Address Resolution

One of the major problems in networking is how to communicate with other network devices.



IP Address	MAC Address
176.10.16.2	FE:ED:31:A3:47:14
176.10.16.3	FE:ED:31:22:AA:09
176.10.16.6	FE:ED:31:A2:22:F3

Figure 41 LAN Transmission Address Resolution Issues

- Computer 176.10.16.1 is monitoring the Ethernet segment to update its ARP table with IP-MAC address pairs so that it can send data to other hosts on the LAN.
- Computer 176.10.16.2 prepares the data for transmission. To do that it checks the network cable to see if another computer is using it. If another station is using the cable, computer 176.10.16.2 will have to wait, as only one computer can transmit at a time. The cable is clear so computer 176.10.16.2 can transmit.
- Computer 176.10.16.2 transmits the data frames through the network cable segment.
- All computers on the Ethernet segment analyze the incoming data frames to determine if the transmission is for them. Part of this process adds the IP-MAC source addresses to the ARP table. All devices except the one that the data was sent discard the data frame.

- Computer 176.10.16.3 prepares the data for transmission. It follows all the preparation steps.
- Computer 176.10.16.3 transmits its data frames through the Ethernet segment.
- Again all hosts on the segment analyze the incoming frames. Adding data to their ARP tables and discarding the frame if they were not the specified destination of the data.
- Computer 176.10.16.6 prepares the data for transmission.
- Computer 176.10.16.6 transmits its data frames through the Ethernet segment.
- All hosts on the segment analyze the incoming frames. They add data to their ARP tables and discard the frames if they were not the specified destination of the data. This shows the automatic process that is used on a normal Ethernet LAN for maintaining address associations.
- Computer 176.10.16.1 wants to send data to 176.10.16.4. It has its IP address, but data transmission also requires the MAC address of 176.10.16.4. How does it get that MAC address to perform the data transmission?

In TCP/IP communications, a datagram on a LAN must contain both a destination MAC address and a destination IP address. These addresses must be correct and match the destination MAC and IP addresses of the host device. If it does not match, the datagram will be discarded by the destination host. Communications within a LAN segment require two addresses. There needs to be a way to automatically map IP to MAC addresses. It would be too time consuming for the user to create the maps manually. The TCP/IP suite has a protocol, called Address Resolution Protocol (ARP), which can automatically obtain MAC addresses for local transmission. Different issues are raised when data is sent outside of the local area network.

Computer 192.168.10.34 needs to communicate with computer 192.168.1.1. How does it get the MAC address for 192.168.1.1, and would it do any good if it was able to get the MAC address? Remember that MAC addresses are only useful in a LAN. They will not be any help outside of the 192.168.10.0 network. So the MAC address of the router is needed to get the data out of the LAN and on to the WAN system.

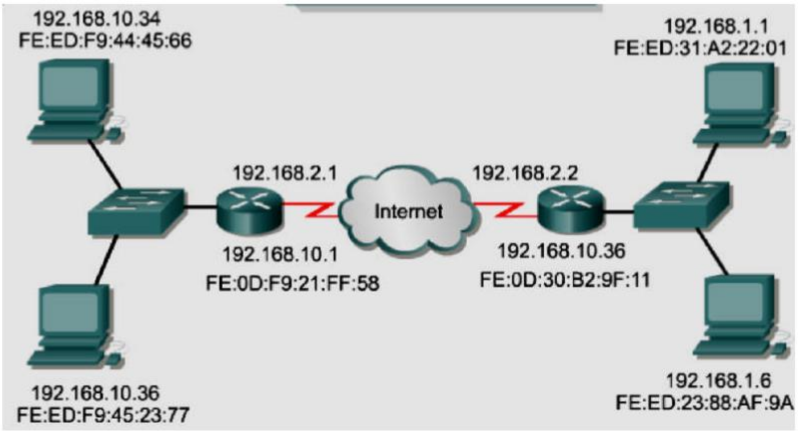
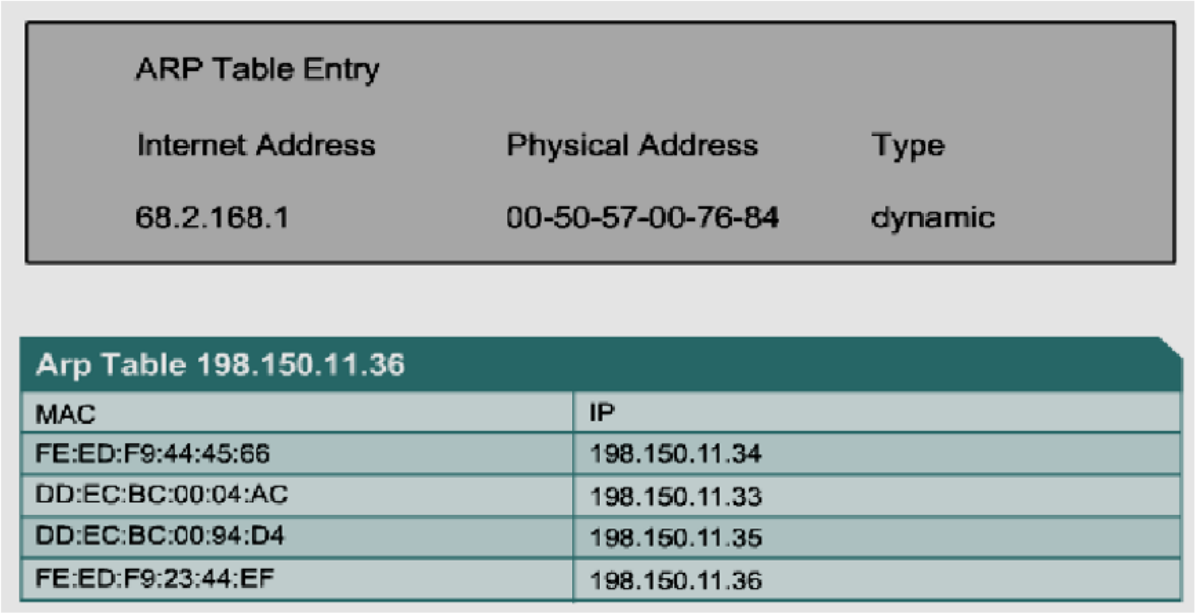


Figure 42 Non-Local Address Resolution Issues

Communications between two LAN segments have an additional task. Both the IP and MAC addresses are needed for both the destination host and the intermediate routing device. TCP/IP has a variation on ARP called Proxy ARP that will provide the MAC address of an intermediate device for transmission outside the LAN to another network segment.

6.9 Address Resolution Protocol (ARP)

With TCP/IP networking, a data packet must contain both a destination MAC address and a destination IP address. If the packet is missing either one, the data will not pass from Layer 3 to the upper layers. In this way, MAC addresses and IP addresses act as checks and balances for each other. After devices determine the IP addresses of the destination devices, they can add the destination MAC addresses to the data packets. Some devices will keep tables that contain MAC addresses and IP addresses of other devices that are connected to the same LAN. These are called Address Resolution Protocol (ARP) tables. ARP tables are stored in RAM memory, where the cached information is maintained automatically on each of the devices. It is very unusual for a user to have to make an ARP table entry manually. Each device on a network maintains its own ARP table. When a network device wants to send data across the network, it uses information provided by the ARP table. When a source determines the IP address for a destination, it then consults the ARP table in order to locate the MAC address for the destination. If the source locates an entry in its table, destination IP address to destination MAC address, it will associate the IP address to the MAC address and then uses it to encapsulate the data. The data packet is then sent out over the networking media to be picked up by the destination device.



ARP Table Entry

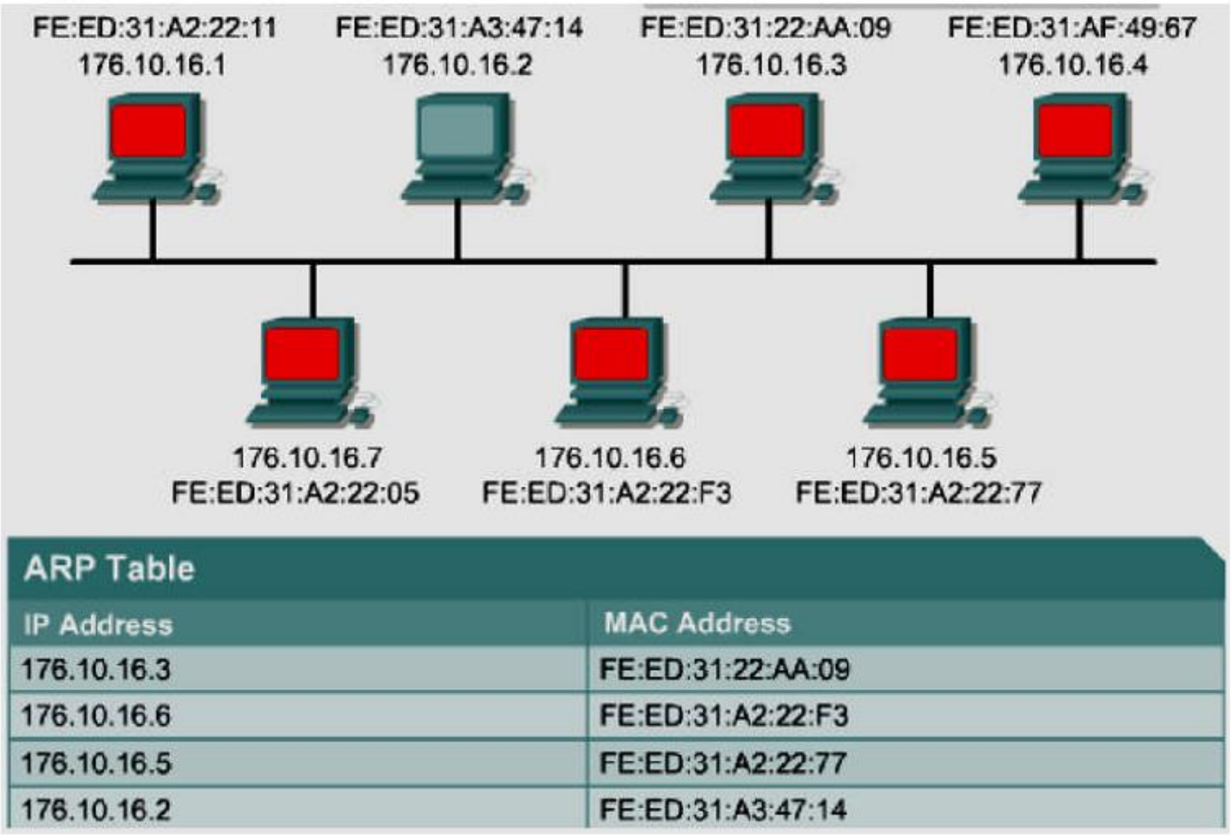
Internet Address	Physical Address	Type
68.2.168.1	00-50-57-00-76-84	dynamic

Arp Table 198.150.11.36

MAC	IP
FE:ED:F9:44:45:66	198.150.11.34
DD:EC:BC:00:04:AC	198.150.11.33
DD:EC:BC:00:94:D4	198.150.11.35
FE:ED:F9:23:44:EF	198.150.11.36

Figure 43 ARP Table Entry

There are two ways that devices can gather MAC addresses that they need to add to the encapsulated data. One way is to monitor the traffic that occurs on the local network segment. All stations on an Ethernet network will analyze all traffic to determine if the data is for them. Part of this process is to record the source IP and MAC address of the datagram to an ARP table. So as data is transmitted on the network, the address pairs populate the ARP table. Another way to get an address pair for data transmission is to broadcast an ARP request.



ARP Table

IP Address	MAC Address
176.10.16.3	FE:ED:31:22:AA:09
176.10.16.6	FE:ED:31:A2:22:F3
176.10.16.5	FE:ED:31:A2:22:77
176.10.16.2	FE:ED:31:A3:47:14

Figure 44ARP Table Functions

- Computer 176.10.16.1 is monitoring the Ethernet segment to update its ARP table.
- Computer 176.10.16.2 prepares the data for transmission. To do that it checks the network cable to see if another computer is using it. If another station is using the cable, computer 176.10.16.2 will have to wait, as only one computer can transmit at a time. The cable is clear so computer 176.10.16.2 can transmit.
- Computer 176.10.16.2 transmits the data frames through the network cable segment.
- All computers on the Ethernet segment analyze the incoming data frames to determine if the transmission is for them. Part of this process is to add the IP-MAC source addresses from the data to the ARP table.

- Computer 176.10.16.3 prepares the data for transmission. It follows all the preparation steps.
- Computer 176.10.16.3 transmits its data frames through the Ethernet segment.
- Again all hosts on the segment analyze the incoming frames and add data to their ARP tables.
- Computer 176.10.16.6 prepares the data for transmission.
- Computer 176.10.16.6 transmits its data frames through the Ethernet segment.
- All hosts on the segment analyze the incoming frames.
- Computer 176.10.16.5 prepares the data for transmission. Notice the first pair in the ARP table, it is reaching its timeout value. If a computer does not transmit data for a certain length of time, their IPMAC pair is dropped from the ARP table.
- Computer 176.10.16.3 transmits its data frames through the Ethernet segment. The first value in the ARP table exceeded the timeout value so it is removed. The ARP table is dynamically updated. It adds and removes entire based on segment activity and timeout values.
- Again all hosts on the segment analyze the incoming frames. New values are added to the ARP table.
- Computer 176.10.16.2 prepares the data for transmission.
- Computer 176.10.16.1 transmits the data frames through the network cable segment.
- All computers on the Ethernet segment analyze the incoming data frames to determine if the transmission is for them. The IP-MAC pair for 176.10.16.2 is added back into the table. If this transmission had come before the timeout value was exceeded, the pair would not have been removed from the table, the timeout value would have just been reset.

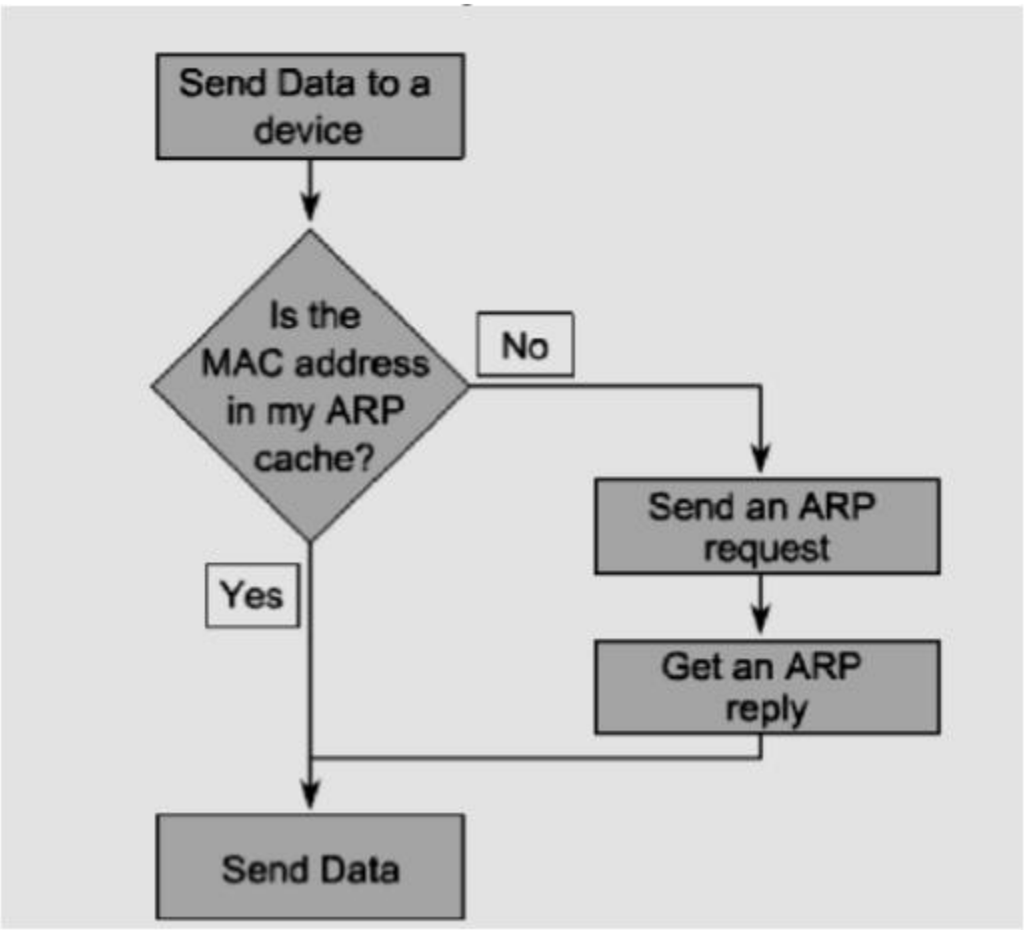
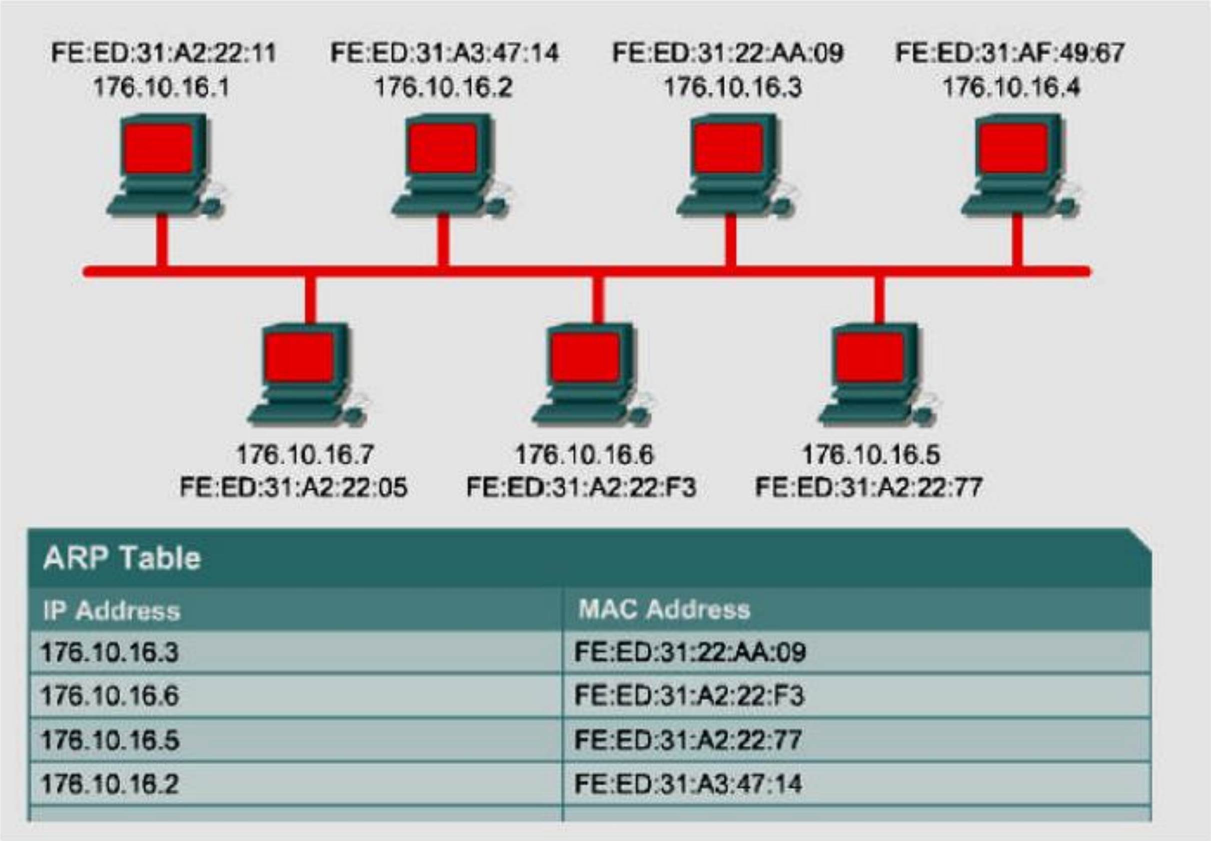


Figure 45 The ARP Process

The computer that requires an IP and MAC address pair broadcasts an ARP request. All the other devices on the LAN analyze this request. If one of the local devices matches the IP address of the request, it sends back an ARP reply that contains its IP-MAC pair. If the IP address is for the LAN and the computer does not exist or is turned off, there is no response to the ARP request. In this situation, the source device reports an error. If the request is for a different IP network, there is another process that can be used.



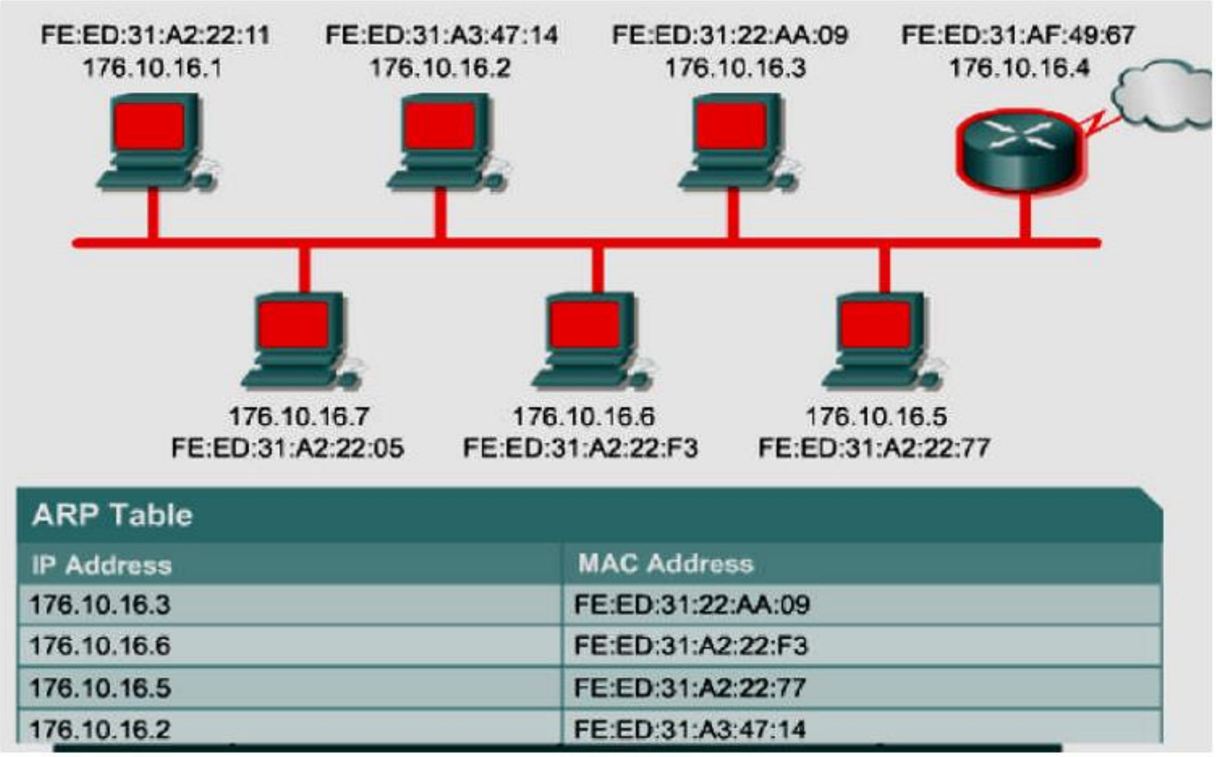
ARP Table

IP Address	MAC Address
176.10.16.3	FE:ED:31:22:AA:09
176.10.16.6	FE:ED:31:A2:22:F3
176.10.16.5	FE:ED:31:A2:22:77
176.10.16.2	FE:ED:31:A3:47:14

Figure 46ARP Request

- Computer 176.10.16.1 needs to send a data transmission to computer 176.10.16.4.
- Computer 176.10.16.1 prepares the data for transmission to computer 176.10.16.4. As it is building the frame for transmission. It finds that the IP-MAC pair for 176.10.16.4 is not in its ARP table. Computer 176.10.16.1 needs this pair, so it must do an ARP request to get it.
- Computer 176.10.16.1 discards the process of encapsulation for the data transmission and instead creates an ARP request to get the MAC address of computer 176.10.16.4.
- Computer 176.10.16.1 transmits the data frames through the network cable segment.
- All computers on the Ethernet segment analyze the incoming data frames to determine if the transmission is for them.

- All computers except computer 176.10.16.4 drop the frames because they do not match the destination IP address of the incoming frames.
- Computer 176.10.16.4 prepares the ARP reply data for transmission.
- Computer 176.10.16.4 transmits its data frames through the Ethernet segment.
- Again all hosts on the segment analyze the incoming frames and add data to their ARP tables.
- Computer 176.10.16.1 prepares the data for transmission.
- Computer 176.10.16.1 transmits its data frames through the Ethernet segment.
- All hosts on the segment analyze the incoming frames.
- All computers except computer 176.10.16.4 drop the frames because they do not match the destination MAC address of the incoming frames.
- Computer 176.10.16.2 prepares the data for transmission.
- Computer 176.10.16.4 processes the data transmission.



ARP Table

IP Address	MAC Address
176.10.16.3	FE:ED:31:22:AA:09
176.10.16.6	FE:ED:31:A2:22:F3
176.10.16.5	FE:ED:31:A2:22:77
176.10.16.2	FE:ED:31:A3:47:14

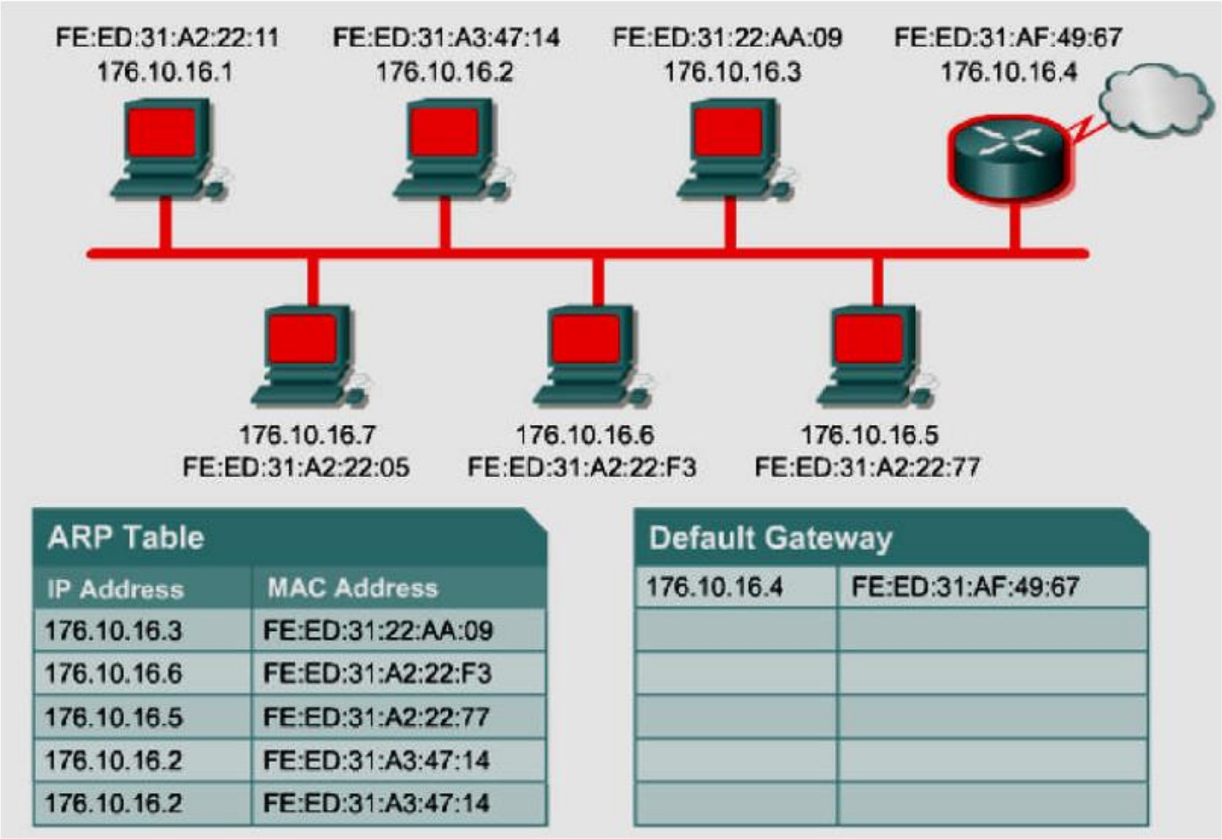
Figure 47 Proxy ARP Request

- Computer 176.10.16.1 needs to send a data transmission to computer 176.10.16.4.
- Computer 176.10.16.1 prepares the data for transmission to computer 176.10.16.4. As it is building the frame for transmission. It finds that the IP-MAC pair for 176.10.16.4 is not in its ARP table. Computer 176.10.16.1 needs this pair, so it must do an ARP request to get it.
- Computer 176.10.16.1 discards the process of encapsulation for the data transmission and instead creates an ARP request to get the MAC address of computer 176.10.16.4.
- Computer 176.10.16.1 transmits the data frames through the network cable segment.
- All computers on the Ethernet segment analyze the incoming data frames to determine if the transmission is for them.

- All devices except router 176.10.16.4 drop the frames because they do not match the destination IP address of the incoming frames.
- Router 176.10.16.4 compares the address with its Ethernet interface IP address. The calculation reveals that this packet is going outside of the LAN. Since this router has proxy ARP enabled, it prepares an ARP reply to the requesting host with its MAC address and the IP address of the destination device.
- Router 176.10.16.4 transmits its data frames through the Ethernet segment.
- Again all hosts on the segment analyze the incoming frames and add data to their ARP tables.
- Computer 176.10.16.1 prepares the data for transmission.
- Computer 176.10.16.1 transmits its data frames through the Ethernet segment.
- All hosts on the segment analyze the incoming frames.
- All computers except computer 176.10.16.4 drop the frames because they do not match the destination MAC address of the incoming frames.
- Router 176.10.16.4 processes the data for transmission to forward to the next network hop.
- Computer 176.10.16.4 processes the data transmission.

Routers do not forward broadcast packets. If the feature is turned on, a router performs a proxy ARP. Proxy ARP is a variation of the ARP protocol. In this variation, a router sends an ARP response with the MAC address of the interface, on which the request was received, to the requesting host. The router responds with the MAC addresses for those requests in which the IP address is not in the range of addresses of the local subnet.

Another method to send data to the address of a device that is on another network segment is to set up a default gateway. The default gateway is a host option where the IP address of the router interface is stored in the network configuration of the host. The source host compares the destination IP address and its own IP address to determine if the two IP addresses are located on the same segment. If the receiving host is not on the same segment, the source host sends the data using the actual IP address of the destination and the MAC address of the router. The MAC address for the router was learned from the ARP table by using the IP address of that router. If the default gateway on the host or the proxy ARP feature on the router is not configured, no traffic can leave the LAN. One or the other is required to have a connection outside of the LAN.



ARP Table

IP Address	MAC Address
176.10.16.3	FE:ED:31:22:AA:09
176.10.16.6	FE:ED:31:A2:22:F3
176.10.16.5	FE:ED:31:A2:22:77
176.10.16.2	FE:ED:31:A3:47:14
176.10.16.2	FE:ED:31:A3:47:14

Default Gateway

176.10.16.4	FE:ED:31:AF:49:67

Figure 48 Default Gateway

- Computer 176.10.16.1 needs to send a data transmission to computer 199.11.20.5.

- Computer 176.10.16.1 prepares the data for transmission to computer 199.11.20.5. As it is builds the frame for transmission. It finds that the IP-MAC pair for 199.11.20.5 is not in its ARP table. With the default gateway set on this computer the destination address is compared with the hosts source address. The calculation shown that the destination is on another network. So the host builds the data frame using the destination IP address and the default gateways MAC address.
- Computer 176.10.16.1 transmits its data frames through the network cable segment.
- All hosts on the segment analyze the incoming frames.
- All computers except for router 176.10.16.4 drop the frames because they do not match the destination MAC address of the incoming frames