Translating Addresses

Reading: Section 4.1 and 9.1

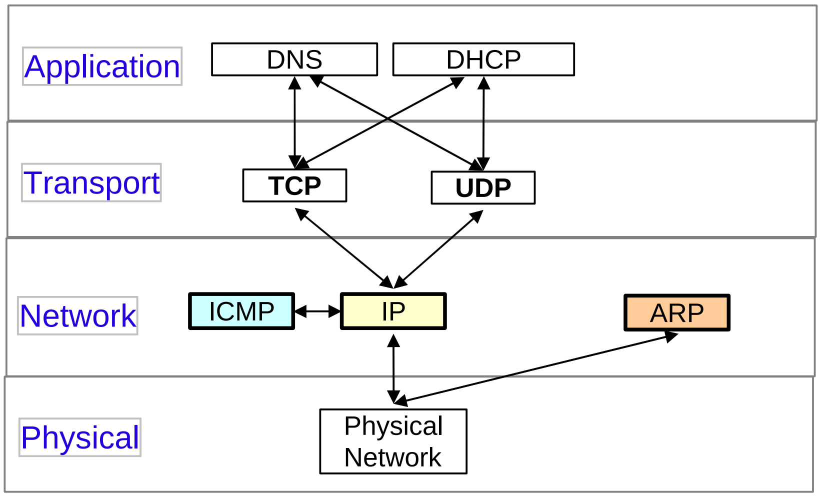
Slides by Rexford @ Princeton, slightly altered by M.D.

Goals of Today's Lecture

- Three different kinds of addresses
  - Host names (e.g., www.cnn.com)
  - IP addresses (e.g., 64.236.16.20)
  - MAC addresses (e.g., 00-15-C5-49-04-A9)
- Protocols for translating between addresses
  - Domain Name System (DNS)
  - Dynamic Host Configuration Protocol (DHCP)
  - Address Resolution Protocol (ARP)
- Two main topics
  - Decentralized management of the name space
  - Boot-strapping an end host that attaches to the 'net

2

Context



3

Separating Names and IP Addresses

- Names are easier (for us!) to remember
  - www.cnn.com vs. 64.236.16.20
- IP addresses can change underneath
  - Move www.cnn.com to 173.15.201.39
  - E.g., renumbering when changing providers
- Name could map to multiple IP addresses
  - www.cnn.com to multiple replicas of the Web site
- Map to different addresses in different places
  - Address of a nearby copy of the Web site
  - E.g., to reduce latency, or return different content
- Multiple names for the same address
  - E.g., aliases like ee.mit.edu and cs.mit.edu

4

Separating IP and MAC Addresses

- LANs are designed for arbitrary network protocols
  - Not just for IP (e.g., IPX, Appletalk, X.25, …)
    - Though now IP is the main game in town
  - Different LANs may have different addressing schemes
    - Though now Ethernet address is the main game in town
- A host may move to a new location
  - So, cannot simply assign a static IP address
    - Since IP addresses depend on network addresses
  - Instead, must reconfigure the adapter
    - To assign it an IP address based on its current location
- Must identify the adapter during bootstrap process
  - Need to talk to the adapter to assign it an IP address

5

Three Kinds of Identifiers

- **Host name** (e.g., www.cnn.com)
  - Mnemonic name appreciated *by humans*
  - Provides little (if any) information about location
  - Hierarchical, variable # of alpha-numeric characters
- **IP address** (e.g., 64.236.16.20)
  - Numerical address appreciated *by routers*
  - Related to host's current location in the topology
  - Hierarchical name space of 32 bits
- **MAC address** (e.g., 00-15-C5-49-04-A9)
  - Numerical address appreciated *within local area network*
  - Unique, hard-coded in the adapter when it is built
  - Flat name space of 48 bits

6

Three Hierarchical Assignment Processes

- **Host name:** www.cs.princeton.edu
  - Domain: registrar for each top-level domain (e.g., .edu)
  - Host name: local administrator assigns to each host
- **IP addresses:** 128.112.7.156
  - Prefixes: ICANN, regional Internet registries, and ISPs
  - Hosts: static configuration, or dynamic using DHCP
- **MAC addresses:** 00-15-C5-49-04-A9
  - Blocks: assigned to vendors by the IEEE
  - Adapters: assigned by the vendor from its block

7

Mapping Between Identifiers

- Domain Name System (DNS)
  - Given a host name, provide the IP address
  - Given an IP address, provide the host name
- Dynamic Host Configuration Protocol (DHCP)
  - Given a MAC address, assign a unique IP address
  - … and tell host other stuff about the Local Area Network
  - To automate the boot-strapping process
- Address Resolution Protocol (ARP)
  - Given an IP address, provide the MAC address
  - To enable communication within the Local Area Network

8

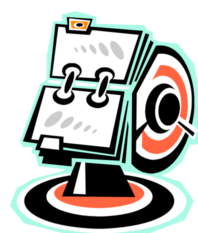
Domain Name System (DNS)

Proposed in 1983 by Paul Mockapetris

Slides by Rexford @ Princeton, slightly altered by M.D.

Outline: Domain Name System

- Computer science concepts underlying DNS
  - Indirection: names in place of addresses
  - Hierarchy: in names, addresses, and servers
  - Caching: of mappings from names to/from addresses
- DNS software components
  - DNS resolvers
  - DNS servers
- DNS queries
  - Iterative queries
  - Recursive queries
- DNS caching based on time-to-live (TTL)



10

Strawman Solution #1: Local File

- Original name to address mapping
  - – Flat namespace
  - – /etc/hosts
  - – SRI kept main copy
  - – Downloaded regularly
- Count of hosts was increasing: moving from a machine per domain to machine per user
  - – Many more downloads
  - – Many more updates

11

Strawman Solution #2: Central Server

- Central server
  - – One place where all mappings are stored
  - – All queries go to the central server
- Many practical problems
  - – Single point of failure
  - – High traffic volume
  - – Distant centralized database
  - – Single point of update
  - – Does not scale

Need a distributed, hierarchical collection of servers

12

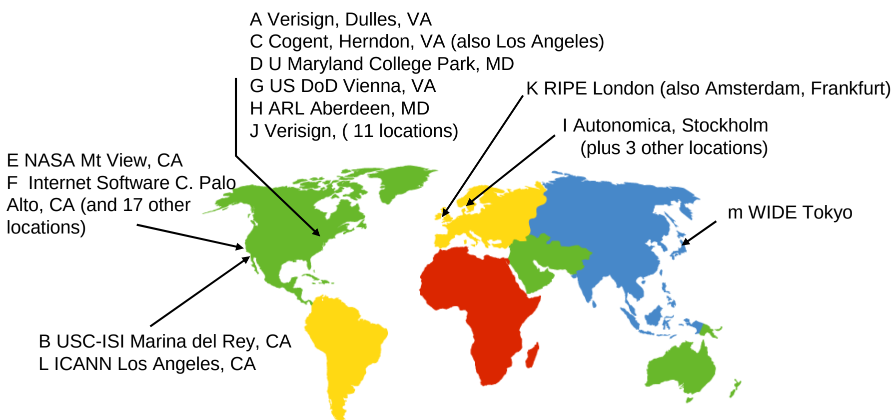
Domain Name System (DNS)

- Properties of DNS
  - Hierarchical name space divided into zones
  - Distributed over a collection of DNS servers
- Hierarchy of DNS servers
  - Root servers
  - Top-level domain (TLD) servers
  - Authoritative DNS servers
- Performing the translations
  - Local DNS servers
  - Resolver software

13

DNS Root Servers

- 13 root servers (see http://www.root-servers.org/)
- Labeled A through M



14

TLD and Authoritative DNS Servers

- Top-level domain (TLD) servers
  - Generic domains (e.g., com, org, edu)
  - Country domains (e.g., uk, fr, ca, jp)
  - Typically managed professionally
    - Network Solutions maintains servers for "com"
    - Educause maintains servers for "edu"
- Authoritative DNS servers
  - Provide public records for hosts at an organization
  - For the organization's servers (e.g., Web and mail)
  - Can be maintained locally or by a service provider

15

Exercise 9.7

- Use the whois utility to find out who is in charge of your site. Look up your site both by DNS name and by IP network number. For the latter you may have to try an alternative whois server (e.g., whois –h whois.arin.net …). Try princeton.edu and cisco.com as well.

16

Exercise 9.7 - A

- whois villanova.edu returns

Domain Name: VILLANOVA.EDU
Registrar: EDUCAUSE
Whois Server: whois.educause.net
Referral URL: http://www.educause.edu/edudomain
Name Server: AUTH1.DNS.COGENTCO.COM
Name Server: NS1.VILLANOVA.EDU
Updated Date: 20-mar-2008
Creation Date: 19-mar-1998
Expiration Date: 19-mar-2009

17

Exercise 9.7 - A

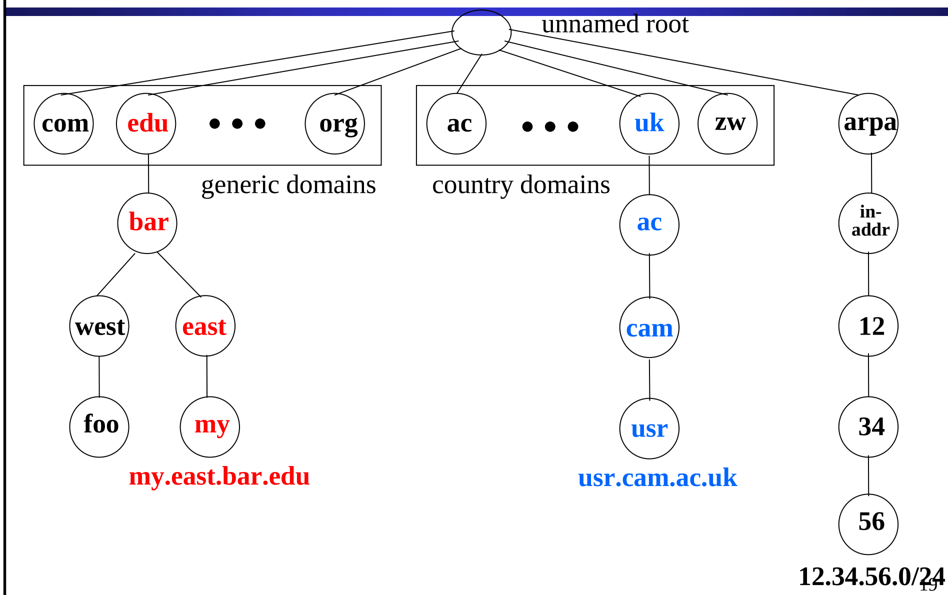
- whois –h whois.arin.net 153.104.0.0 returns

OrgName: Villanova University
OrgID: VILLAN
Address: 800 Lancaster Avenue
City: Villanova
StateProv: PA
PostalCode: 19085
Country: US

NetRange: 153.104.0.0 - 153.104.255.255
CIDR: 153.104.0.0/16
NetName: VILLANOVA
NetHandle: NET-153-104-0-0-1
Parent: NET-153-0-0-0-0
NetType: Direct Assignment
NameServer: NS1.VILLANOVA.EDU

18

Distributed Hierarchical Database



19

Exercise 9.4

- What is the relationship between a domain name (e.g., cs.princeton.edu) and a IP subnet number (e.g., 192.12.69.0)? Do all hosts on the subnet have to be identified by the same name server?

20

Exercise 9.4 - A

There is little if any relationship, formally, between a domain and an IP network, although it is nonetheless fairly common for an organization (or department) to have its DNS server resolve names for all the hosts in its network (or subnet), and no others. The DNS server for cs.princeton.edu could, however, be on a different network entirely (or even on a different continent) from the hosts whose names it resolves.

Alternatively, each x.cs.princeton.edu host could be on a different network, and each host that is on the same network as the cs.princeton.edu nameserver could be in a different DNS domain.

21

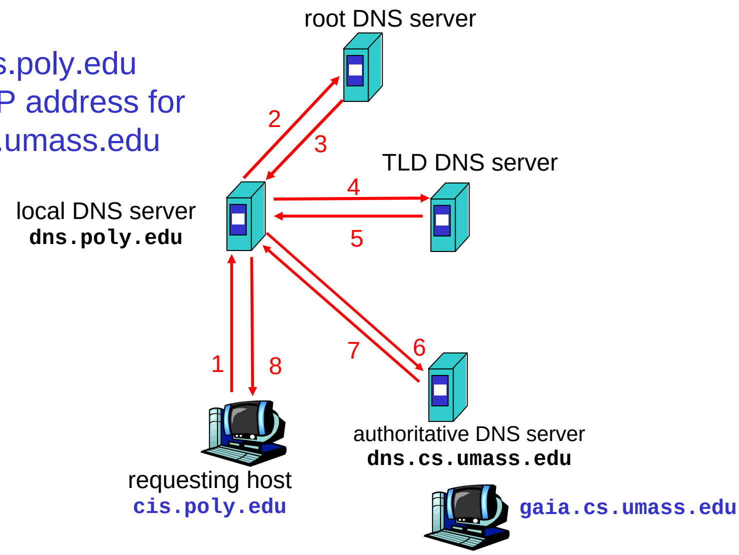
Using DNS

- Local DNS server ("default name server")
  - Usually near the end hosts who use it
  - Local hosts configured with local server (e.g., /etc/resolv.conf) or learn the server via DHCP
- Client application
  - Extract server name (e.g., from the URL)
  - Do *gethostbyname()* to trigger resolver code
- Server application
  - Extract client IP address from socket
  - Optional *gethostbyaddr()* to translate into name

22

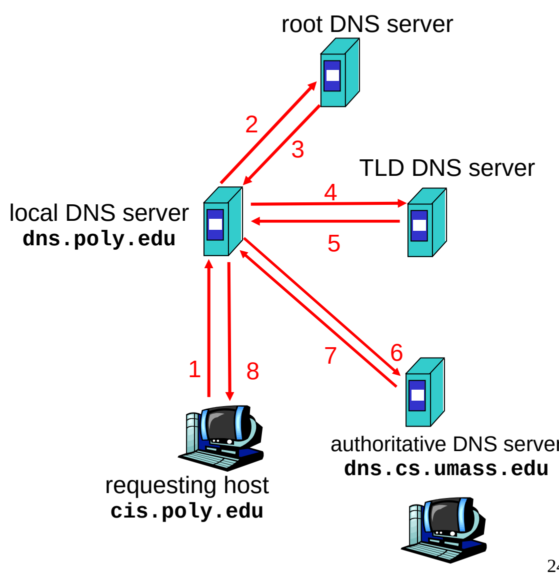
Example

Host at cis.poly.edu wants IP address for gaia.cs.umass.edu



Recursive vs. Iterative Queries

- Recursive query
  - Ask server to get answer for you
  - E.g., request 1 and response 8
- Iterative query
  - Ask server who to ask next
  - E.g., all other request-response pairs



Q

- How do the local DNS servers know the identity of the root servers?

25

Q&A

- How do the local DNS servers know the identity of the root servers?

- *The local DNS servers are statically configured with the identity of the root servers.*

26

Q

- Give two reasons why a local DNS server is not configured with the IP addresses of the top-level domain servers (e.g., for ".edu" or ".org"). How does the local DNS server learn the addresses of the top-level domain servers?

27

Q&A

- Give two reasons why a local DNS server is not configured with the IP addresses of the top-level domain servers (e.g., for ".edu" or ".org"). How does the local DNS server learn the addresses of the top-level domain servers?

- *The top-level domain servers may change over time, and new top-level domains (such as the country codes in recent years) may be added. The local DNS server is configured only with the addresses of the 13 root servers, and contacts one of these servers to learn the address of the TLD server of interest.*

28

DNS Caching

- Performing all these queries take time
  - And all this before the actual communication takes place
  - E.g., 1-second latency before starting Web download
- Caching can substantially reduce overhead
  - The top-level servers very rarely change
  - Popular sites (e.g., www.cnn.com) visited often
  - Local DNS server often has the information cached
- How DNS caching works
  - DNS servers cache responses to queries
  - Responses include a "time to live" (TTL) field
  - Server deletes the cached entry after TTL expires

29

Negative Caching

- Remember things that don't work
  - Misspellings like www.cnn.comm and www.cnnn.com
  - These can take a long time to fail the first time
  - Good to remember that they don't work
  - … so the failure takes less time the next time around

30

DNS Resource Records

DNS: distributed database storing resource records (RR)

RR format: **`(name, value, type, TTL)`**

- Type=A
  - **`name`** is hostname
  - **`value`** is IP address
- Type=NS
  - **`name`** is domain (e.g. foo.com)
  - **`value`** is hostname of authoritative name server for this domain
- Type=CNAME
  - **`name`** is alias name for some “canonical” (the real) name
    `www.ibm.com` is really `servereast.backup2.ibm.com`
  - **`value`** is canonical name
- Type=MX
  - **`value`** is name of mailserver associated with **`name`**

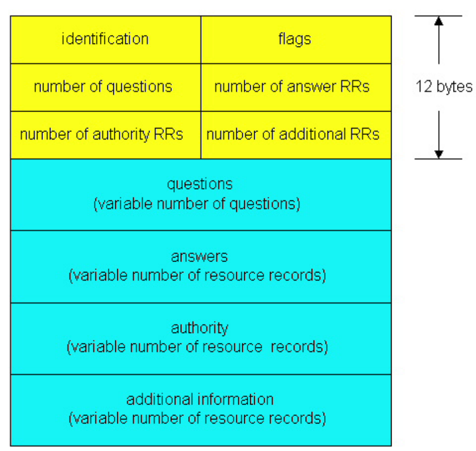
31

DNS Protocol

DNS protocol : *query* and *reply* messages, both with same *message format*

Message header

- Identification: 16 bit # for query, reply to query uses same #
- Flags:
  - Query or reply
  - Recursion desired
  - Recursion available
  - Reply is authoritative



32

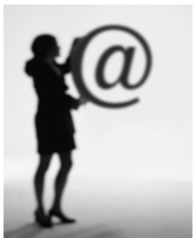
Reliability

- DNS servers are replicated
  - Name service available if at least one replica is up
  - Queries can be load balanced between replicas
- UDP used for queries
  - Need reliability: must implement this on top of UDP
- Try alternate servers on timeout
  - Exponential backoff when retrying same server
- Same identifier for all queries
  - Don't care which server responds

33

Inserting Resource Records into DNS

- Example: just created startup "FooBar"
- Register foobar.com at Network Solutions
  - Provide registrar with names and IP addresses of your authoritative name server (primary and secondary)
  - Registrar inserts two RRs into the .com TLD server:
    - (foobar.com, dns1.foobar.com, NS)
    - (dns1.foobar.com, 212.212.212.1, A)
- Put in authoritative server dns1.foobar.com
  - Type A record for www.foobar.com
  - Type MX record for foobar.com



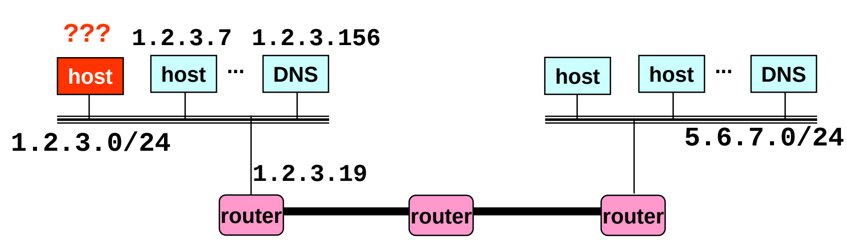
Boot-Strapping an End Host

DHCP and ARP

Slides by Rexford @ Princeton, slightly altered by M.D.

How To Bootstrap an End Host?

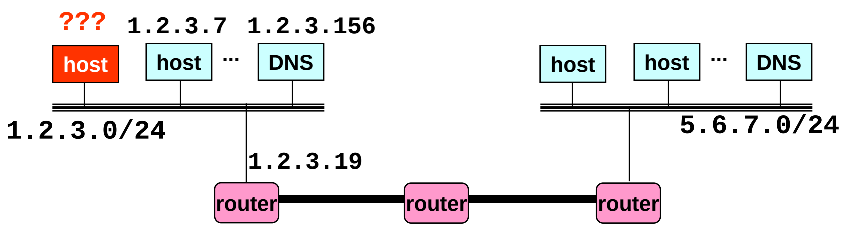
- What local Domain Name System server to use?
- What IP address the host should use?
- How to send packets to remote destinations?
- How to ensure incoming packets arrive?



36

Avoiding Manual Configuration

- Dynamic Host Configuration Protocol (DHCP)
  - End host learns how to send packets
  - Learn IP address, DNS servers, and gateway
- Address Resolution Protocol (ARP)
  - Others learn how to send packets to the end host
  - Learn mapping between IP address & interface address



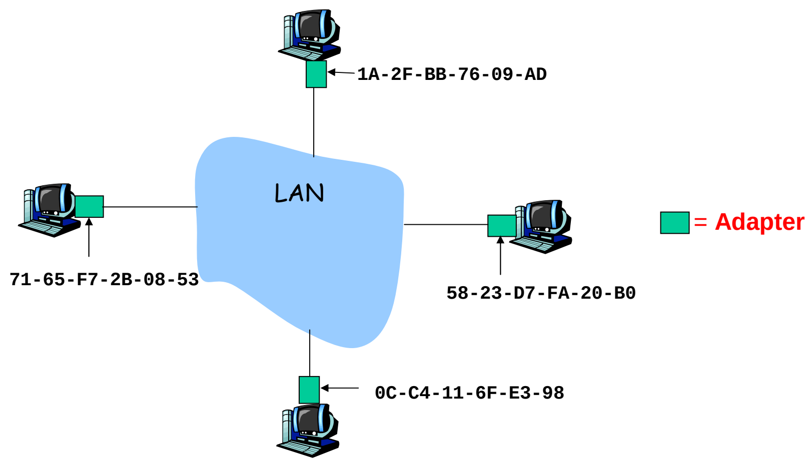
37

Key Ideas in Both Protocols

- **Broadcasting:** when in doubt, shout!
  - Broadcast query to all hosts in the local-area-network
  - … when you don't know how to identify the right one
- **Caching:** remember the past for a while
  - Store the information you learn to reduce overhead
  - Remember your own address & other host's addresses
- **Soft state:** … but eventually forget the past
  - Associate a time-to-live field with the information
  - … and either refresh or discard the information
  - Key for robustness in the face of unpredictable change

38

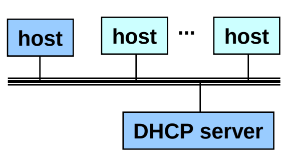
Media Access Control (MAC) Addresses



39

Bootstrapping Problem

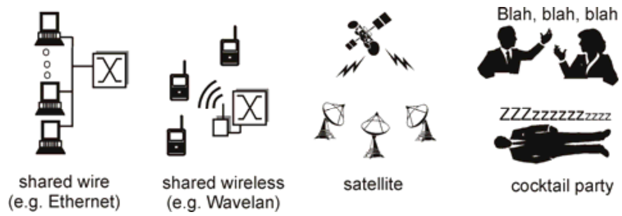
- Host doesn't have an IP address yet
  - So, host doesn't know what source address to use
- Host doesn't know who to ask for an IP address
  - So, host doesn't know what destination address to use
- Solution: shout to discover a server who can help
  - Broadcast a DHCP server-discovery message
  - Server sends a DHCP "offer" offering an address



40

Broadcasting

- Broadcasting: sending to everyone
  - Special destination address: FF-FF-FF-FF-FF-FF
  - All adapters on the LAN receive the packet
- Delivering a broadcast packet
  - Easy on a “shared media”
  - Like shouting in a room – everyone can hear you



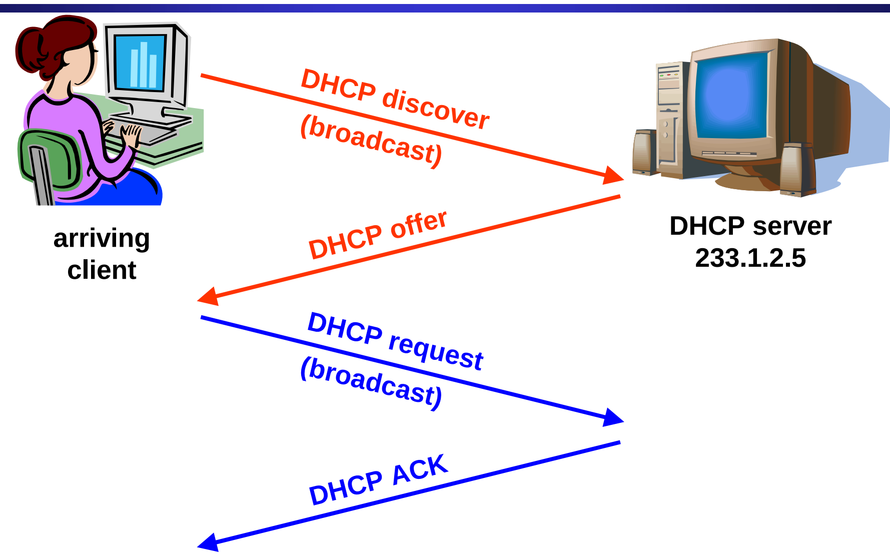
41

Response from the DHCP Server

- DHCP “offer message” from the server
  - Configuration parameters (proposed IP address, mask, gateway router, DNS server, ...)
  - Lease time (the time the information remains valid)
- Multiple servers may respond
  - Multiple servers on the same broadcast media
  - Each may respond with an offer
  - The client can decide which offer to accept
- Accepting one of the offers
  - Client sends a DHCP request echoing the parameters
  - The DHCP server responds with an ACK to confirm
  - … and the other servers see they were not chosen

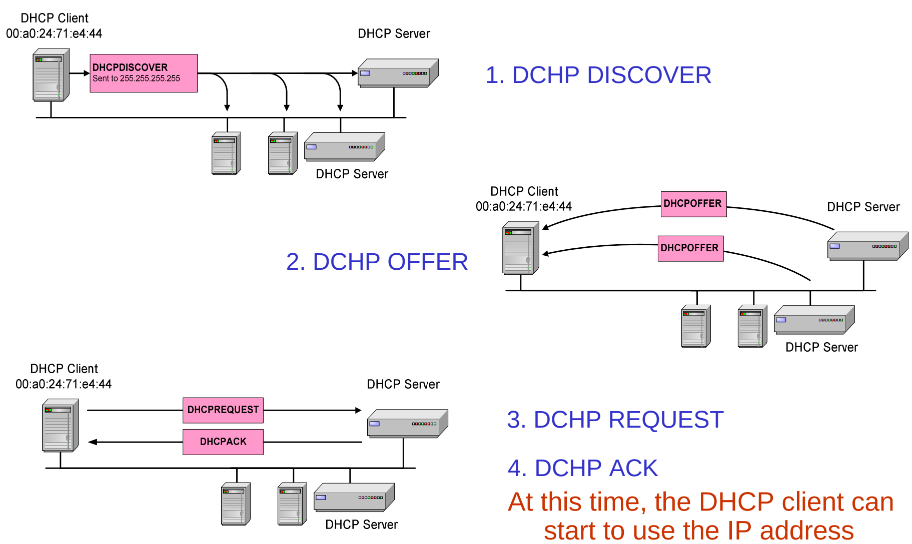
42

Dynamic Host Configuration Protocol



43

DHCP Operation



At this time, the DHCP client can start to use the IP address

44

Deciding What IP Address to Offer

- Server as centralized configuration database
  - All parameters are statically configured in the server
  - E.g., a dedicated IP address for each MAC address
  - Avoids complexity of configuring hosts directly
  - … while still having a permanent IP address per host
- Or, dynamic assignment of IP addresses
  - Server maintains a pool of available addresses
  - … and assigns them to hosts on demand
  - Leads to less configuration complexity
  - … and more efficient use of the pool of addresses
  - Though, it is harder to track the same host over time

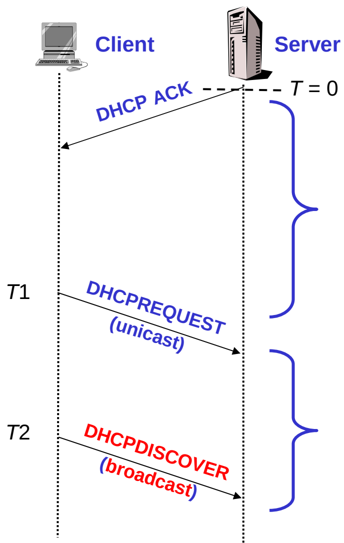
45

Soft State: Refresh or Forget

- Why is a lease time necessary?
  - Client can release the IP address (DHCP RELEASE)
    - E.g., "ipconfig /release" at the DOS prompt
    - E.g., clean shutdown of the computer
  - But, the host might not release the address
    - E.g., the host crashes (blue screen of death!)
    - E.g., buggy client software
  - And you don't want the address to be allocated forever
- Performance trade-offs
  - Short lease time: returns inactive addresses quickly
  - Long lease time: avoids overhead of frequent renewals

46

DHCP Lease Renewal



Typical lease time is 1 - 30 days.

At T1 = 1/2 Lease-time (default) client sends a *unicast* DHCP REQUEST to the server to renew the lease

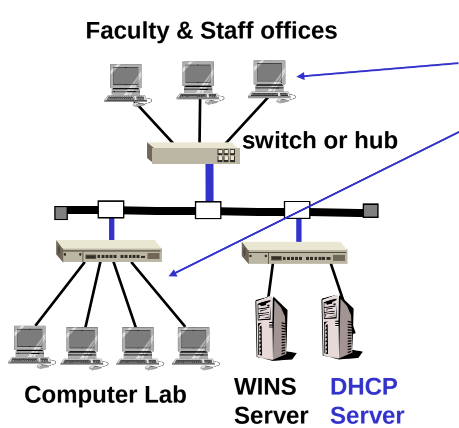
If no response from server:

At T2 = 3/4 Lease-time (default) client begins to *broadcast* DHCP DISCOVER to locate a new server.

47

DHCP on Single Network

One DHCP server can configure all machines on a network.



DHCP provides each client:

IP Address: 153.104.2xx.x

Netmask: 255.255.252.0

Router: 153.104.200.1

Broadcast addr: 153.104.103.255

WINS server: 153.104.6.55

Domain-name: csc.villanova.edu

DNS servers: 153.104.136.2, …

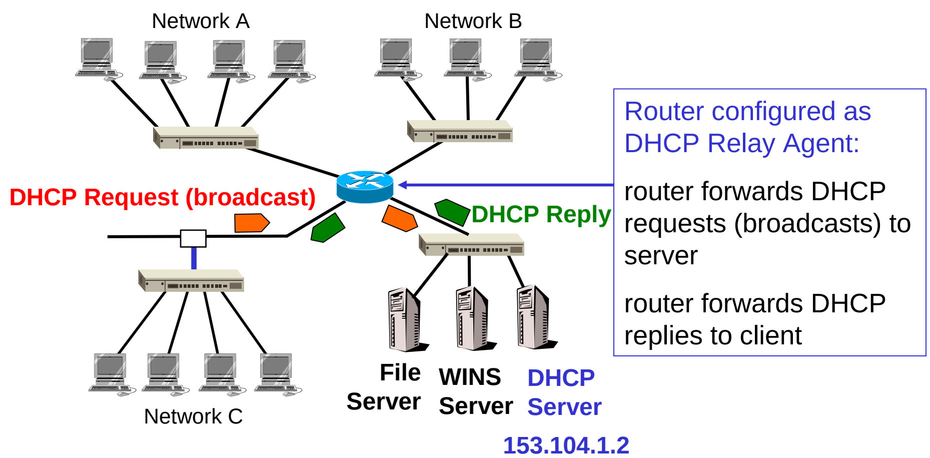
Lease Time: 10 days

Other information

48

DHCP on Multiple Networks

One server can manage several networks using a *Relay Agent*.

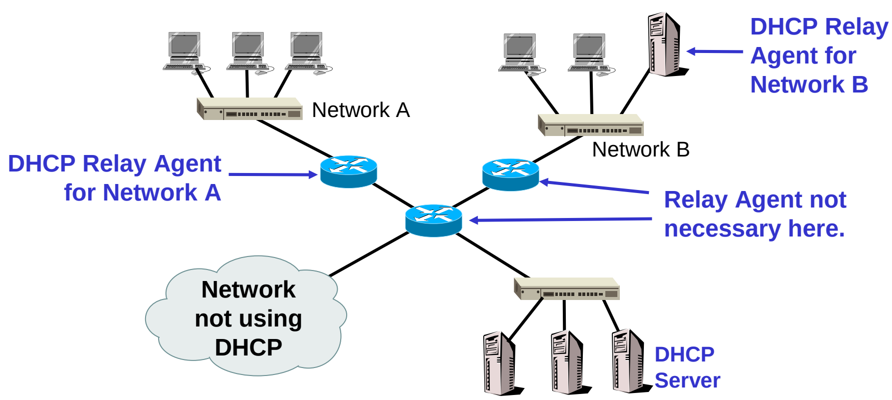


No client reconfiguration is needed to use a relay agent.

49

DHCP on Remote Networks

There can be other routers between the DCHP Relay Agent and the DHCP server.



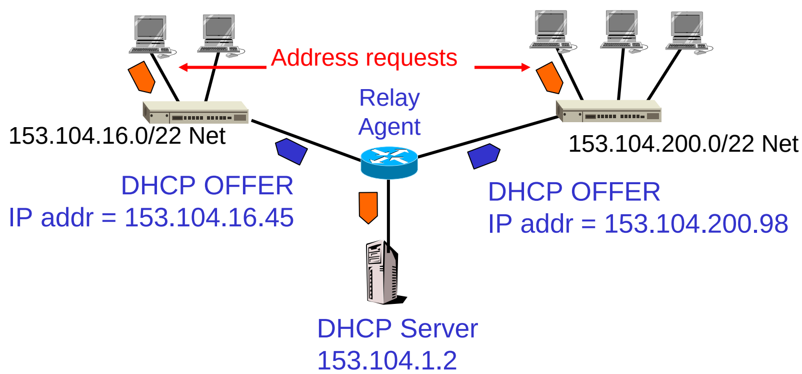
DHCP relaying is transparent to the clients -- no extra client configuration is needed to use a relay agent.

50

Where is the Client?

A DHCP Server can support multiple networks. Assigned IP addresses must match the clients' network.

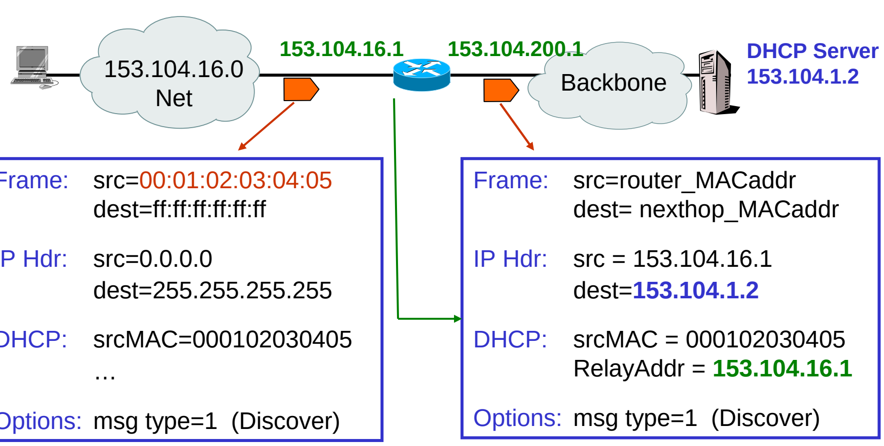
How does the server know which network the client is on?



51

Relay Agent Operation (1)

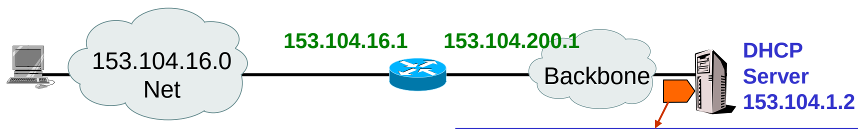
The Relay Agent inserts the IP address of its own interface into the broadcast request from the client, then forwards the request to a DHCP server.



52

Relay Agent Operation (2)

The DHCP server uses the RelayAddr value in the DHCP message to derive the client's network address (153.104.16.0)



RelayAddr value indicates which network the client is on. This address is also used to reply.

srcMAC (client hardware addr) in DHCP message identifies the client.

Frame: src=router_MACaddr
dest= server_MACaddr

IP Hdr: src= 153.104.16.1
dest=**153.104.1.2**

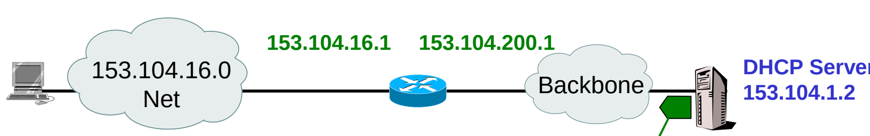
DHCP: RelayIP = **153.104.16.1**
srcMAC = 000102030405

Options: msg type=1 (Discover)

53

Relay Agent Operation (3)

The DHCP server sends a reply addressed to the relay agent, using the RelayAddr as the destination IP address.



Server sends a *unicast* reply to the relay agent, using the RelayAddr value.

Offered IP address and other data for the DHCP client.

Server returns original RelayAddr and srcMAC values to assist in relaying.

Frame: src=server_MACaddr
dest= next_hop_MACaddr

IP Hdr: src= **153.104.1.2**
dest= **153.104.16.1**

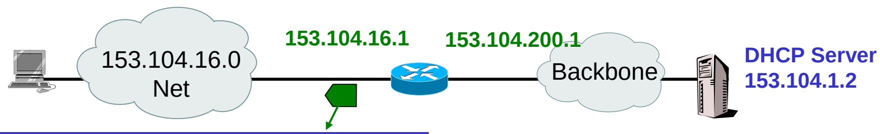
DHCP: RelayIP = **153.104.16.1**
OfferIP= 153.104.16.45
srcMAC = 000102030405

Options: msg type=2 (Offer)

54

Relay Agent Operation (4)

When the relay agent receives a reply with its own IP address, it relays the reply to the client.



Frame: src= router-MACaddr
dest= 00:01:02:03:04:05

IP Hdr: src=**153.104.1.2**
dest=255.255.255.255

DHCP: RelayIP = **153.104.16.1**
OfferIP = 153.104.16.45
srcMAC = 000102030405

Options: msg type=2 (Offer)

Agent copies client's hardware address from the srcMAC field to the destination field

55

DHCP Performance

- Network traffic from DHCP is *light*. If lease time is 1 day, there would be 8 packets/client/day for DHCP
- A server handles *thousands of clients*, according to *The DHCP Handbook*
- Memory: a server managing 10,000 client leases would consume up to 20MB of memory for the lease table

56

Note

- DNS cannot be used for DHCP configured hosts.

57

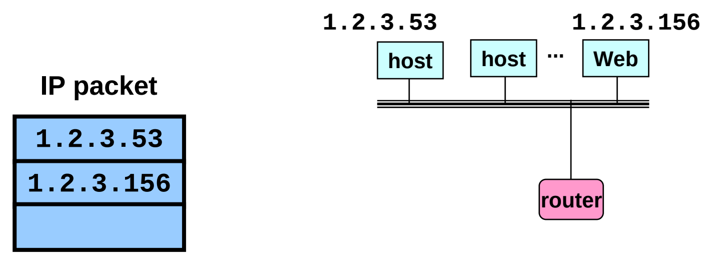
So, Now the Host Knows Things

- IP address
- Mask
- Gateway router
- DNS server
- …

- And can send packets to other IP addresses
  - But, how to learn the MAC address of the destination?

58

Sending Packets Over a Link



- Adapters only understand MAC addresses
  - Translate the destination IP address to MAC address
  - Encapsulate the IP packet inside a link-level frame

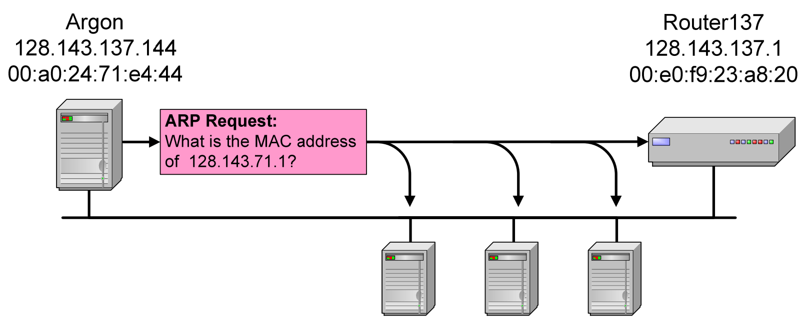
59

Address Resolution Protocol

- Every node maintains an ARP table
  - (IP address, MAC address) pair
- Consult the table when sending a packet
  - Map destination IP address to destination MAC address
  - Encapsulate and transmit the data packet
- But, what if the IP address is not in the table?
  - Sender broadcasts: “Who has IP address 1.2.3.156?”
  - Receiver responds: “MAC address 58-23-D7-FA-20-B0”
  - Sender caches the result in its ARP table
  - Receiver caches the sender info in its ARP table
  - Cache table fills up => Least Recently Used policy used

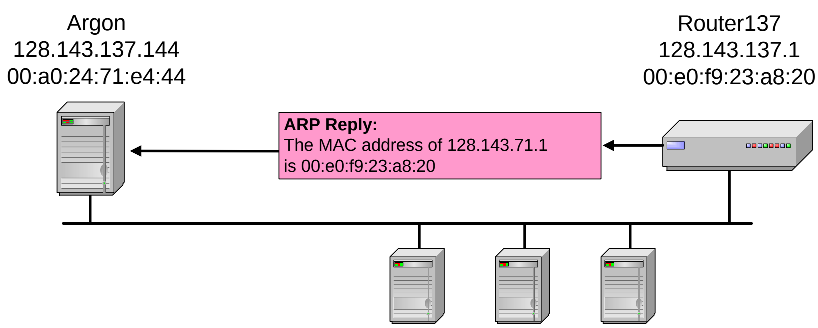
60

ARP Request



61

ARP Reply



62

ARP Table

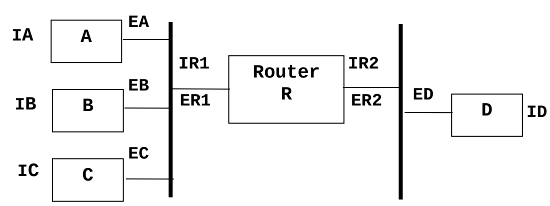
- Use

  arp –a

  to view the ARP cache on your own machine.

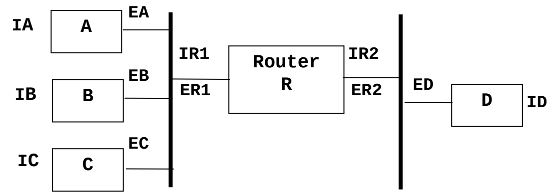
- Command available in both Unix and Windows

ARP Exercise



- IA, IB, … are IP addresses; EA, EB, … are MAC addresses. Assume that all ARP tables are initially empty. Give the complete contents of the ARP table for A, B, C, D and R after each of the following operations.

ARP Exercise (1)

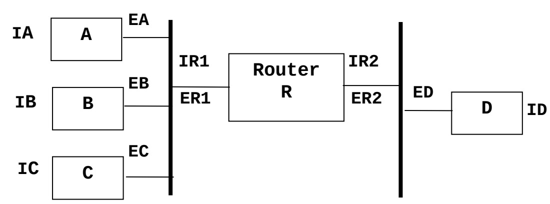


1) Host A sends a packet to host C

ARP Table of A

ARP Table of C

ARP Exercise (2)



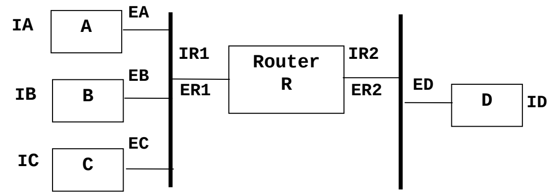
2) Host B sends a packet to host D

ARP Table of B

ARP Table of D

ARP Table of R

ARP Exercise (3)



3) Host D sends a packet to host B

Q

- How could the ARP protocol be used to determine if another host on my network is using my IP address?

- Ans: Send an ARP broadcast with my IP address. If another host responds, we have identified a
- host using my IP address. Recall that only the owner of an IP address should respond.

Q&A

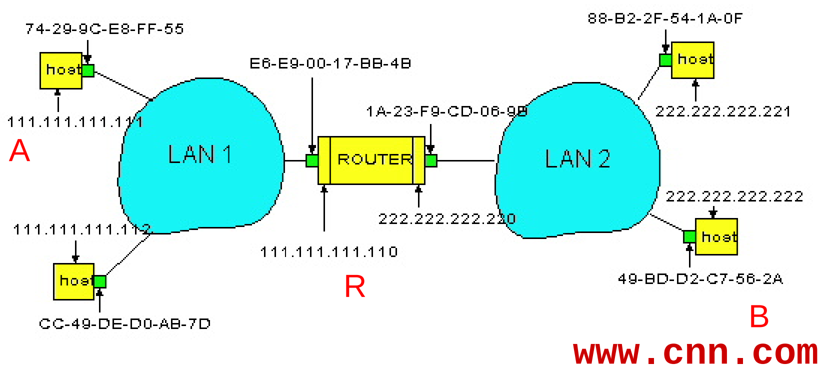
- How could the ARP protocol be used to determine if another host on my network is using my IP address?

- *Send an ARP broadcast with my IP address. If another host responds, we have identified a host using my IP address. Recall that only the owner of an IP address should respond.*

69

Putting it Together: A Sending a Packet to B

How does host A send an IP packet to B (www.cnn.com)?

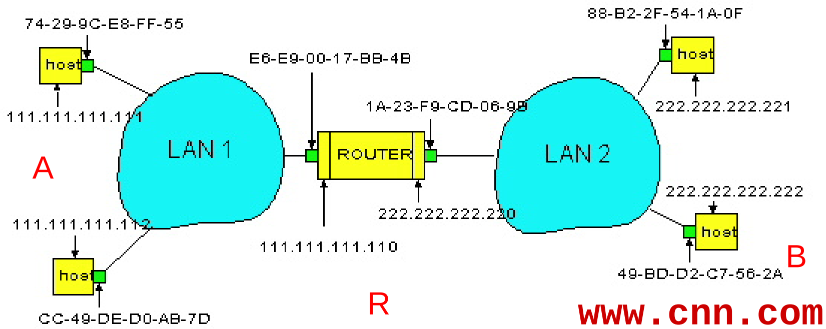


A sends packet to R, and R sends packet to B.

70

Basic Steps

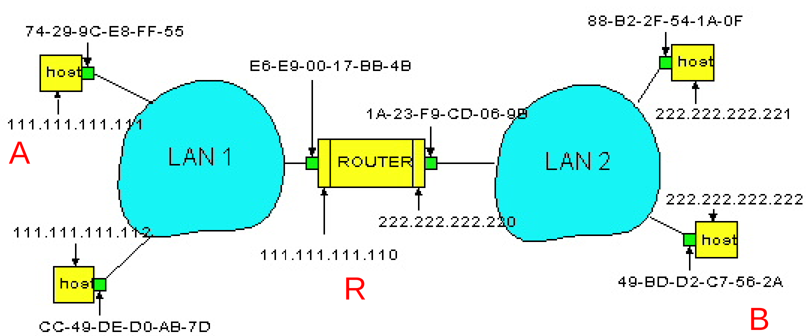
- Host A must learn the IP address of B via DNS
- Host A uses gateway R to reach external hosts
- Host A sends the frame to R's MAC address
- Router R forwards IP packet to outgoing interface
- Router R learns B's MAC address and forwards frame



71

Host A Learns the IP Address of B

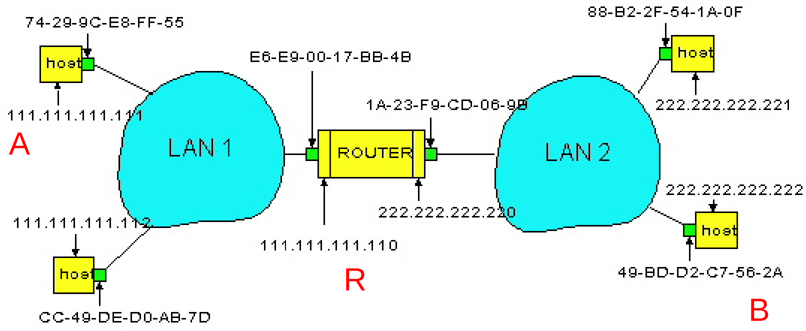
- Host A does a DNS query to learn B's address
  - Suppose gethostbyname() returns 222.222.222.222
- Host A constructs an IP packet to send to B
  - Source 111.111.111.111, destination 222.222.222.222



72

Host A Learns the IP Address of B

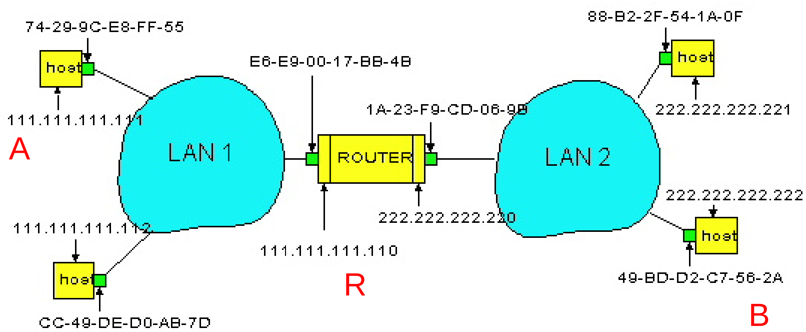
- IP header
  - From A: 111.111.111.111
  - To B: 222.222.222.222
- Ethernet frame
  - From A: 74-29-9C-E8-FF-55
  - To gateway: ????



73

Host A Decides to Send Through R

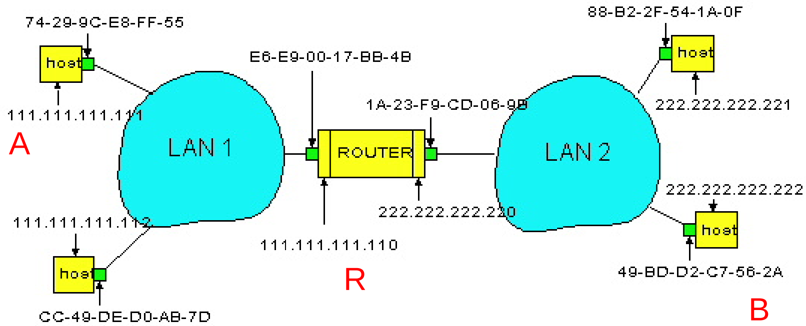
- Host A has a gateway router R
  - Used to reach destinations outside of 111.111.111.0/24
  - Address 111.111.111.110 for R learned via DHCP
- But, what is the MAC address of the gateway?



74

Host A Sends Packet Through R

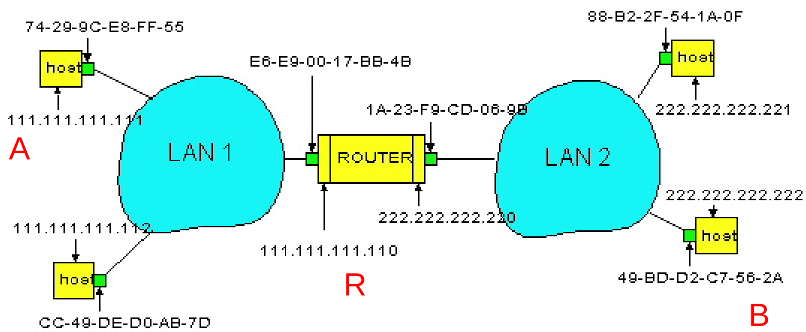
- Host A learns the MAC address of R's interface
  - ARP request: broadcast request for 111.111.111.110
  - ARP response: R responds with E6-E9-00-17-BB-4B
- Host A encapsulates the packet and sends to R



75

Host A Sends Packet Through R

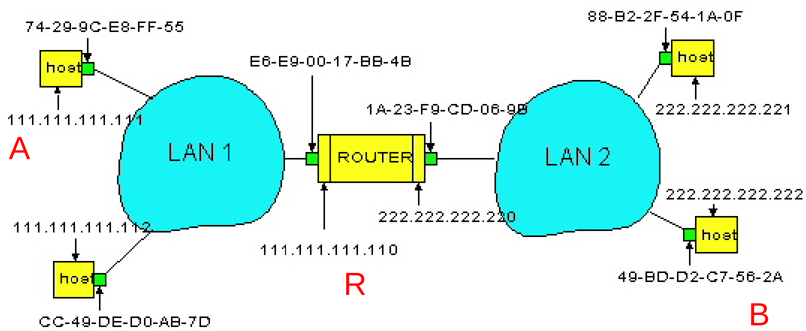
- IP header
  - From A: 111.111.111.111
  - To B: 222.222.222.222
- Ethernet frame
  - From A: 74-29-9C-E8-FF-55
  - To R: E6-E9-00-17-BB-4B



76

R Decides how to Forward Packet

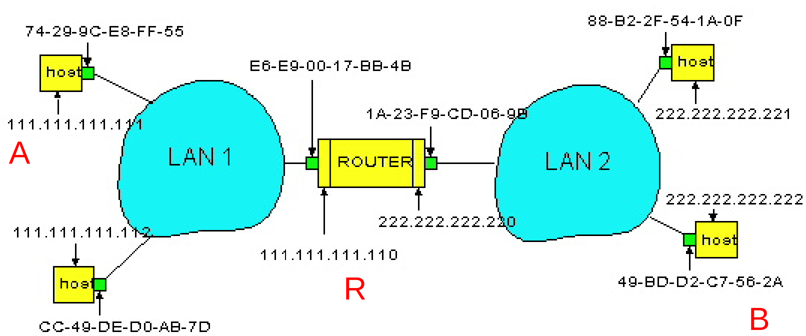
- Router R's adapter receives the packet
  - R extracts the IP packet from the Ethernet frame
  - R sees the IP packet is destined to 222.222.222.222
- Router R consults its forwarding table
  - Packet matches 222.222.222.0/24 via other adapter



77

Router R Wants to Forward Packet

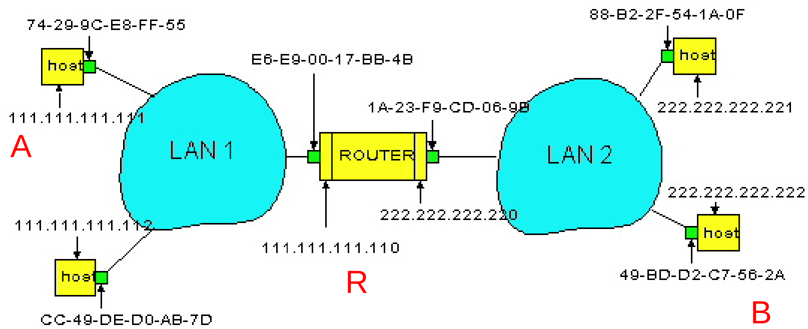
- IP header
  - From A: 111.111.111.111
  - To B: 222.222.222.222
- Ethernet frame
  - From R: 1A-23-F9-CD-06-9B
  - To B: ???



78

R Sends Packet to B

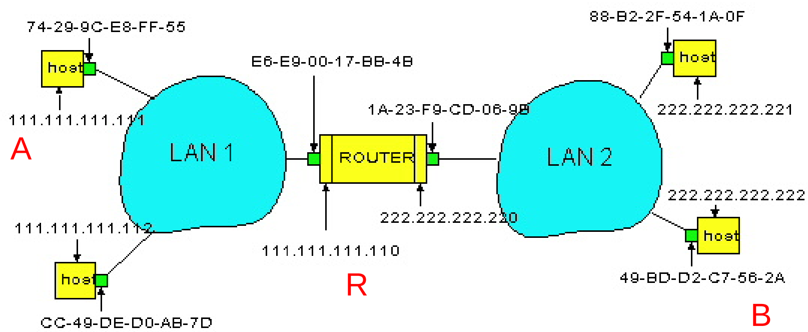
- Router R's learns the MAC address of host B
  - ARP request: broadcast request for 222.222.222.222
  - ARP response: B responds with 49-BD-D2-C7-56-2A
- Router R encapsulates the packet and sends to B



79

Router R Wants to Forward Packet

- IP header
  - From A: 111.111.111.111
  - To B: 222.222.222.222
- Ethernet frame
  - From R: 1A-23-F9-CD-06-9B
  - To B: 49-BD-D2-C7-56-2A



80

Q

- Suppose a rogue computer sends an unsolicited ARP reply message, with its own MAC address and the IP address of the LAN gateway router, to another host on the LAN. What will happen when this host (i.e., the host who believed the bogus ARP reply) transmits an IP packet destined to an external Internet address?

81

Q&A

- Suppose a rogue computer sends an unsolicited ARP reply message, with its own MAC address and the IP address of the LAN gateway router, to another host on the LAN. What will happen when this host (i.e., the host who believed the bogus ARP reply) transmits an IP packet destined to an external Internet address?

- *The host would send the packet in an Ethernet frame destined to the MAC address of the rogue machine, which may discard the traffic, or may inspect the contents before directing the packet to the legitimate gateway.*

82

Exercise 9.1

- ARP and DNS both depend on caches. ARP cach entry lifetimes are typically 10 mins, while DNS cache is on the order of days. Justify this difference.

83

Exercise 9.1 - A

ARP traffic is always local, so ARP retransmissions are confined to a small area. Subnet broadcasts every few minutes are not a major issue either in terms of bandwidth or CPU, so a small cache lifetime does not create an undue burden.

Much of DNS traffic is nonlocal; limiting such traffic becomes more important for congestion reasons alone. There is also a sizable total CPU-time burden on the root name servers. And an active web session can easily generate many more DNS queries than ARP queries.

If the DNS cache-entry lifetime is too long, however, then when a host's IP address changes the host is effectively unavailable for a prolonged interval.

84

85

Conclusion

- Domain Name System
  - Distributed, hierarchical database
  - Distributed collection of servers
  - Caching to improve performance
- Bootstrapping an end host
  - Dynamic Host Configuration Protocol (DHCP)
  - Address Resolution Protocol (ARP)
- Next class: Middleboxes
  - Reading: Section 8.4 and Ch. 2
  - Network Address Translator (NAT)
  - Firewalls

86