CSE 410/565: Computer Security

Instructor: Dr. Ziming Zhao

Today

1. Network security

What is a Computer Network

A computer network consists of two or more computers that are linked in order to share resources, exchange files, or allow electronic communications

Computer network

- Hardware
- Software

What is the Internet

A network of networks, joining many government, university and private computers together and providing an infrastructure for the use of E-mail, bulletin boards, file archives, hypertext documents, databases and other computational resources.

The **vast collection of computer networks** which form and **act as a single huge network** for transport of data and messages across distances which can be anywhere from the same office to anywhere in the world.

Written by William F. Slater, III
1996
President of the Chicago Chapter of the Internet Society

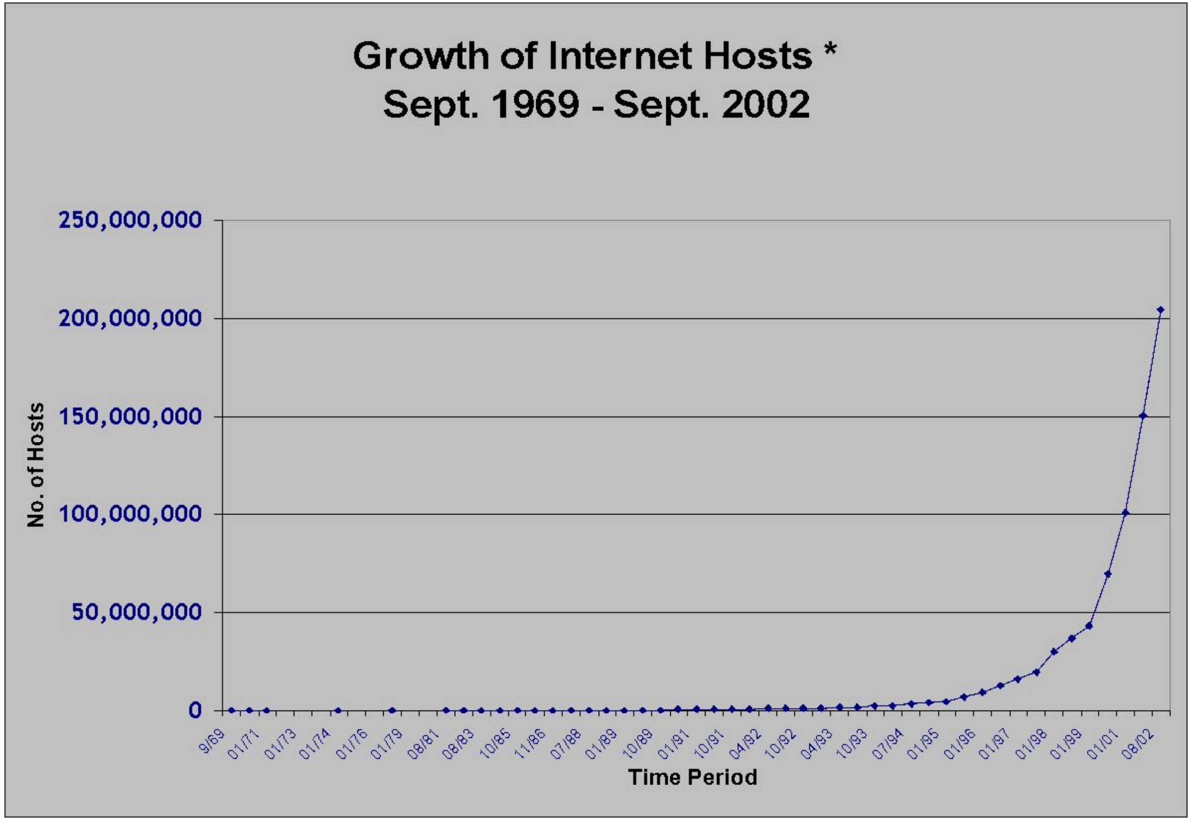
Brief History of the Internet

- 1968 - DARPA (Defense Advanced Research Projects Agency) contracts with BBN (Bolt, Beranek & Newman) to create ARPAnet
- 1970 - First five nodes:
    - UCLA
    - Stanford
    - UC Santa Barbara
    - U of Utah, and
    - BBN
- 1974 - TCP specification by Vint Cerf (Turing Award 2004)
- 1984 – On January 1, the Internet with its 1000 hosts converts en masse to using TCP/IP for its messaging

Internet Growth Trends

- 1977: 111 hosts on Internet
- 1981: 213 hosts
- 1983: 562 hosts
- 1984: 1,000 hosts
- 1986: 5,000 hosts
- 1987: 10,000 hosts
- 1989: 100,000 hosts
- 1992: 1,000,000 hosts
- 2001: 150 – 175 million hosts
- 2002: over 200 million hosts
- 2012: 8.7 billion [forbes.com]
- 2020: forty billion devices [forbes.com]

Internet Growth Trends



Copyright belongs to William F. Slater, III

Computer Network Hardware

- Network Hosts
- Communication Mediums
- Networking Devices

Network Host

A network host is a computer connected to a computer network.

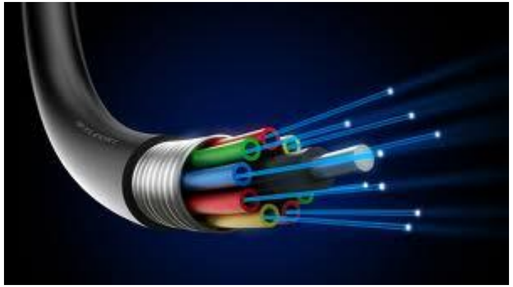
A network host may request/offer information resources, services, and applications from/to other hosts on the network.

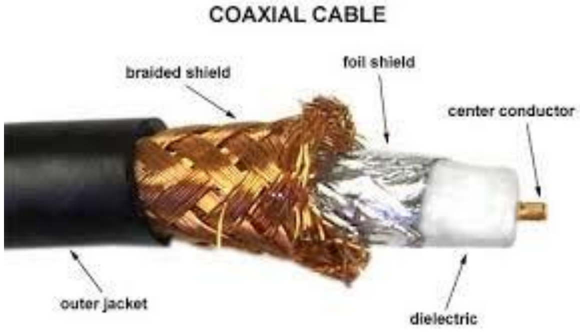
Server

A server is a system (hardware and software) that responds to requests across a computer network to provide, or help to provide, a network service.

Communication Mediums In Networks

The computers on a network may be linked through cables, telephone lines, radio waves, satellites, or infrared light beams.





Networking Devices

- Hub
- Switch
- Router

Computer Network Software

- **Network Protocols** (The language)
- Software on hosts that speak and understand network languages
  - Network Adapter Driver (Kernel)
  - TCP/IP Layer Software (Kernel)
  - Socket Functions (Kernel)
  - User space Libraries
- Software on Networking Devices
  - Cisco IOS: a family of software used on most Cisco Systems routers and current Cisco network switches

OSI Model

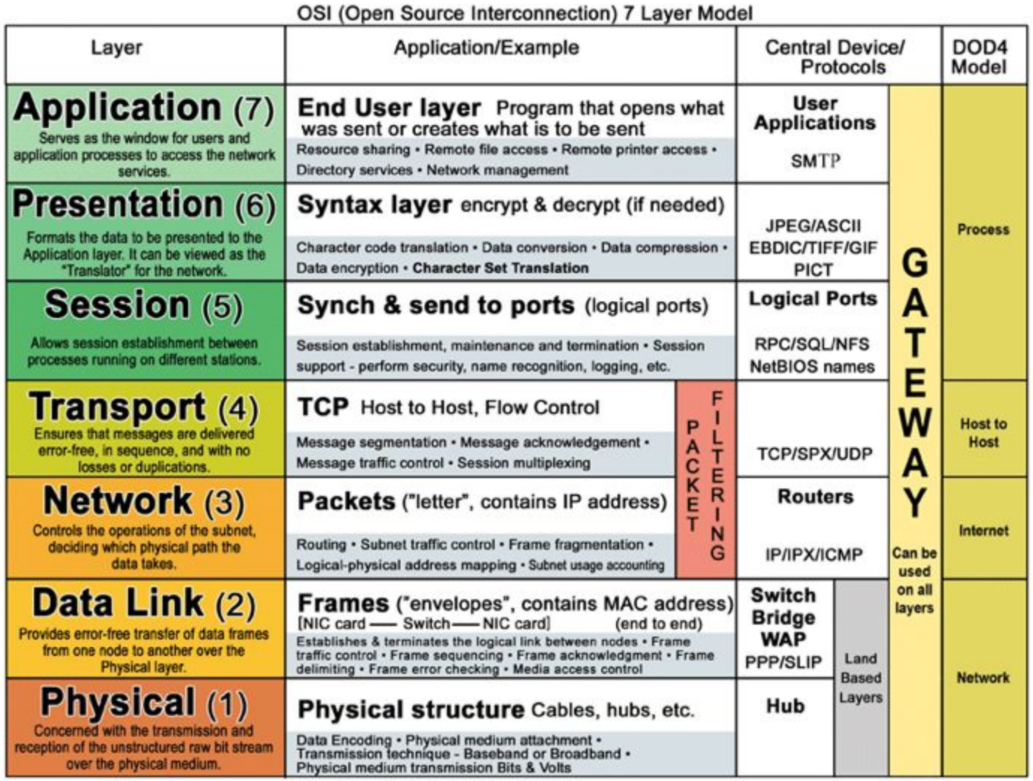
- OSI - Open Systems Interconnection

- OSI model was first introduced in 1984 by the International Organization for Standardization (ISO).
    - Outlines **WHAT** needs to be done to send data from one computer to another.
    - Not **HOW** it should be done.

OSI Model

- The OSI model
    - is a **theoretical blueprint** that helps us understand how data gets from one user's computer to another.
    - It is also a model that helps **develop standards** so that all of our hardware and software talks nicely to each other.
    - It aids standardization of networking technologies by providing an organized structure for hardware and software developers to follow, to insure there products are compatible with current and future technologies.

OSI 7 Layer Model (1984)



OSI (Open Source Interconnection) 7 Layer Model

Layer	Application/Example		Central Device/ Protocols			DOD4 Model
Application (7) Serves as the window for users and application processes to access the network services.	**End User layer** Program that opens what was sent or creates what is to be sent Resource sharing • Remote file access • Remote printer access • Directory services • Network management		**User Applications** SMTP		**GATEWAY** **Can be used on all layers**	**Process**
Presentation (6) Formats the data to be presented to the Application layer. It can be viewed as the "Translator" for the network.	**Syntax layer** encrypt & decrypt (if needed) Character code translation • Data conversion • Data compression • Data encryption • **Character Set Translation**		JPEG/ASCII EBDIC/TIFF/GIF PICT			
Session (5) Allows session establishment between processes running on different stations.	**Synch & send to ports** (logical ports) Session establishment, maintenance and termination • Session support - perform security, name recognition, logging, etc.		**Logical Ports** RPC/SQL/NFS NetBIOS names			
Transport (4) Ensures that messages are delivered error-free, in sequence, and with no losses or duplications.	**TCP** Host to Host, Flow Control Message segmentation • Message acknowledgement • Message traffic control • Session multiplexing	PACKET FILTERING	TCP/SPX/UDP			**Host to Host**
Network (3) Controls the operations of the subnet, deciding which physical path the data takes.	**Packets** ("letter", contains IP address) Routing • Subnet traffic control • Frame fragmentation • Logical-physical address mapping • Subnet usage accounting		**Routers** IP/IPX/ICMP			**Internet**
Data Link (2) Provides error-free transfer of data frames from one node to another over the Physical layer.	**Frames** ("envelopes", contains MAC address) [NIC card —— Switch —— NIC card] (end to end) Establishes & terminates the logical link between nodes • Frame traffic control • Frame sequencing • Frame acknowledgment • Frame delimiting • Frame error checking • Media access control		**Switch** **Bridge** **WAP** PPP/SLIP	Land Based Layers		**Network**
Physical (1) Concerned with the transmission and reception of the unstructured raw bit stream over the physical medium.	**Physical structure** Cables, hubs, etc. Data Encoding • Physical medium attachment • Transmission technique - Baseband or Broadband • Physical medium transmission Bits & Volts		**Hub**			

Open Systems Interconnection (OSI) Model

- Physical Layer: Physical properties of the various communications media.
- Data Link Layer: Logical organization of data bits transmitted on a particular medium.
- Network Layer: Defines the addressing and routing structure of the network.
- Transport Layer: Defines if / how retransmissions will be used to ensure data delivery end to end.
- Session Layer: Describes how request and reply packets are paired in a remote procedure call.
- Presentation Layer: Syntax of data being transferred.
- Application Layer: Where user and system programs are run

The Postal Analogy

- A- Write a 20 page letter to a foreign country.
- P- Translate the letter so the receiver can read it.
- S- Insure the intended recipient can receive letter.
- T- Separate and number pages. Like registered mail, tracks delivery and requests another package if one is “lost” or “damaged” in the mail.
- N- Postal Center sorting letters by zip code to route them closer to destination.
- D- Local Post Office determining which vehicles to deliver letters.
- P- Physical Trucks, Planes, Rail, autos, etc which carry letter between stations.

TCP/IP Suite Overview

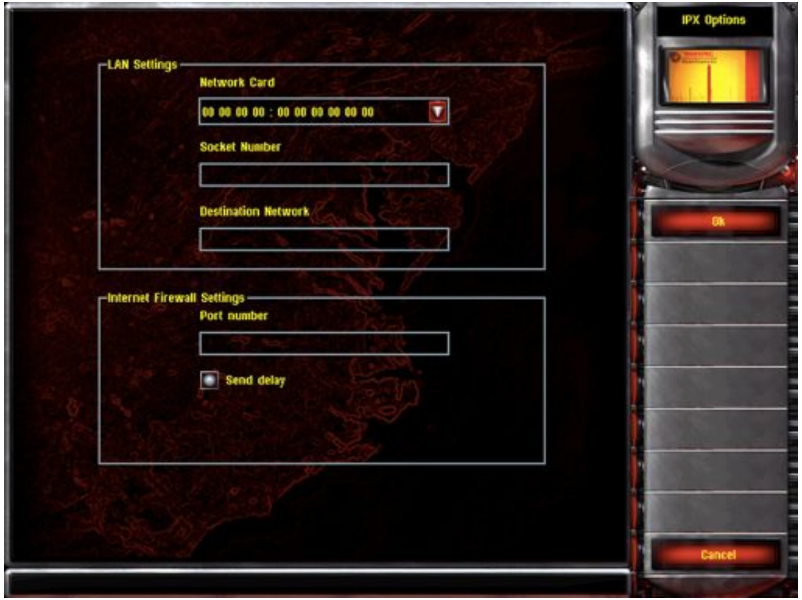
- History
- Compare the reference models
- Protocol Data Unit

Other Protocol Suites

- IBM System Network Architecture, 1974
- Digital Equipment Corporation DECNet, 1975
- International Organization for Standardization model, 1978
- IPX/SPX

Other Protocol Suites

- IPX/SPX



LAN Settings

Network Card

00 00 00 00 : 00 00 00 00 00 00

Socket Number

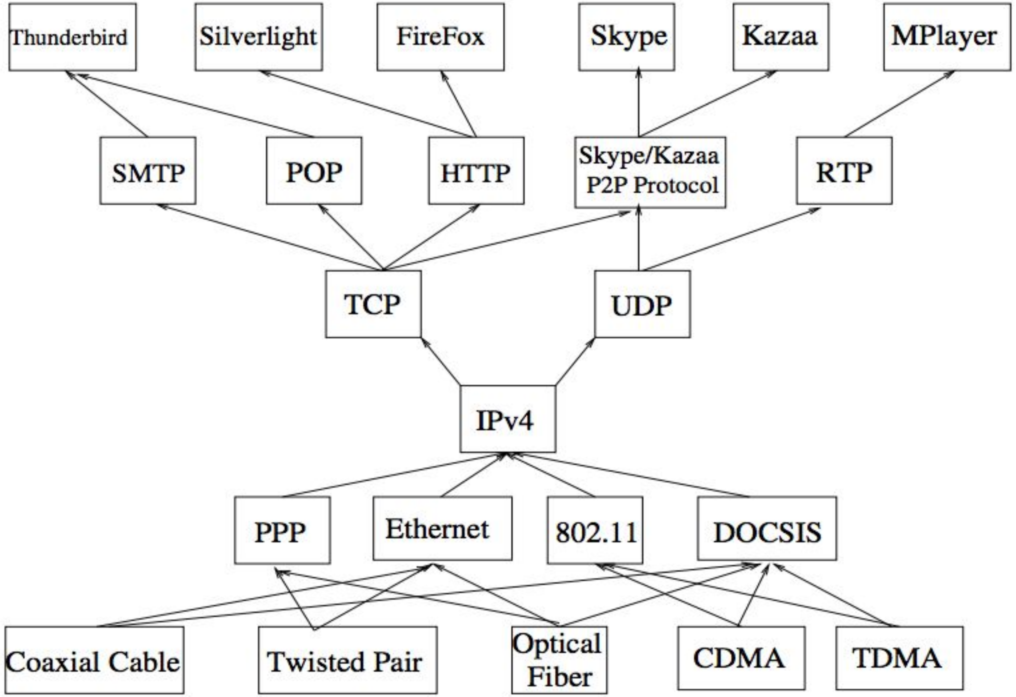
Destination Network

Internet Firewall Settings

Port number

○ Send delay

The hourglass Internet architecture



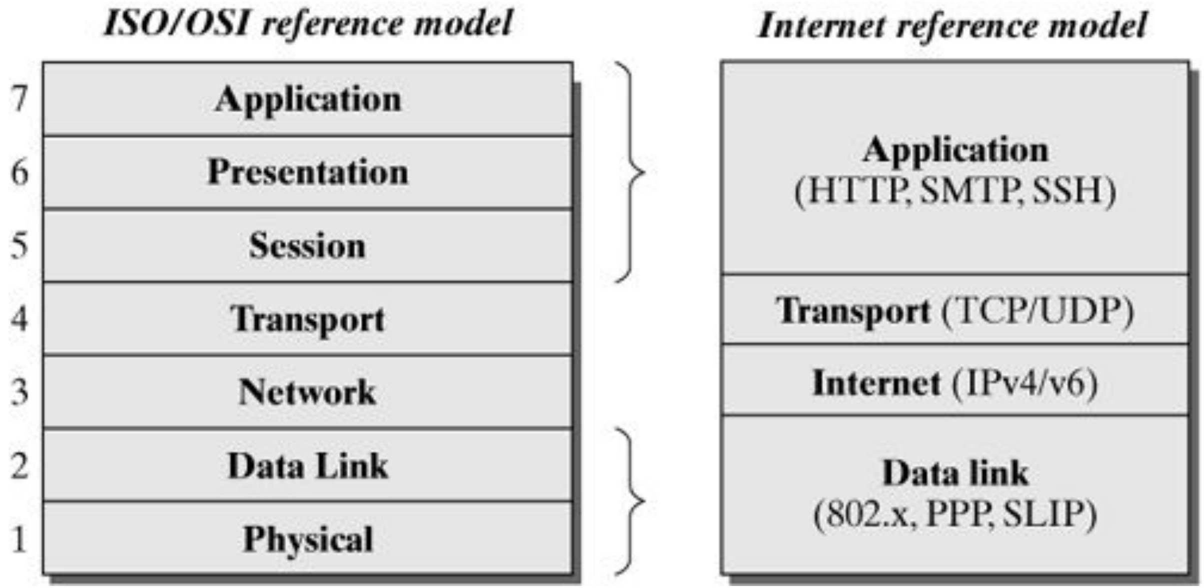
“The Evolution of Layered Protocol Stacks Leads to an Hourglass-Shaped Architecture”, ACM Sigcomm 2011

Internet Reference Model or TCP/IP Model

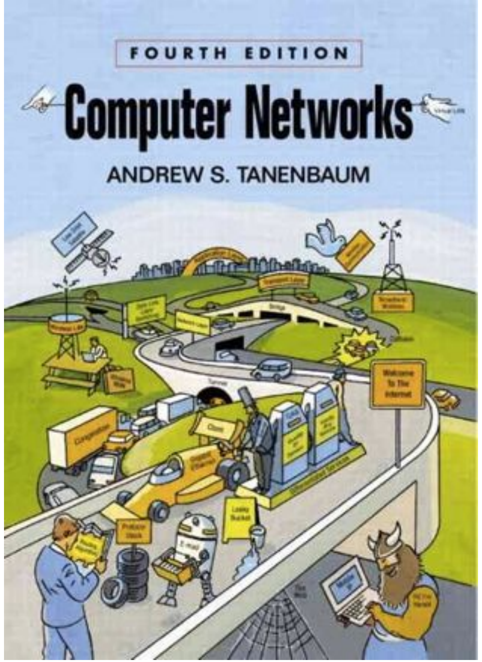
- TCP/IP does not have an official layer structure

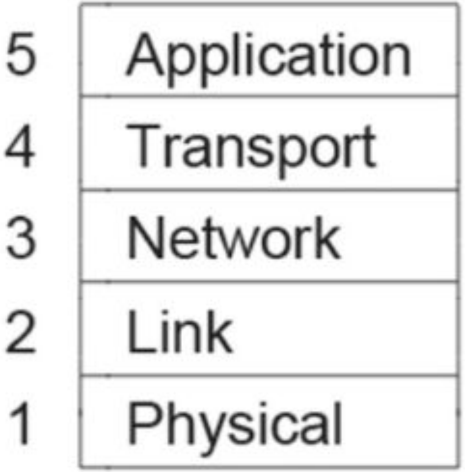
- But protocols imply one
  - Application layer
  - Transport (host to host) layer
  - Internet layer
  - Network access layer
  - Physical layer

Internet Reference Model or TCP/IP Model



Internet Reference Model or TCP/IP Model





5	Application
4	Transport
3	Network
2	Link
1	Physical

Physical and data link layers are tightly coupled.

Protocol Data Unit (PDU)

- A protocol data unit is information delivered as a unit among peer entities of networks containing control information, address information or data.

- In layered systems, PDU represents **a unit of data** specified **in the protocol of a given layer**, which consists of protocol control information and user data.

Protocol Data Unit (PDU)

- In Application layer, PDU is referred to as data
- In Transport layer, PDU is a segment (TCP segment)
- In Internet layer, PDU is a packet or diagram
- In Data link layer, PDU is a frame
- In Physical layer, PDU is just bit

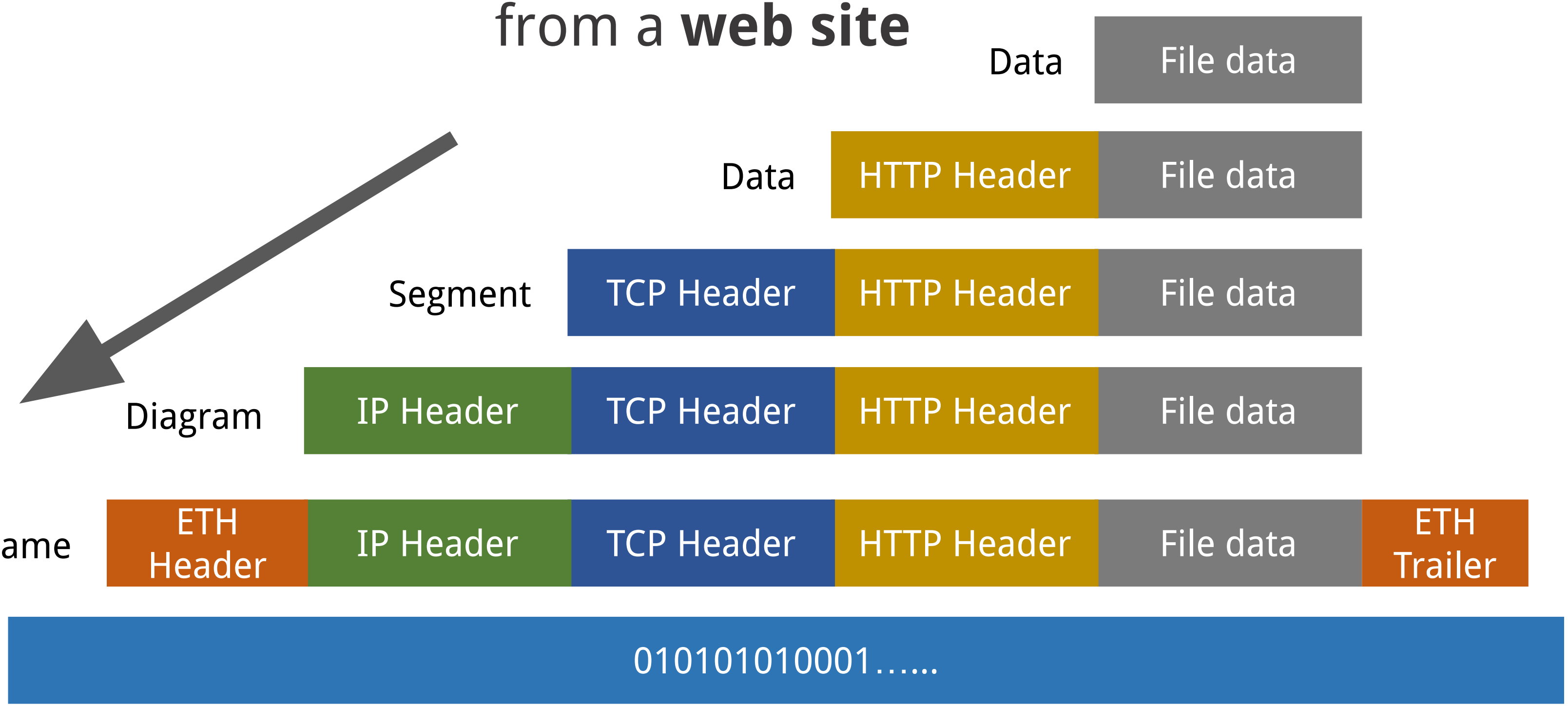
Protocol Data Unit (PDU)

- User data is passed from layer to layer. Outgoing data is packaged and identified for delivery to the layer underneath

- Control information is added/removed/changed to/from user data at each layer
  - Header
    - Identifies the protocol in use, the **sender** and intended **recipient**
  - Trailer
    - Provides **data integrity checks** for the payload
  - Each layer has a different header/trailer

PDUs in the TCP/IP Suite

Scenario: You use a browser on your laptop to download a file from a **web site**



Data | File data

Data | HTTP Header | File data

Segment | TCP Header | HTTP Header | File data

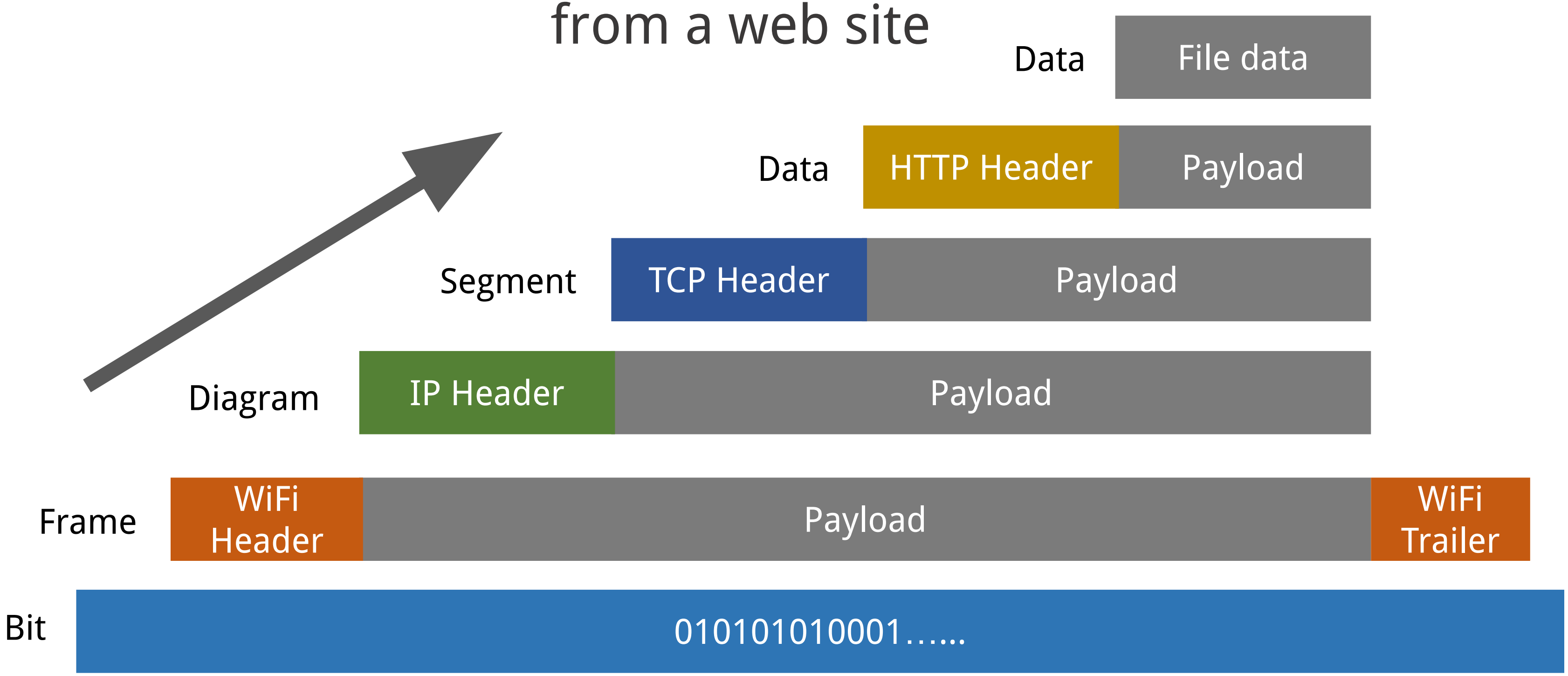
Diagram | IP Header | TCP Header | HTTP Header | File data

Frame | ETH Header | IP Header | TCP Header | HTTP Header | File data | ETH Trailer

Bit | 010101010001……

PDUs in the TCP/IP Suite

Scenario: You use a **browser** on your laptop to download a file from a web site



Data — File data

Data — HTTP Header | Payload

Segment — TCP Header | Payload

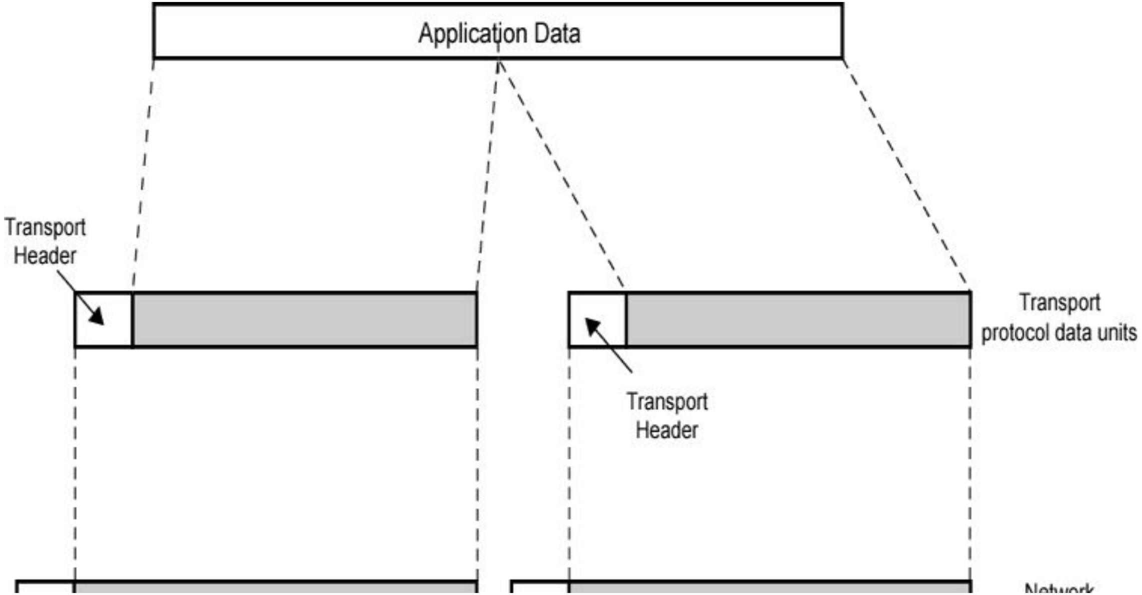
Diagram — IP Header | Payload

Frame — WiFi Header | Payload | WiFi Trailer

Bit — 010101010001……

Transport PDU

- Transport layer may **segment** upper layer data
- Each segment has a transport header added



Network PDU

- Network layer may **fragment** upper layer data

- IP packets can be up to 64KB, but Different link-layers have different Maximum transmission units (MTUs)
  - Ethernet : 1500 B

- Split IP packet into multiple fragments
  - IP header on each fragment

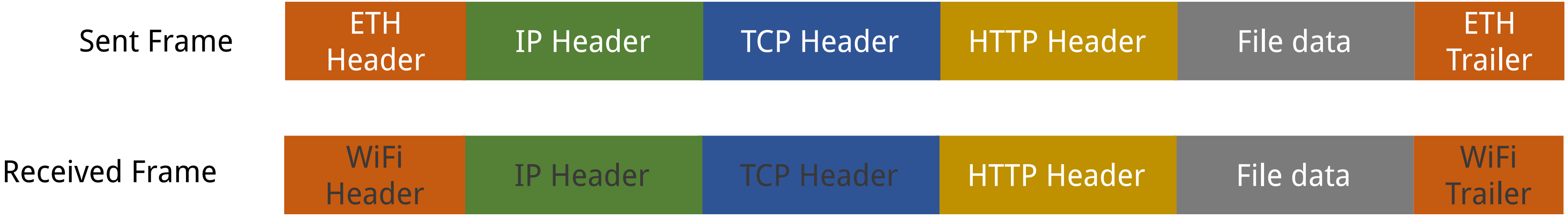
Data Link Layer/Physical Layer Overview

- Link layer addressing
- How hubs and switches work

Data Link Layer (Host-to-Network)

- This layer is the protocol layer that transfers data between adjacent network nodes in a wide area network (WAN) or between nodes on the same local area network (LAN).

- Data-link **frames do not cross** the boundaries of a local network. Data-link protocols to focus on local delivery, addressing



Sent Frame	ETH Header	IP Header	TCP Header	HTTP Header	File data	ETH Trailer
Received Frame	WiFi Header	IP Header	TCP Header	HTTP Header	File data	WiFi Trailer

Data Link Layer (Host-to-Network)

- Frame transferred by different link protocols over different links,
  - e.g., Ethernet on first link,
  - frame relay on intermediate links
  - 802.11 on last link

- Transportation analogy trip from Buffalo to San Francisco
  - Taxi: home to BUF (Local Transportation)
  - Plane: BUF to SFO airport (Intercity Transportation)
  - Shuttle: airport to hotel (Another Local Transportation)

Data Link Layer (Host-to-Network)

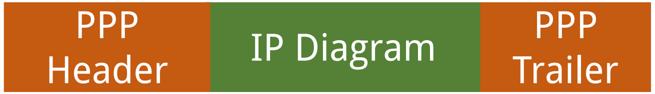
- TCP/IP reference model does not discuss these layers too much
  - the node should connect to the network with a protocol such that it can send IP packets
  - this protocol is not defined by TCP/IP
  - mostly in hardware

Data Link Layer Protocols

PPP: Point-to-Point Protocol

- Dial-up Internet Access: establish a connection to an Internet service provider (ISP) by dialing a telephone number on a conventional telephone line. The user's computer or router uses an attached modem to encode and decode information into and from audio frequency signals, respectively.

PPP Frame

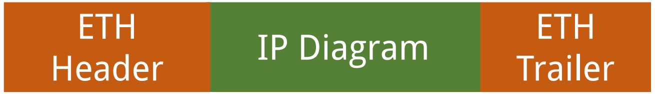


Data Link Layer Protocols

Ethernet: IEEE 802.3

- a family of computer networking technologies commonly used in local area networks (LANs)

ETH Frame



ETH Header

IP Diagram

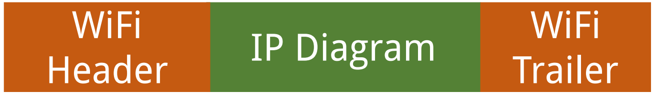
ETH Trailer

Data Link Layer Protocols

Wi-Fi: IEEE 802.11

- a technology that allows electronic devices to connect to a wireless LAN (WLAN) network

WiFi Frame



WiFi Header

IP Diagram

WiFi Trailer

MAC Address

- Media Access Control **address**

- The MAC address is a unique value associated with a **network adapter** (**network interface controller**, **NIC**).

- Also known as hardware addresses or physical addresses.

- MAC addresses are 12-digit hexadecimal numbers (48 bits in length). Usually written in:
  - MM : MM : MM : SS : SS : SS

MAC Address

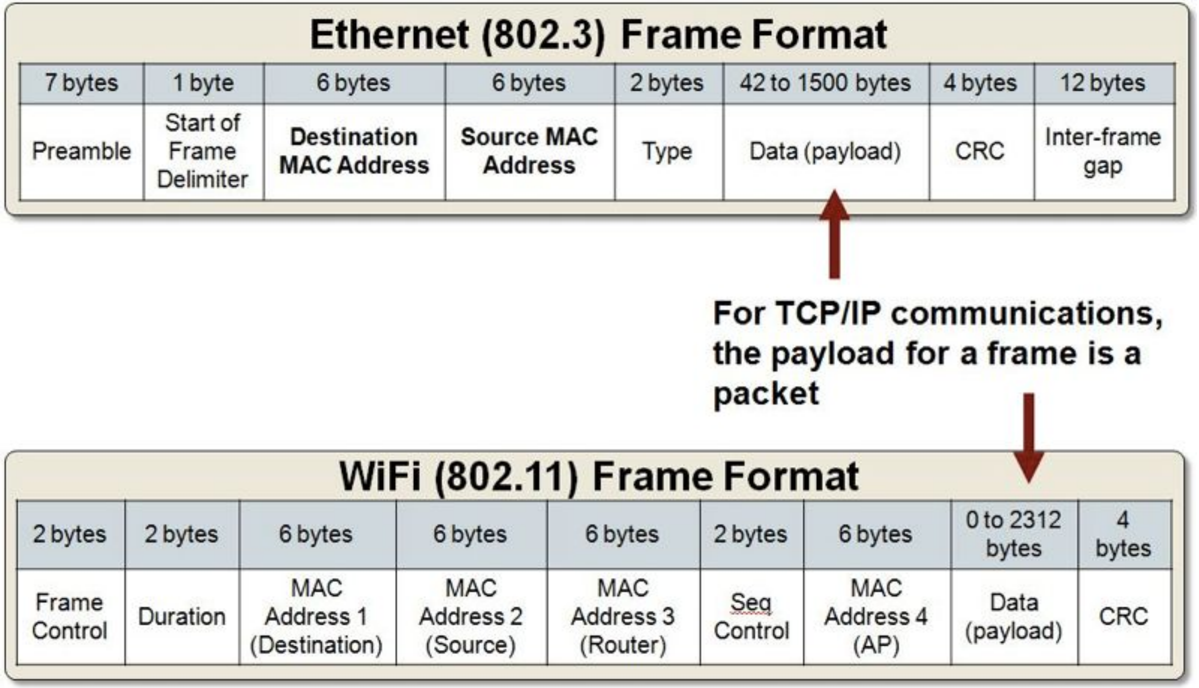
- Theoretically, every single NIC in the world should have a totally unique MAC address.

- The first half of a MAC address contains the ID number of the adapter manufacturer, which is called OUI (Organizationally Unique Identifier).

- The second half of a MAC address is the Device ID that represents the serial number assigned to the adapter by the manufacturer.

Ethernet and Wi-Fi Frame Header

- Ethernet and Wi-Fi adapters have MAC addresses



Ethernet (802.3) Frame Format

7 bytes	1 byte	6 bytes	6 bytes	2 bytes	42 to 1500 bytes	4 bytes	12 bytes
Preamble	Start of Frame Delimiter	**Destination MAC Address**	**Source MAC Address**	Type	Data (payload)	CRC	Inter-frame gap

For TCP/IP communications, the payload for a frame is a packet

WiFi (802.11) Frame Format

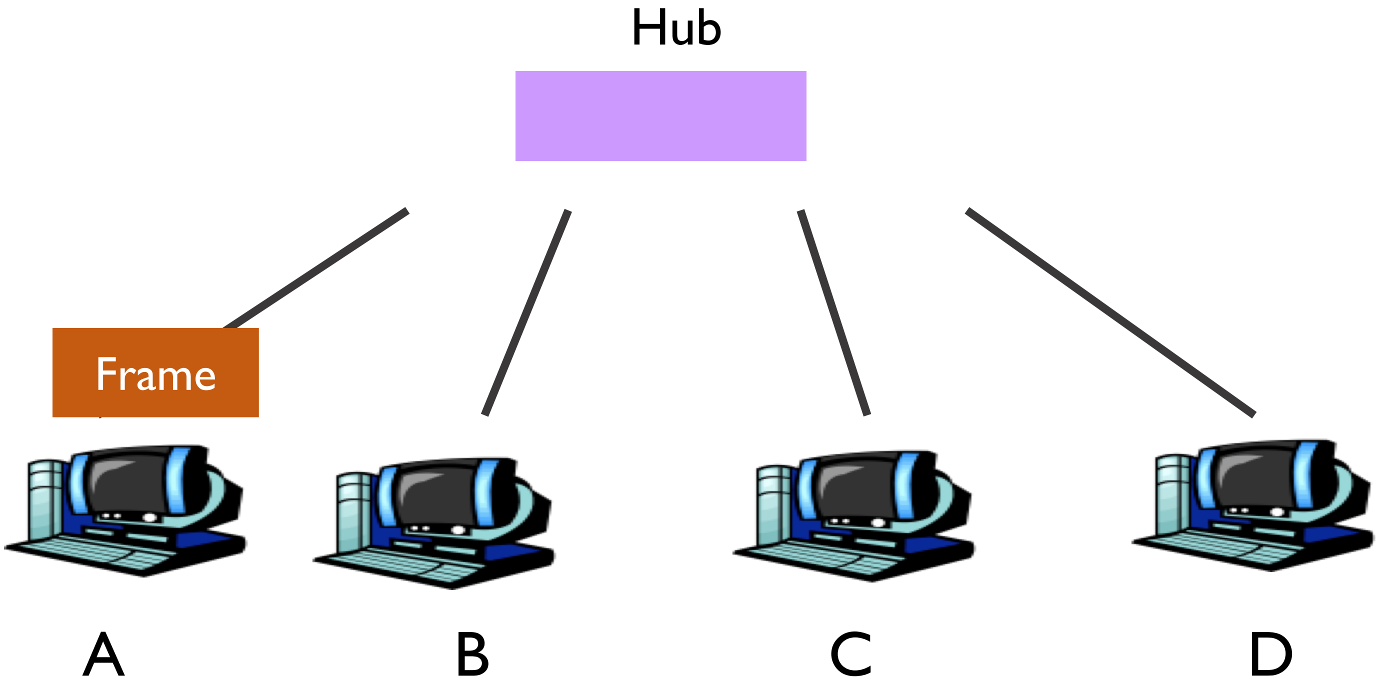
2 bytes	2 bytes	6 bytes	6 bytes	6 bytes	2 bytes	6 bytes	0 to 2312 bytes	4 bytes
Frame Control	Duration	MAC Address 1 (Destination)	MAC Address 2 (Source)	MAC Address 3 (Router)	Seq Control	MAC Address 4 (AP)	Data (payload)	CRC

Ethernet Hub

Ethernet hub is a network hardware device for connecting multiple Ethernet devices together and making them act as a single network segment.

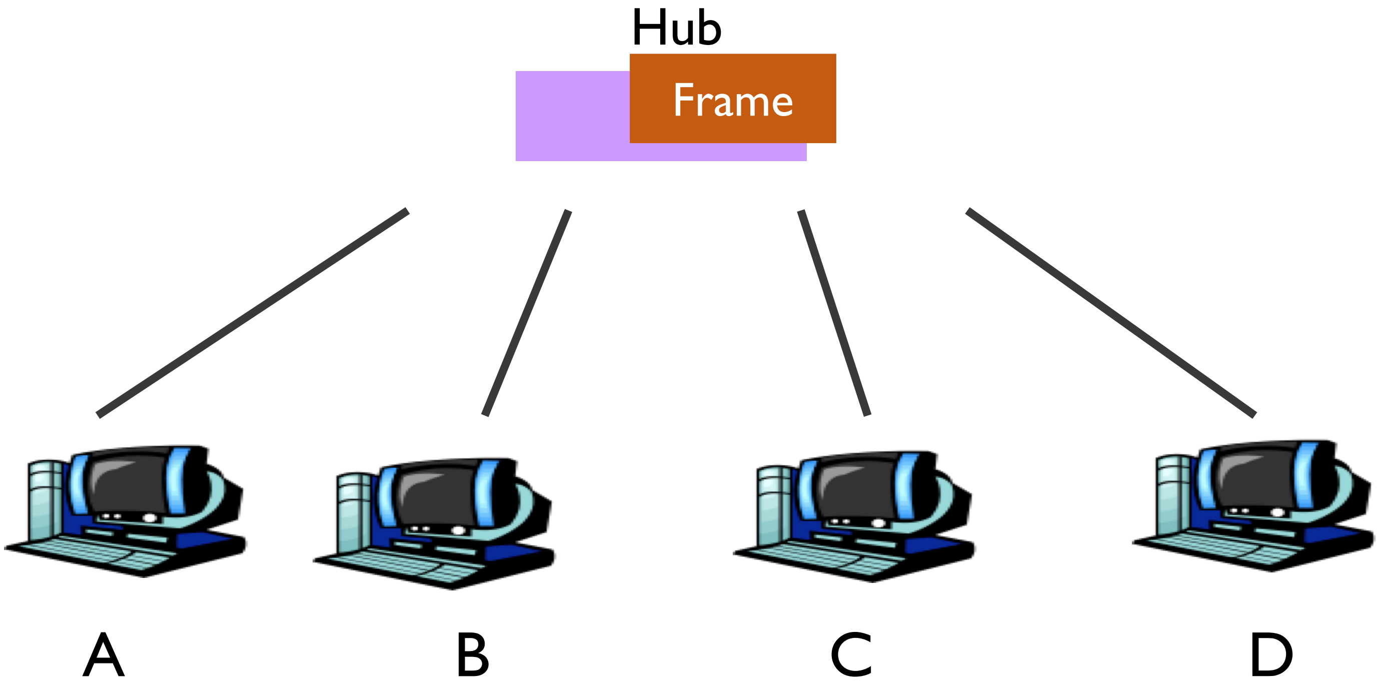
How a Hub Works

- It regenerates frames and broadcasts them to all ports



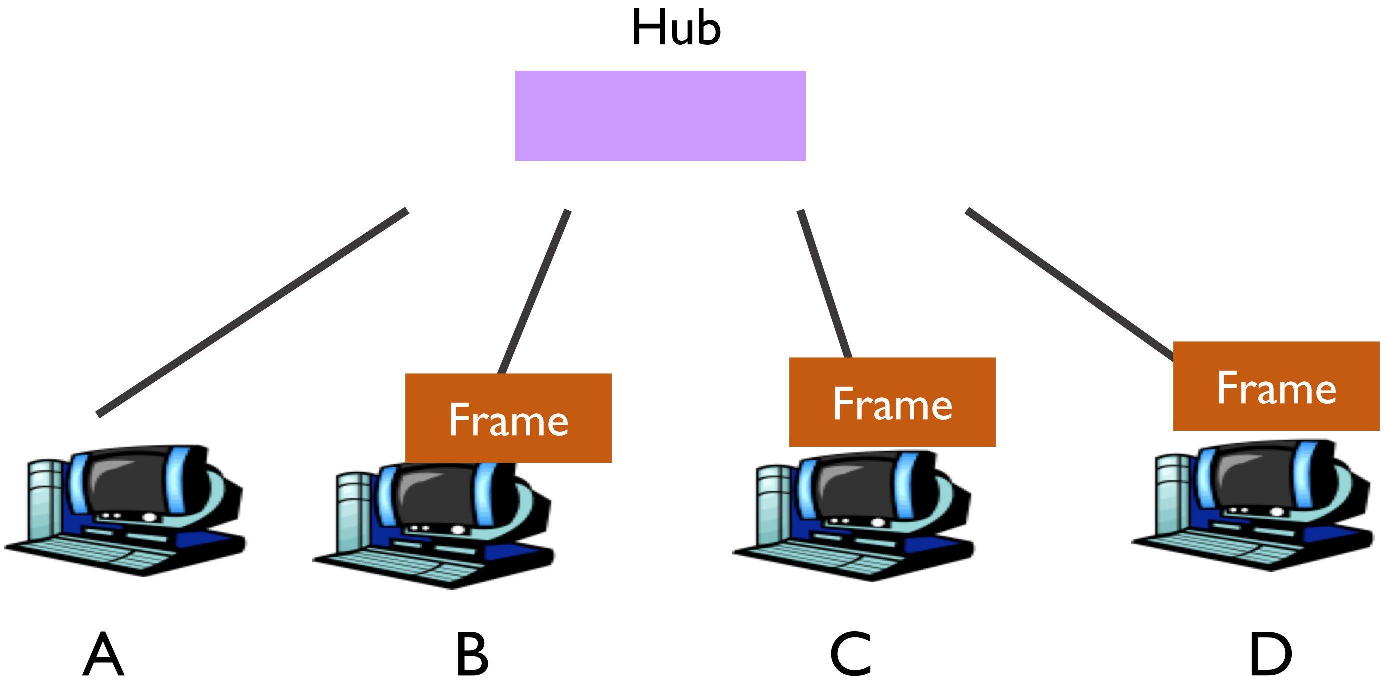
How a Hub Works

- It regenerates frames and broadcasts them to all ports



How a Hub Works

- It regenerates frames and broadcasts them to all ports



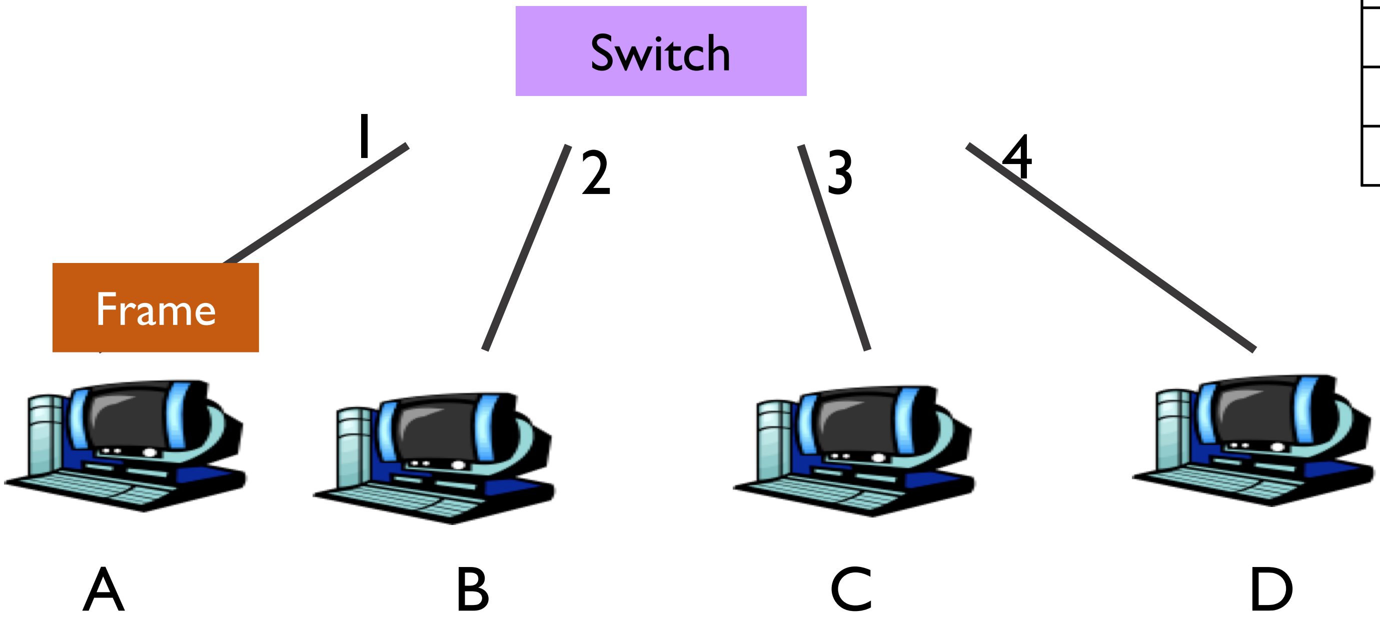
Switch

- Switches forward frames selectively
  - Forward frames only on segments that need them

- Switch table
  - Maps destination MAC address to outgoing interface
  - Construct the switch table automatically

How a Switch Works

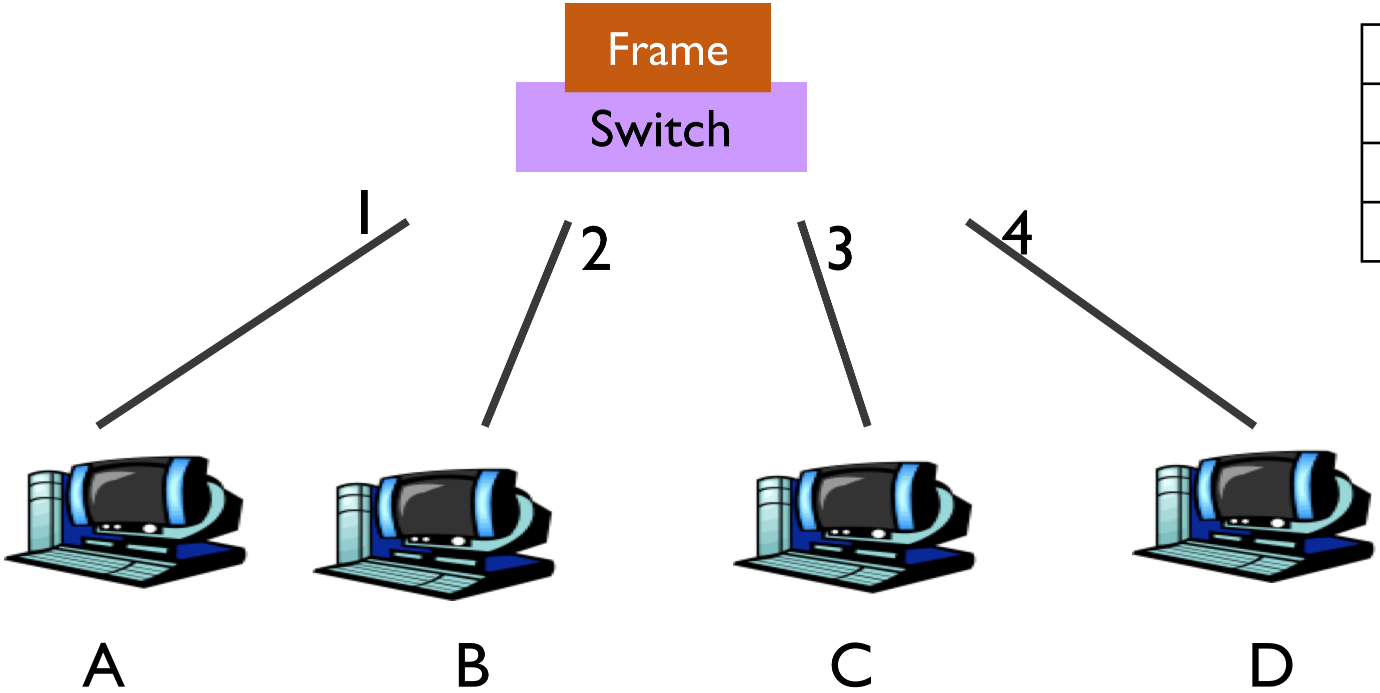
Which layer does a switch work on?



1	00:00:00:aa:aa:aa
2	00:00:00:bb:bb:bb
3	00:00:00:cc:cc:cc
4	00:00:00:dd:dd:dd

How a Switch Works

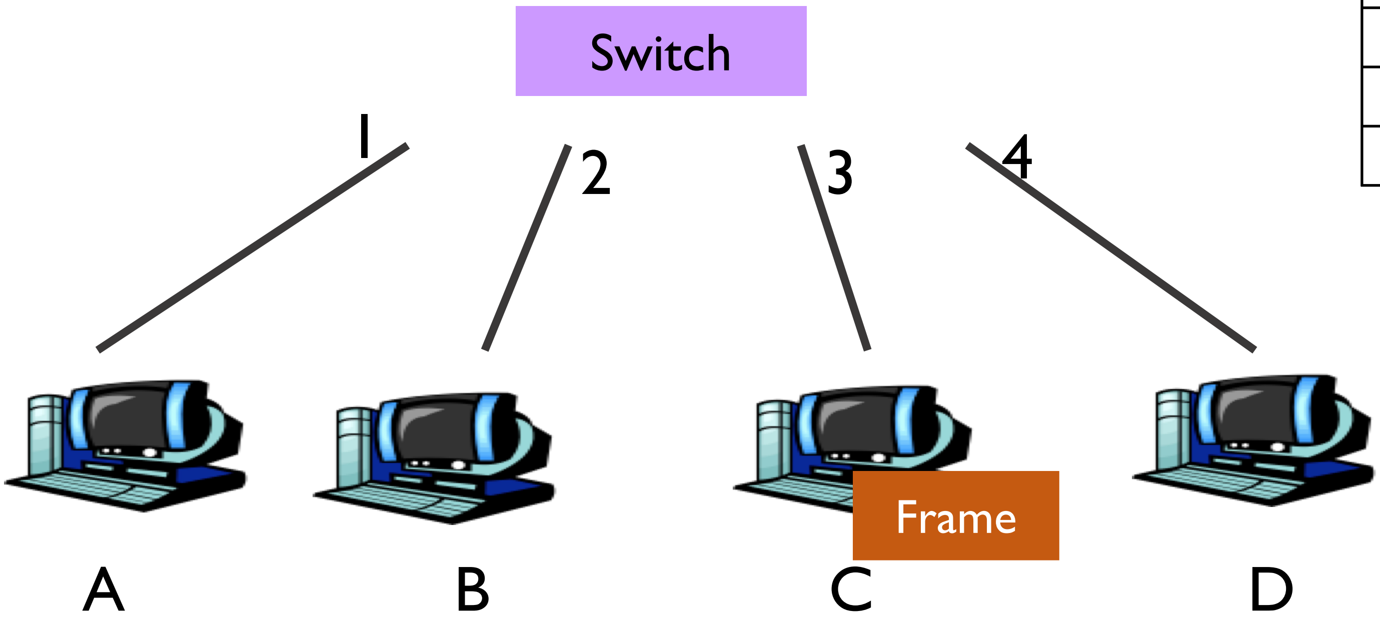
Which layer does a switch work on?



1	00:00:00:aa:aa:aa
2	00:00:00:bb:bb:bb
3	00:00:00:cc:cc:cc
4	00:00:00:dd:dd:dd

How a Switch Works

Which layer does a switch work on?

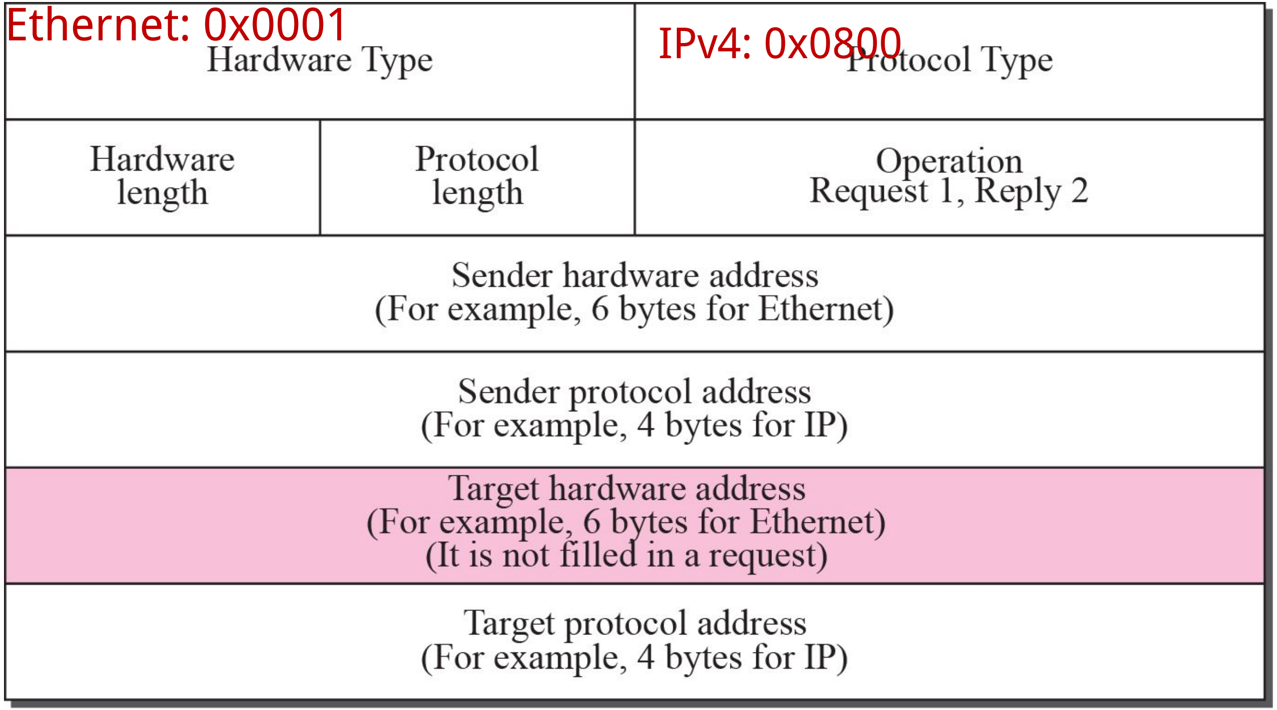


1	00:00:00:aa:aa:aa
2	00:00:00:bb:bb:bb
3	00:00:00:cc:cc:cc
4	00:00:00:dd:dd:dd

Address Resolution Protocol (ARP)

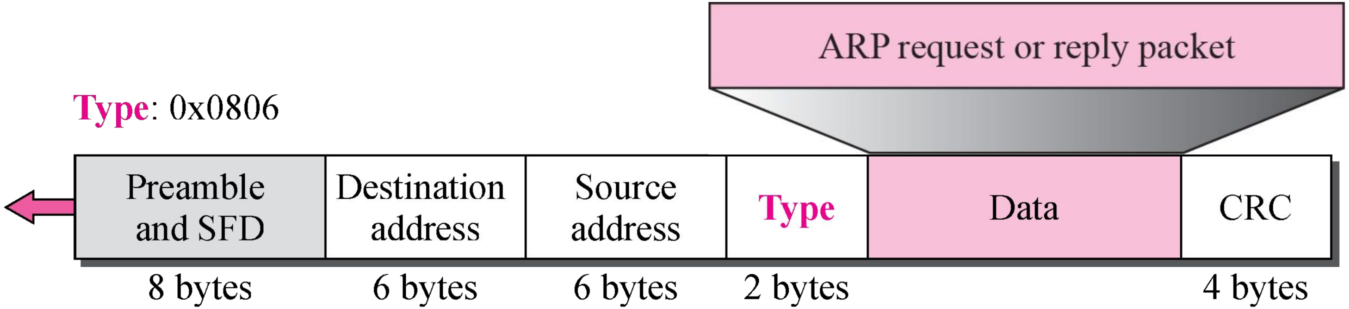
- Translate network-layer address to data-link layer address
  - Primarily used to translate IP addresses to Ethernet MAC addresses in our daily life

ARP Packet Format



<table>
<tr><td colspan="2">Ethernet: 0x0001
Hardware Type</td><td colspan="2">IPv4: 0x0800
Protocol Type</td></tr>
<tr><td>Hardware length</td><td>Protocol length</td><td colspan="2">Operation
Request 1, Reply 2</td></tr>
<tr><td colspan="4">Sender hardware address
(For example, 6 bytes for Ethernet)</td></tr>
<tr><td colspan="4">Sender protocol address
(For example, 4 bytes for IP)</td></tr>
<tr><td colspan="4">Target hardware address
(For example, 6 bytes for Ethernet)
(It is not filled in a request)</td></tr>
<tr><td colspan="4">Target protocol address
(For example, 4 bytes for IP)</td></tr>
</table>

ARP Packet Encapsulated in Ethernet Frame



Scenario: A wants to talk with B. A knows B's IP addr but not MAC addr.

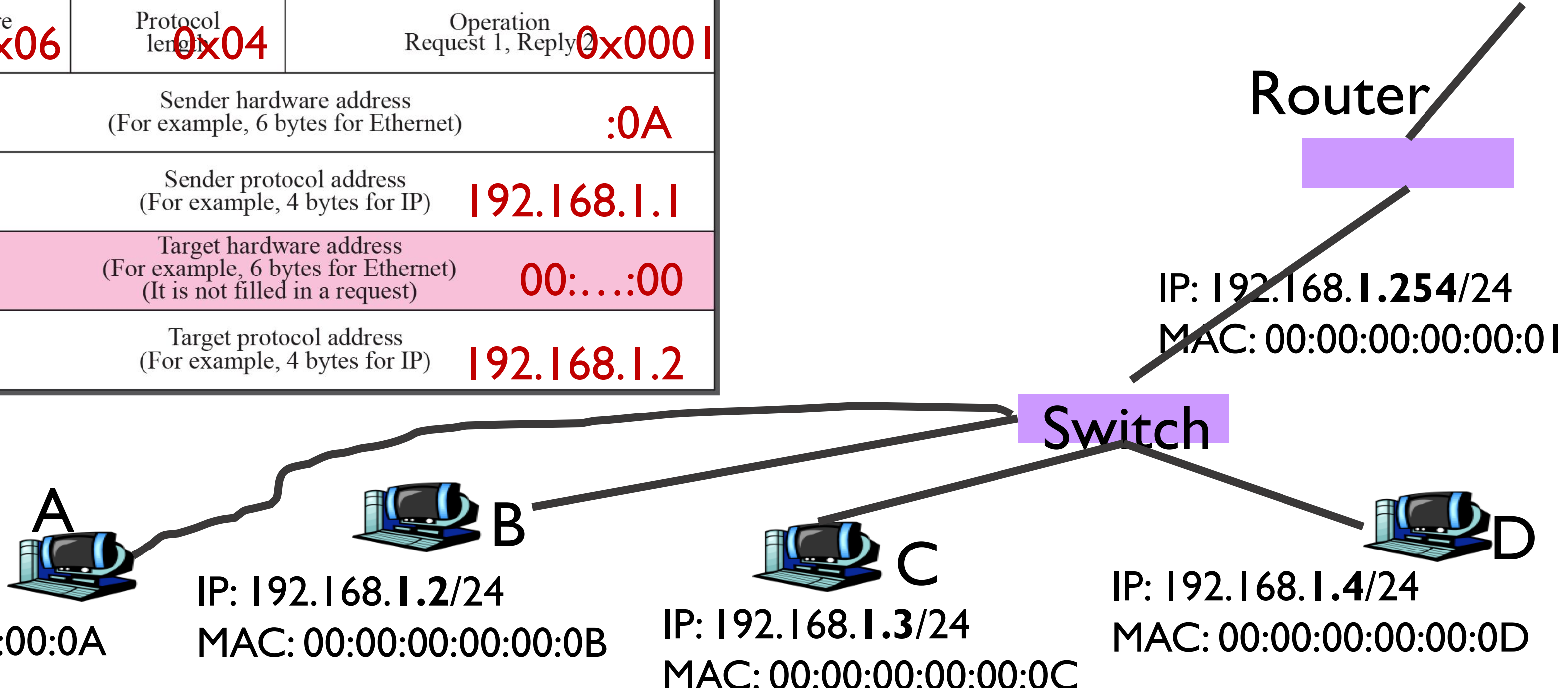
Host A

Step 1: Host A 192.168.1.1 calculates if 192.168.1.2 is in my network – Yes

Step 2: Check if it already knows B's MAC addr - No

Step 3: Host A constructs an ARP Request. What is the MAC for 192.168.1.2?

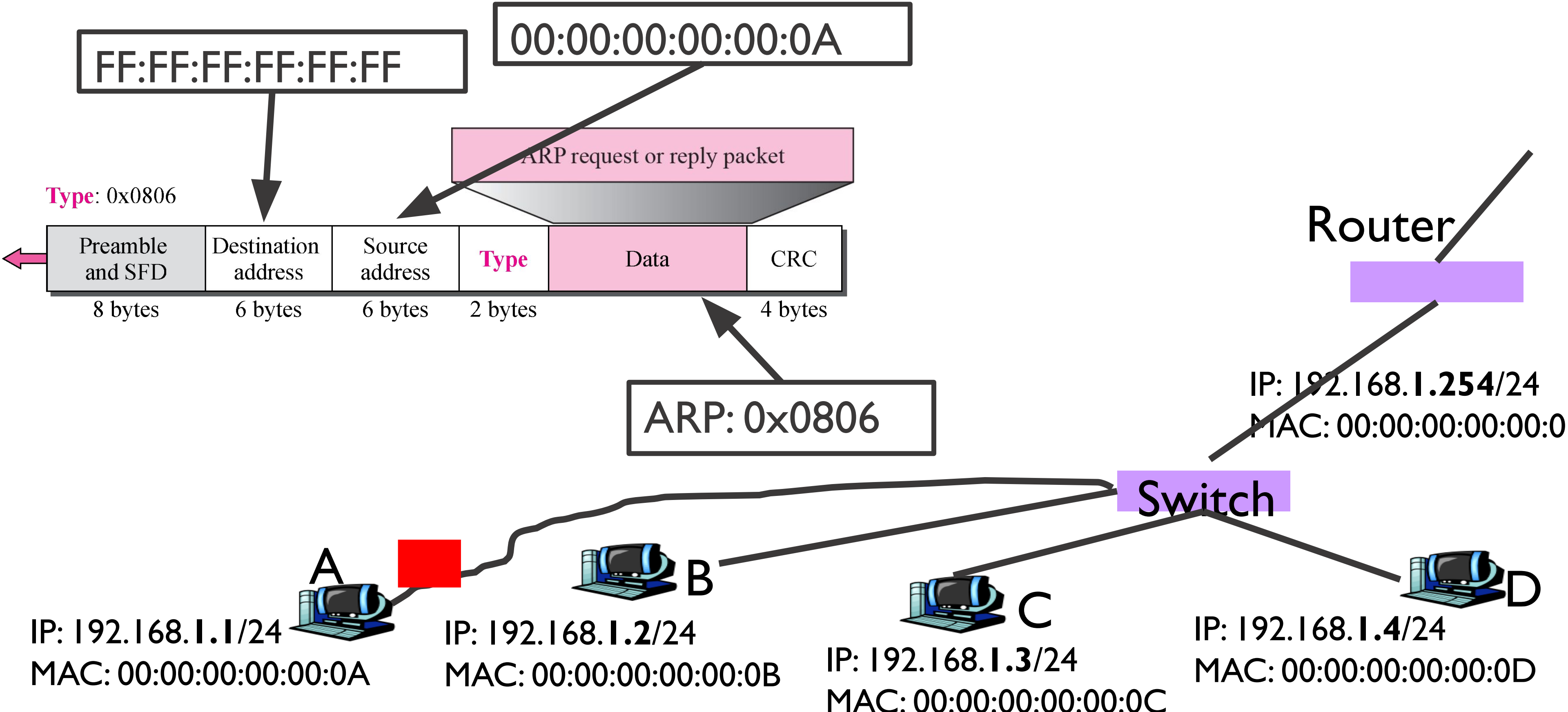
Hardware Type 0x0001		Protocol Type 0x0800
Hardware length 0x06	Protocol length 0x04	Operation Request 1, Reply 2 0x0001
Sender hardware address (For example, 6 bytes for Ethernet) :0A		
Sender protocol address (For example, 4 bytes for IP) 192.168.1.1		
Target hardware address (For example, 6 bytes for Ethernet) (It is not filled in a request) 00:…:00		
Target protocol address (For example, 4 bytes for IP) 192.168.1.2		



Scenario: A wants to talk with B. A knows B's IP addr but not MAC addr.

Host A

Step 4: Host A encapsulates the ARP Request in an Ethernet frame



FF:FF:FF:FF:FF:FF

00:00:00:00:00:0A

ARP: 0x0806

IP: 192.168.1.254/24
MAC: 00:00:00:00:00:0

A
IP: 192.168.**1**.**1**/24
MAC: 00:00:00:00:00:0A

B
IP: 192.168.**1**.**2**/24
MAC: 00:00:00:00:00:0B

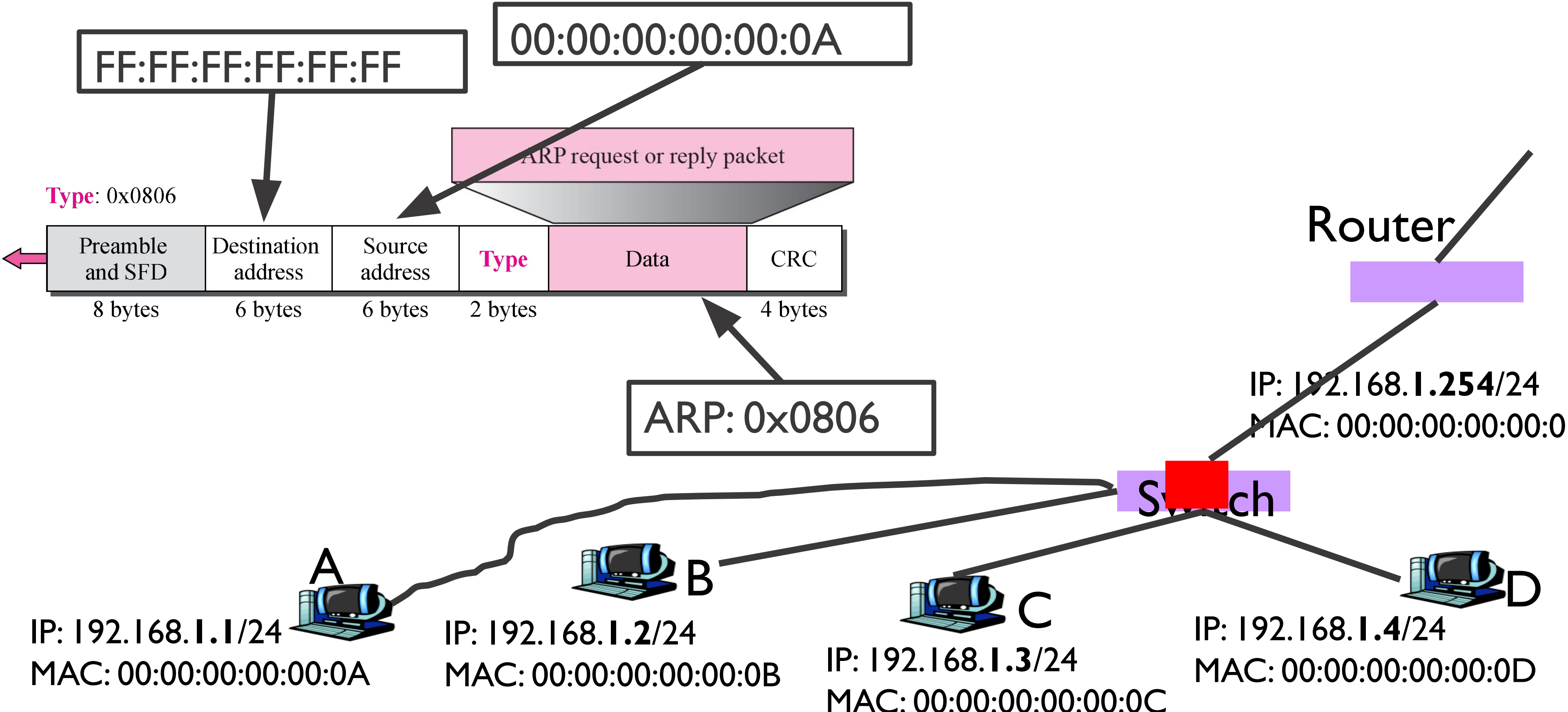
C
IP: 192.168.**1**.**3**/24
MAC: 00:00:00:00:00:0C

D
IP: 192.168.**1**.**4**/24
MAC: 00:00:00:00:00:0D

Scenario: A wants to talk with B. A knows B’s IP addr but not MAC addr.

Host A

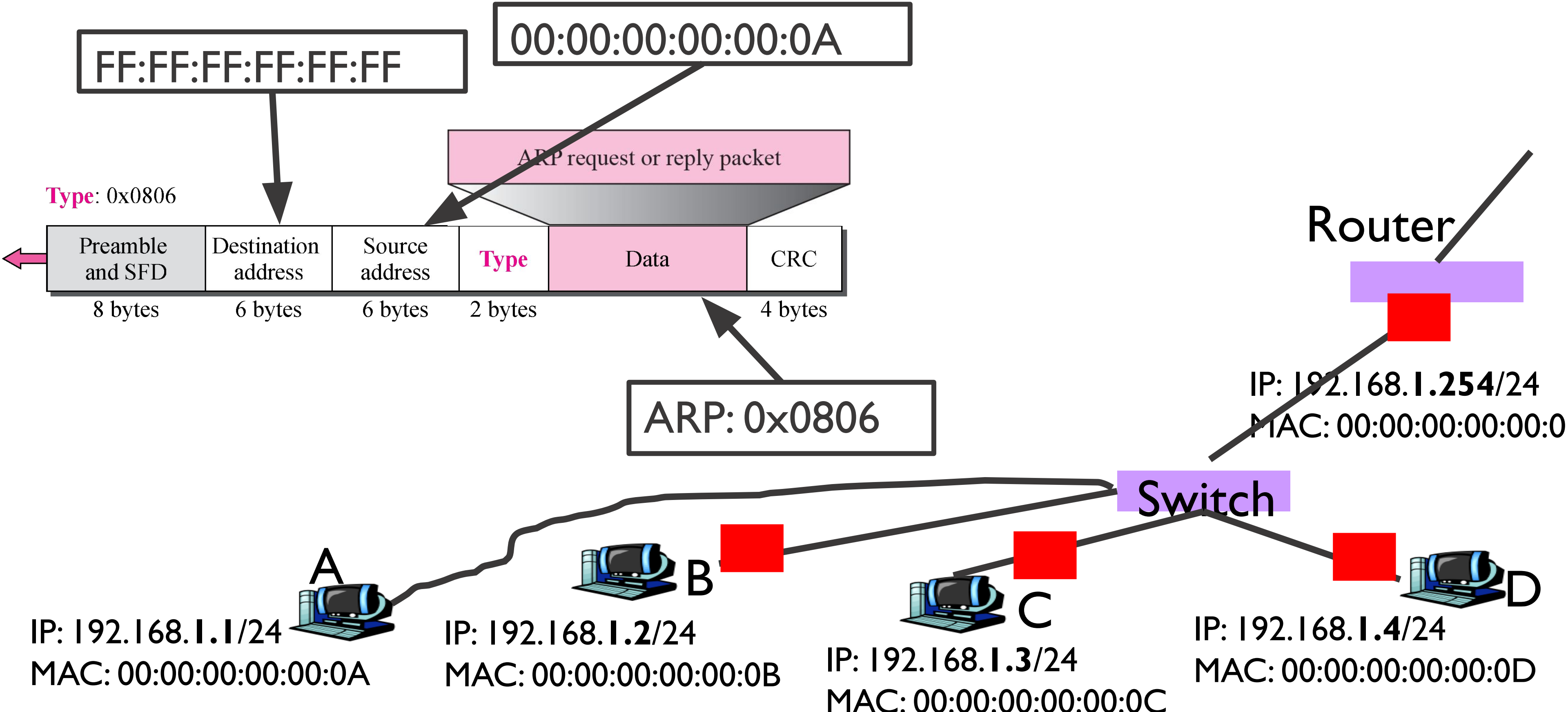
Step 4: Host A encapsulates the ARP Request in an Ethernet frame



Preamble and SFD	Destination address	Source address	Type	Data	CRC
8 bytes	6 bytes	6 bytes	2 bytes		4 bytes

Scenario: A wants to talk with B. A knows B's IP addr but not MAC addr.

Step 5: Router, Host B, C, D parse the frame. And, give the packet to upper layer handler



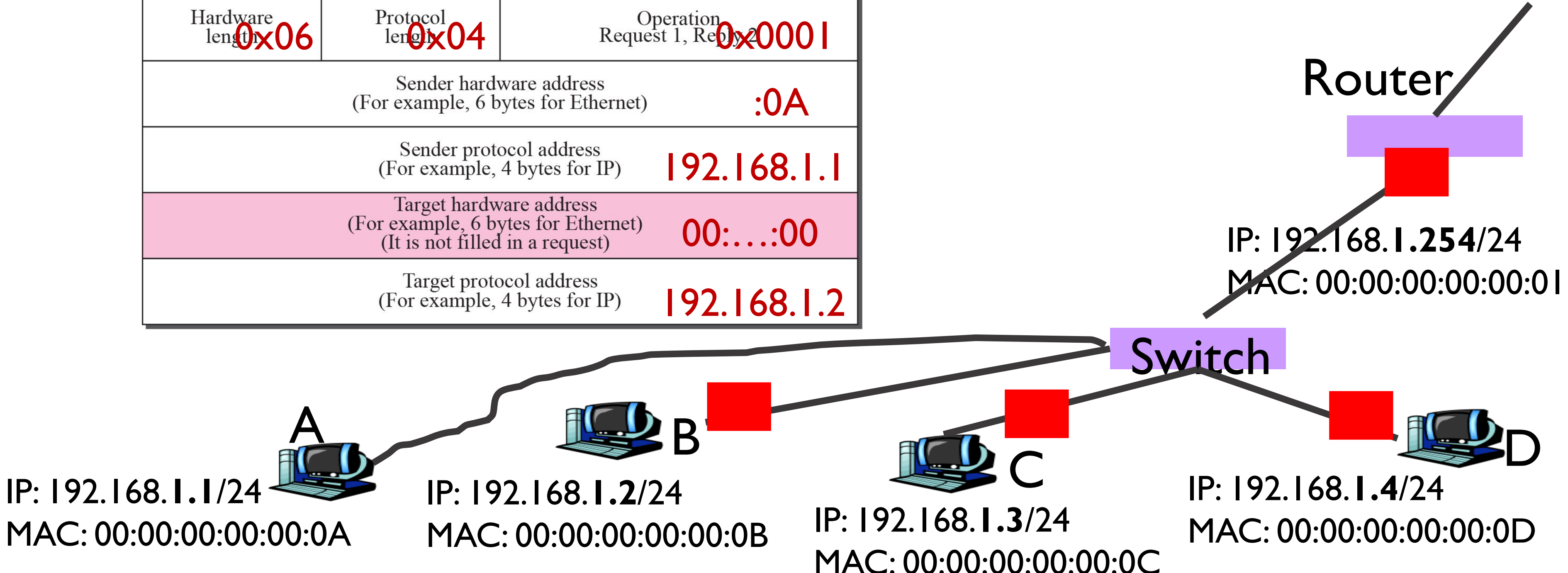
Scenario: A wants to talk with B. A knows B’s IP addr but not MAC addr.

Router, Host B, C, D

Step 4: Compare the IP with its own IP.

Step 5: Honest Router, Host C and D will ignore this package

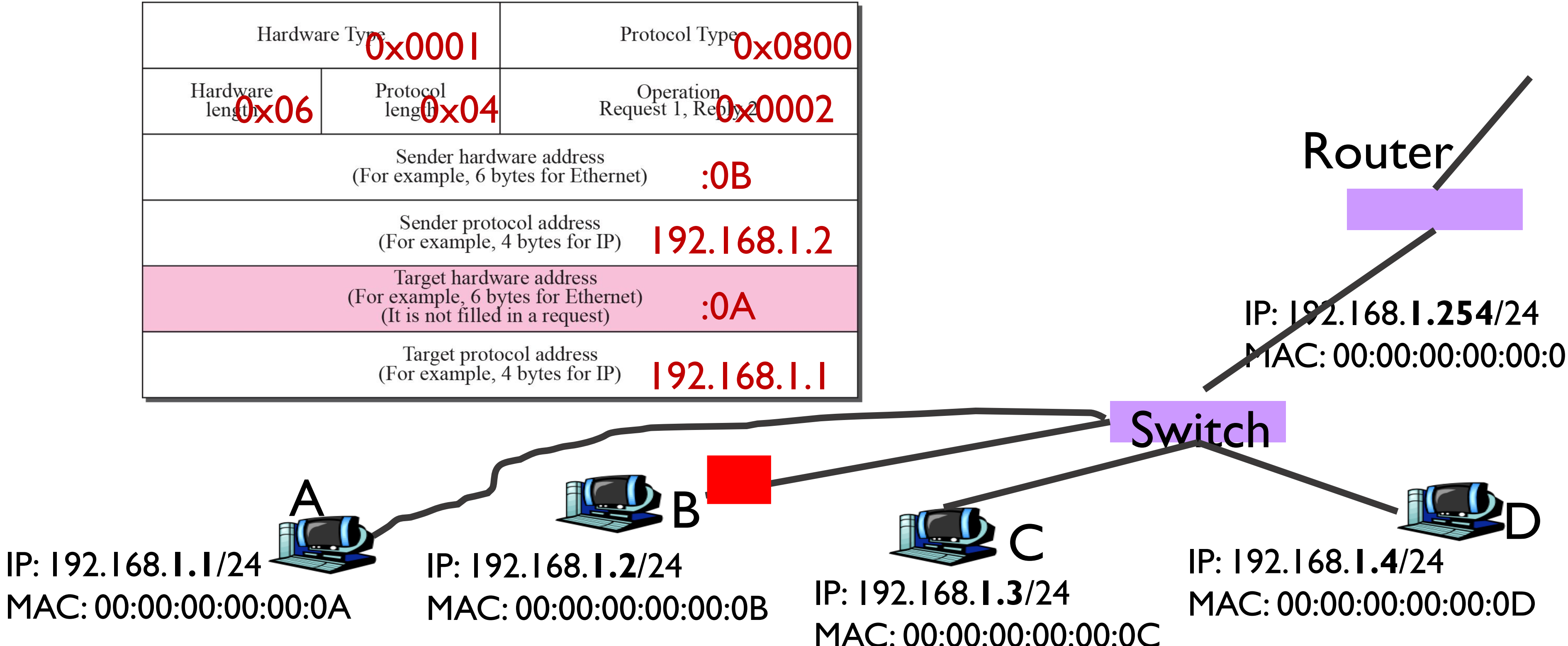
Hardware length	Protocol length	Operation Request 1, Reply 2
Hardware Type 0x0001		Protocol Type 0x0800
0x06	0x04	0x0001
Sender hardware address (For example, 6 bytes for Ethernet)		:0A
Sender protocol address (For example, 4 bytes for IP)		192.168.1.1
Target hardware address (For example, 6 bytes for Ethernet) (It is not filled in a request)		00:…:00
Target protocol address (For example, 4 bytes for IP)		192.168.1.2



Scenario: A wants to talk with B. A knows B's IP.

Host B:

Step 1: Host B constructs an ARP Reply. My IP is192.168.1.2 and my MAC is :0B

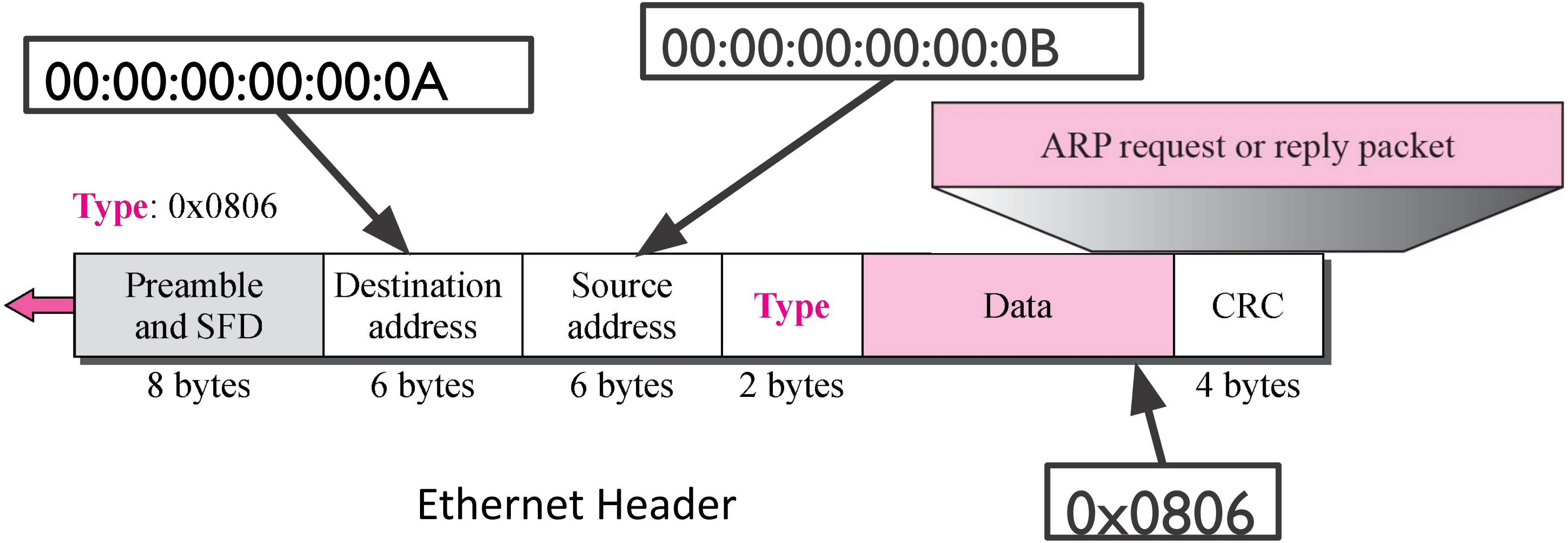


Hardware Type 0x0001		Protocol Type 0x0800
Hardware length 0x06	Protocol length 0x04	Operation Request 1, Reply 2 0x0002
Sender hardware address (For example, 6 bytes for Ethernet) :0B		
Sender protocol address (For example, 4 bytes for IP) 192.168.1.2		
Target hardware address (For example, 6 bytes for Ethernet) (It is not filled in a request) :0A		
Target protocol address (For example, 4 bytes for IP) 192.168.1.1		

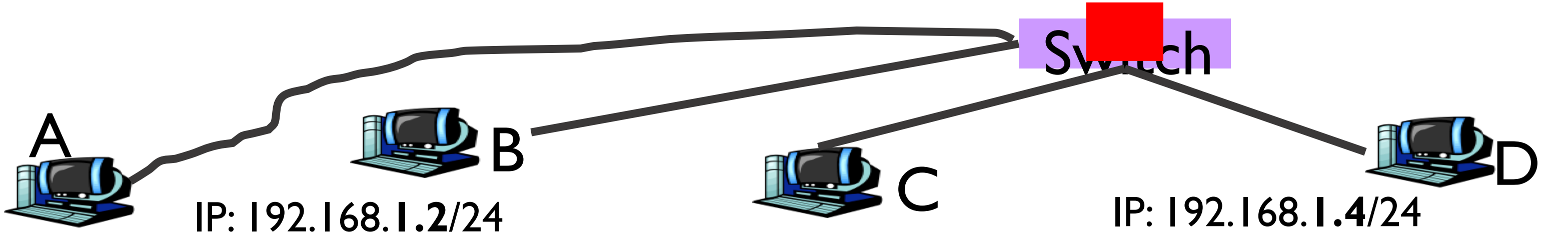
Scenario: A wants to talk with B. A knows B’s IP.

Host B

Step 2: Host B encapsulates the ARP Reply in an Ethernet frame



Ethernet Header



IP: 192.168.**1.1**/24
MAC: 00:00:00:00:00:0A

IP: 192.168.**1.2**/24
MAC: 00:00:00:00:00:0B

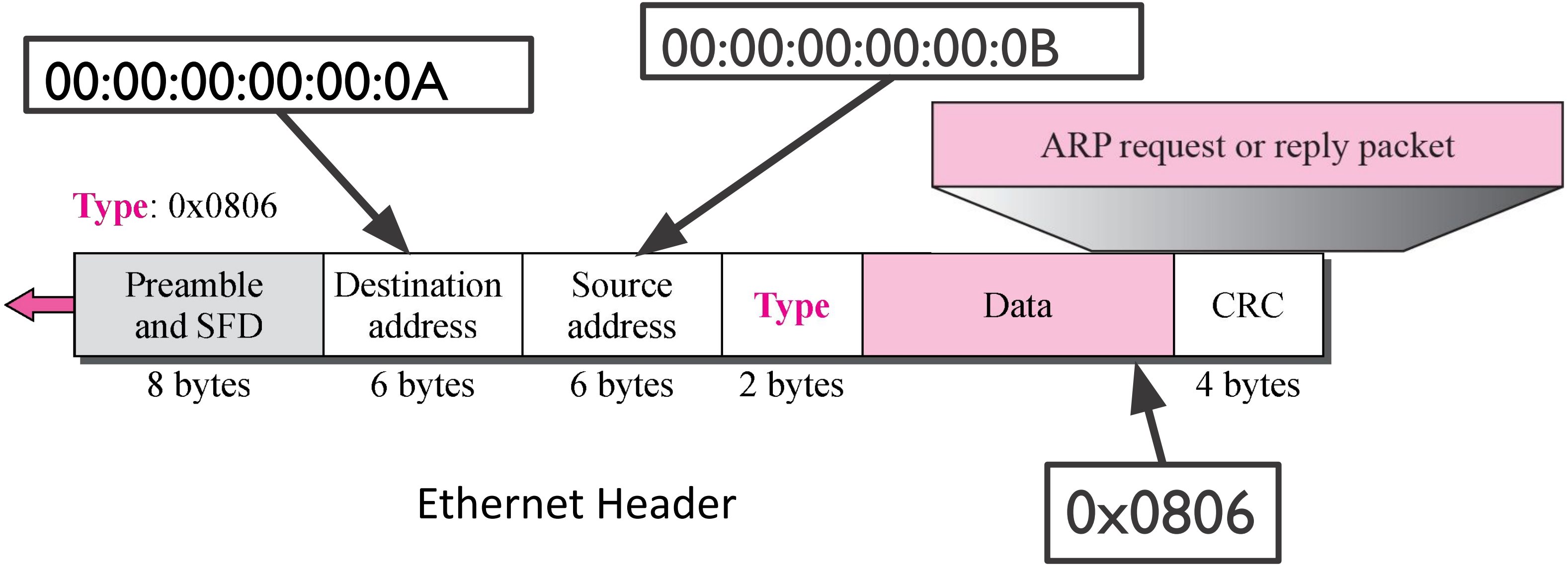
IP: 192.168.**1.3**/24
MAC: 00:00:00:00:00:0C

IP: 192.168.**1.4**/24
MAC: 00:00:00:00:00:0D

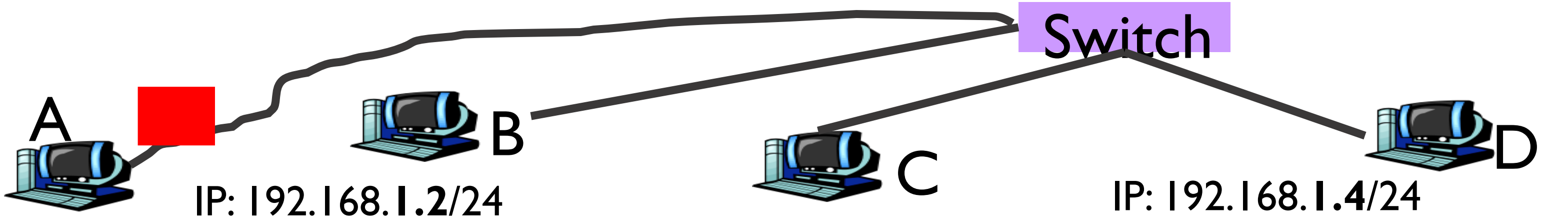
Scenario: A wants to talk with B. A knows B's IP.

Host B

Step 2: Host B encapsulates the ARP Reply in an Ethernet frame



Ethernet Header



IP: 192.168.**1**.**1**/24
MAC: 00:00:00:00:00:0A

IP: 192.168.**1**.**2**/24
MAC: 00:00:00:00:00:0B

IP: 192.168.**1**.**3**/24
MAC: 00:00:00:00:00:0C

IP: 192.168.**1**.**4**/24
MAC: 00:00:00:00:00:0D

ARP Cache

Since sending an ARP request/reply for each upper layer packet is inefficient, hosts maintain a cache (ARP Cache) of current entries.

- Example: the entries expire after 20 minutes.

```
Contents of the ARP Cache of Host A: (192.168.1.1):

at 00:00:00:00:00:0B on eth0 (192.168.1.2) Timeout: 20mins
at 00:00:00:00:00:0C on eth0 (192.168.1.3) Timeout: 15mins
at 00:00:00:00:00:0D on eth0 (192.168.1.4) Timeout: 15mins
```

ARP Spoofing

- Construct spoofed ARP Replies

- A target computer could be convinced to send frames destined for computer B to instead go to computer C

- Computer B will have no idea that this redirection took place

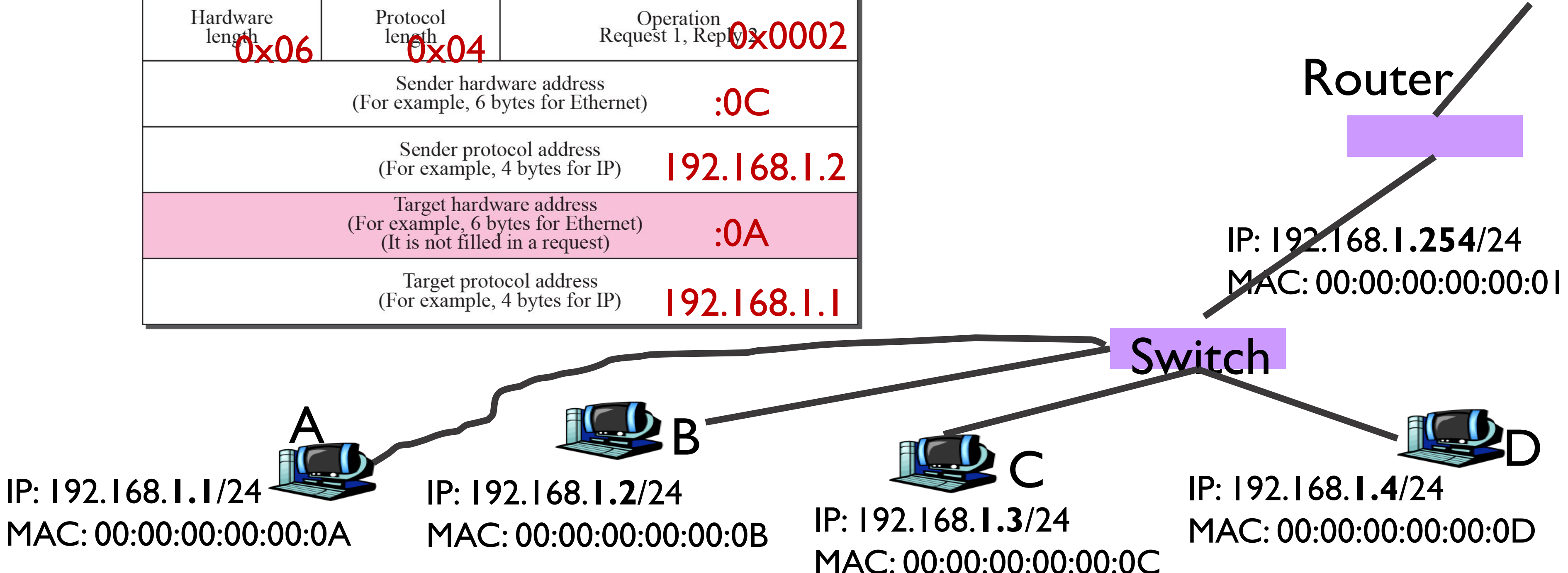
- This process of updating a target computer’s ARP cache is also referred to as “ARP poisoning”

Scenario: A wants to talk with B. A knows B’s IP addr but not MAC addr.

Host C is controlled by an attacker.

Host C constructs an ARP reply

Hardware Type	0x0001
Protocol Type	0x0800
Hardware length	0x06
Protocol length	0x04
Operation Request 1, Reply 2	0x0002
Sender hardware address (For example, 6 bytes for Ethernet)	:0C
Sender protocol address (For example, 4 bytes for IP)	192.168.1.2
Target hardware address (For example, 6 bytes for Ethernet) (It is not filled in a request)	:0A
Target protocol address (For example, 4 bytes for IP)	192.168.1.1



ARP Spoofing

Contents of the ARP Cache of Host A: (192.168.1.1):

at 00:00:00:00:00:0C on eth0 (192.168.1.2) Timeout: 20mins
at 00:00:00:00:00:0D on eth0 (192.168.1.4) Timeout: 15mins

Gratuitous ARP

- When the host's IP address or MAC address has changed, the host can use ARP as a simple announcement protocol

- ARP Replies can be broadcasted even if there is no ARP Request. All other hosts in the network may accept this reply
  - Host C can send the fake ARP even if Host A did not send an ARP Request. And, Host A may accept the fake ARP and update its cache.

ARP Spoofing

- Sniffing
  - By using ARP spoofing, all the traffic can be directed to the hackers. It is possible to perform sniffing on a switched network now.

- DoS
  - Updating ARP caches with non-existent MAC addresses will cause frames to be dropped. These could be sent out in a sweeping fashion to all clients on the network in order to cause a Denial of Service attack (DoS).

- Man-in-the-Middle
  - Host C Spoofs both Host A and B

Attack Model and Root Cause of ARP Spoofing

Attack Model:

- Attacker should reside in the same local network as the victims

Root Cause:

- There is no method in the ARP protocol by which a host can authenticate the peer from which the packet originated

Defenses against ARP Spoofing

Use static ARP entries

- Cannot be updated Spoofed
- ARP replies are ignored
- ARP table needs a static entry for each machine on the network

- Large overhead
  - Deploying these tables
  - Keep the table up-to-date

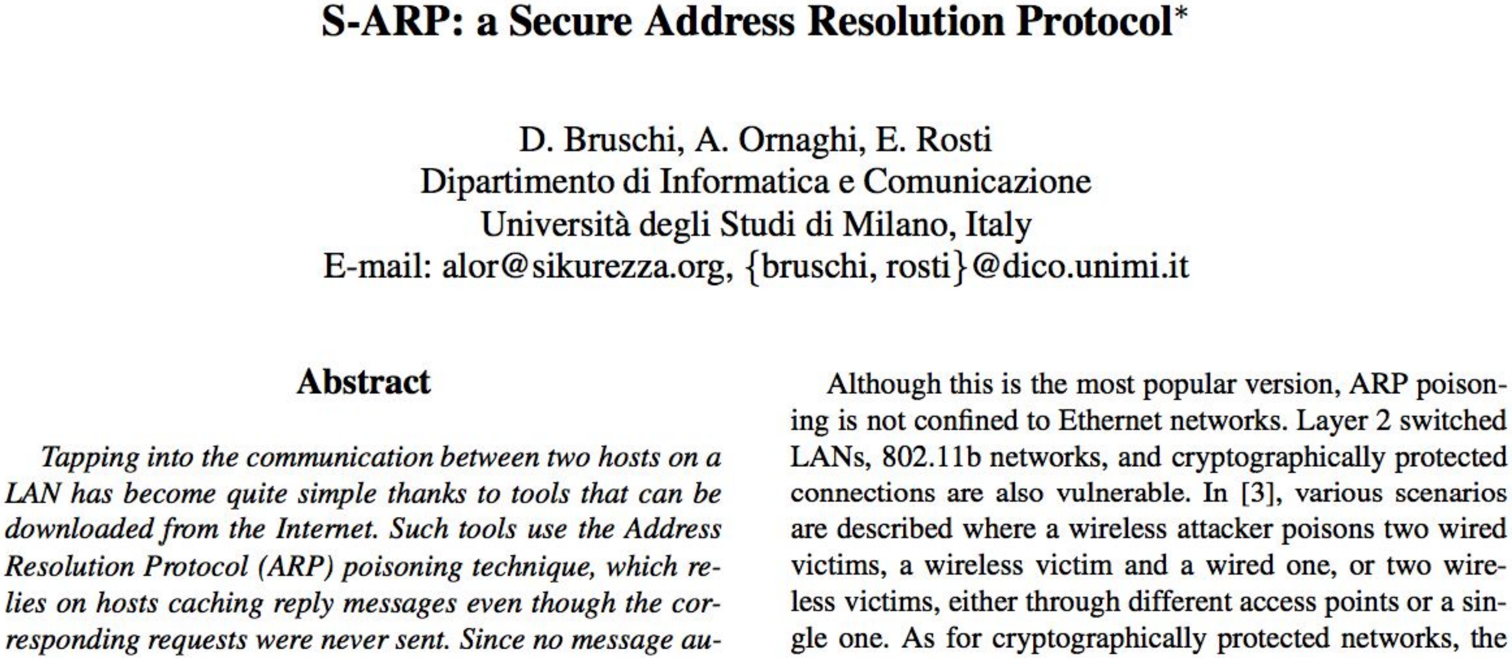
Defenses against ARP Spoofing

Arpwatch

- A free UNIX program which listens for ARP replies on a network.
- Build a table of IP/MAC associations and store it in a file.
- When a MAC/IP pair changes (flip-flop), an email is sent to an administrator.

- Useful for Gratuitous ARP Spoofing

S-ARP (ACSAC 2003)



S-ARP: a Secure Address Resolution Protocol*

D. Bruschi, A. Ornaghi, E. Rosti
Dipartimento di Informatica e Comunicazione
Università degli Studi di Milano, Italy
E-mail: alor@sikurezza.org, {bruschi, rosti}@dico.unimi.it

Abstract

Tapping into the communication between two hosts on a LAN has become quite simple thanks to tools that can be downloaded from the Internet. Such tools use the Address Resolution Protocol (ARP) poisoning technique, which relies on hosts caching reply messages even though the corresponding requests were never sent. Since no message au-

Although this is the most popular version, ARP poisoning is not confined to Ethernet networks. Layer 2 switched LANs, 802.11b networks, and cryptographically protected connections are also vulnerable. In [3], various scenarios are described where a wireless attacker poisons two wired victims, a wireless victim and a wired one, or two wireless victims, either through different access points or a single one. As for cryptographically protected networks, the

S-ARP Protocol Overview

- S-ARP provides message authentication only

- S-ARP uses asymmetric cryptography

- Any S-ARP enabled host is identified by its own IP address and has a public/private key pair

S-ARP Protocol Overview

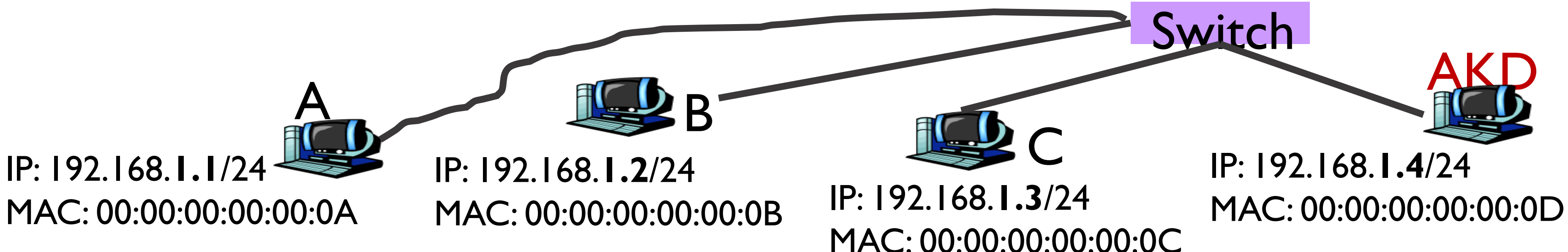
- A simple certificate provides the binding between the host identity and its public key
  - <IP, PubKey>

- A host that wants to connect to the LAN must first generate a public/private key pair and send its certificate to the AKD.

S-ARP Protocol Overview

- Authoritative Key Distributor (AKD) is a trusted host acting as key repository.

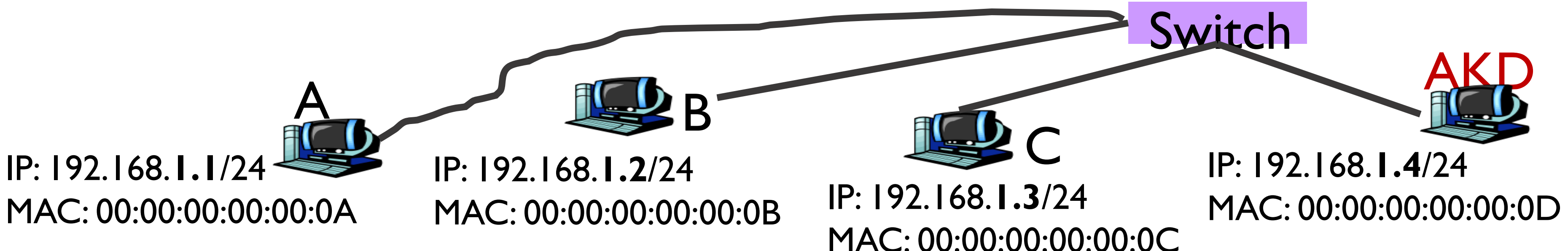
- The correctness of the certificate is verified by the network manager and the host public key together with its IP address is entered in the AKD repository.



S-ARP Protocol Overview

Each host sends its signed certificate containing the public key and the IP address to the AKD, which inserts the public key and the IP address in a local database, after the network manager’s validation

A□AKD:	<PubKeyA, 192.168.1.1>
B□AKD:	<PubKeyB, 192.168.1.2>
C□AKD:	<PubKeyC, 192.168.1.3>
~~C□AKD:~~	~~<PubKeyC, 192.168.1.2>~~



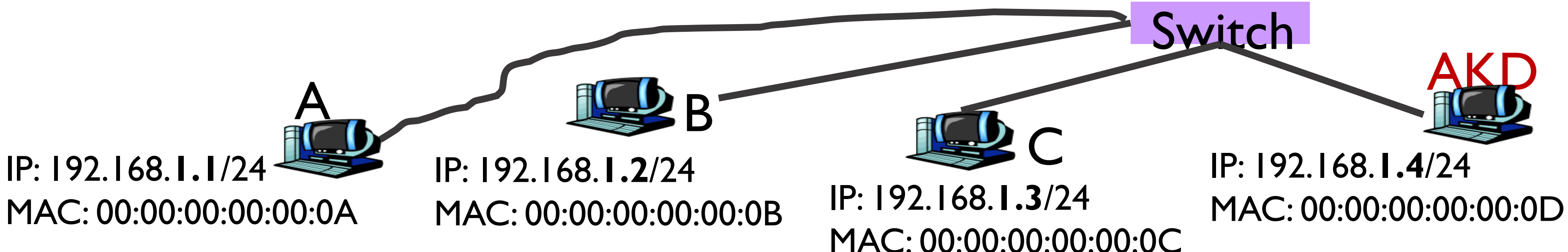
S-ARP Protocol Overview

In S-ARP all reply messages are digitally signed by the sender with the corresponding private key.

When Host A asks: What is the MAC of 192.168.1.2?

B replies “192.168.1.2 has MAC 0B” signed by B’s PrivateKey

<“192.168.1.2 has MAC 0B”, Signature>

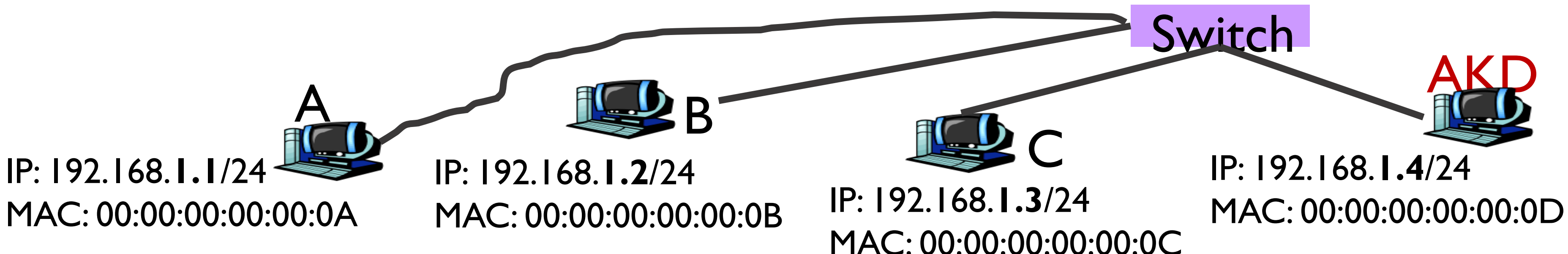


S-ARP Protocol Overview

“192.168.1.2 has MAC 0B” signed by B’s PrivateKey

<“192.168.1.2 has MAC 0B”, Signature>

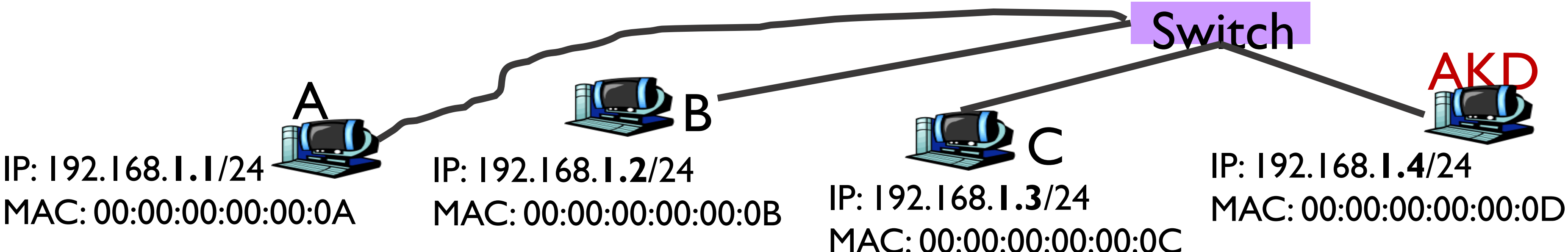
Host A retrieves public key of 192.168.1.2, which is B’s PubKey from AKD, verifies the signature is valid, then accepts 192.168.1.2 has MAC 0B.



S-ARP Protocol Overview

“192.168.1.2 has MAC 0C” signed by C’s PrivateKey
<“192.168.1.2 has MAC 0C”, Signature>

Host A retrieves public key of 192.168.1.2, which is B’s PubKey from AKD, then finds out the signature is not valid.


Switch
AKD
A
B
C
IP: 192.168.1.1/24
MAC: 00:00:00:00:00:0A
IP: 192.168.1.2/24
MAC: 00:00:00:00:00:0B
IP: 192.168.1.3/24
MAC: 00:00:00:00:00:0C
IP: 192.168.1.4/24
MAC: 00:00:00:00:00:0D