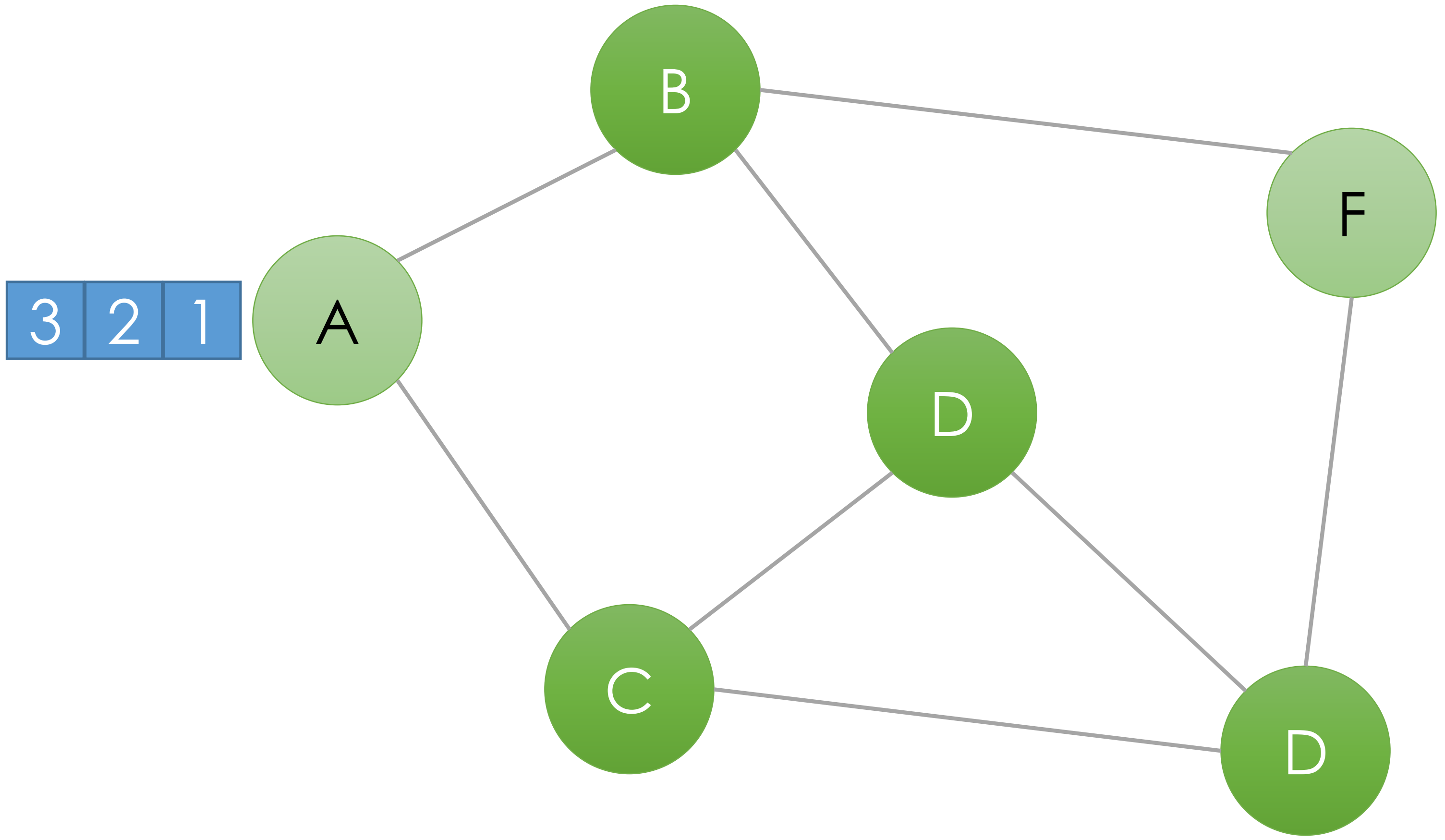
Computer Networks

Circuit and Packet Switching

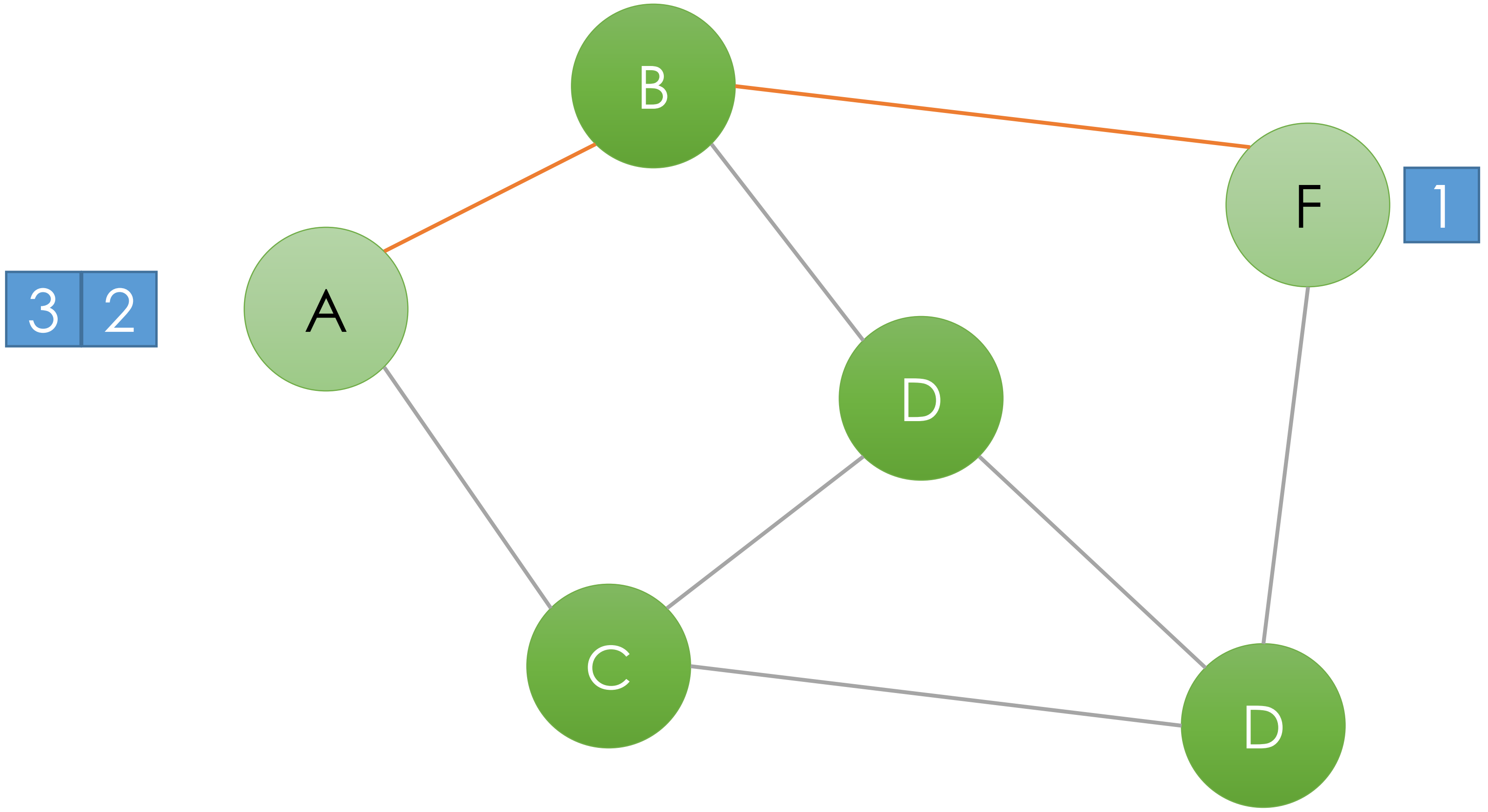
- Circuit switching
  - Legacy phone network
  - Single route through sequence of hardware devices established when two nodes start communication
  - Data sent along route
  - Route maintained until communication ends

- Packet switching
  - Internet
  - Data split into packets
  - Packets transported independently through network
  - Each packet handled on a best efforts basis
  - Packets may follow different routes

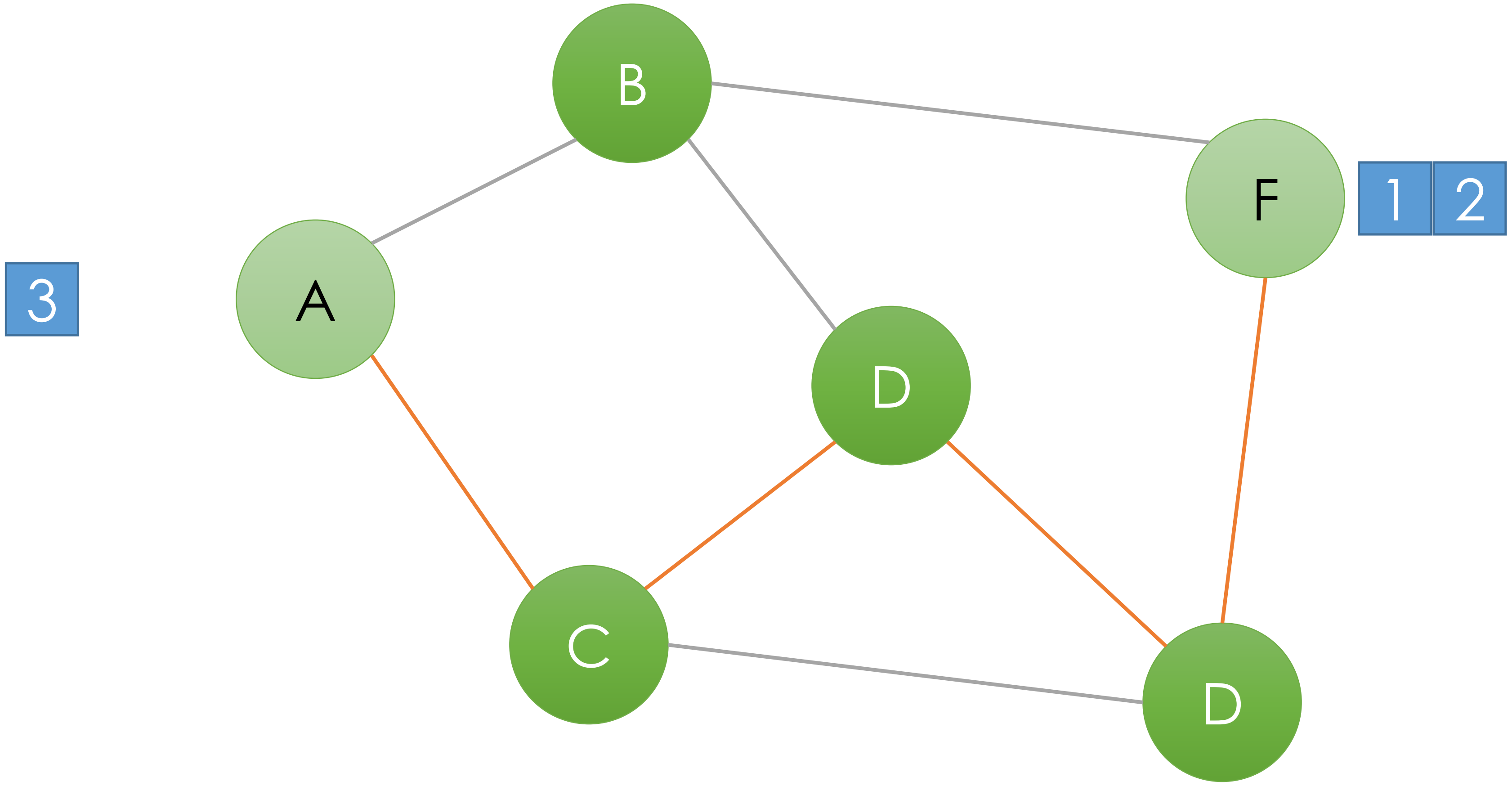
Packet Switching



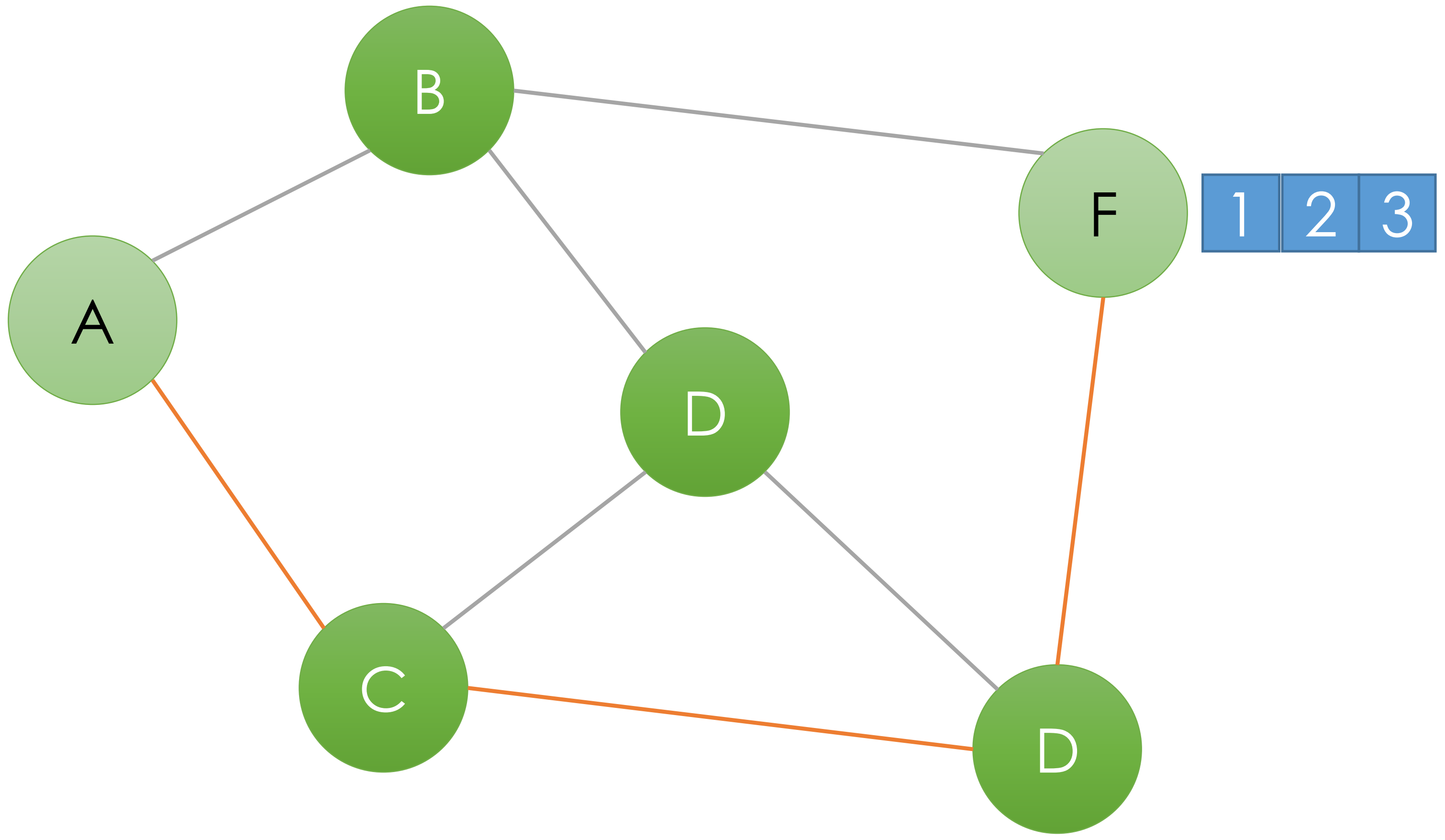
Packet Switching



Packet Switching



Packet Switching

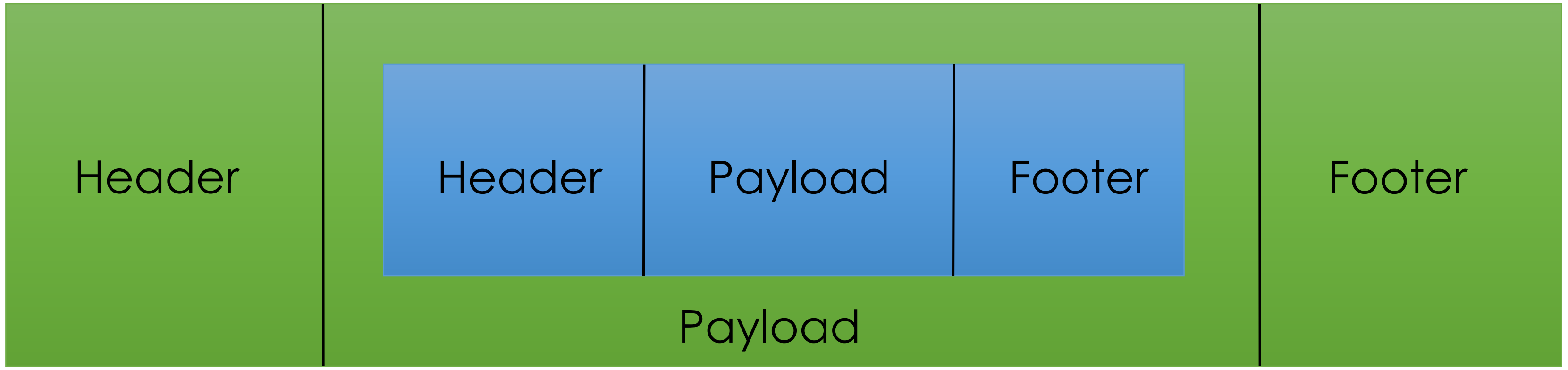


Protocols

- A protocol defines the rules for communication between computers
- Protocols are broadly classified as connectionless and connection oriented
- Connectionless protocol
  - Sends data out as soon as there is enough data to be transmitted
  - E.g., user datagram protocol (UDP)
- Connection-oriented protocol
  - Provides a reliable connection stream between two nodes
  - Consists of set up, transmission, and tear down phases
  - Creates virtual circuit-switched network
  - E.g., transmission control protocol (TCP)

Encapsulation

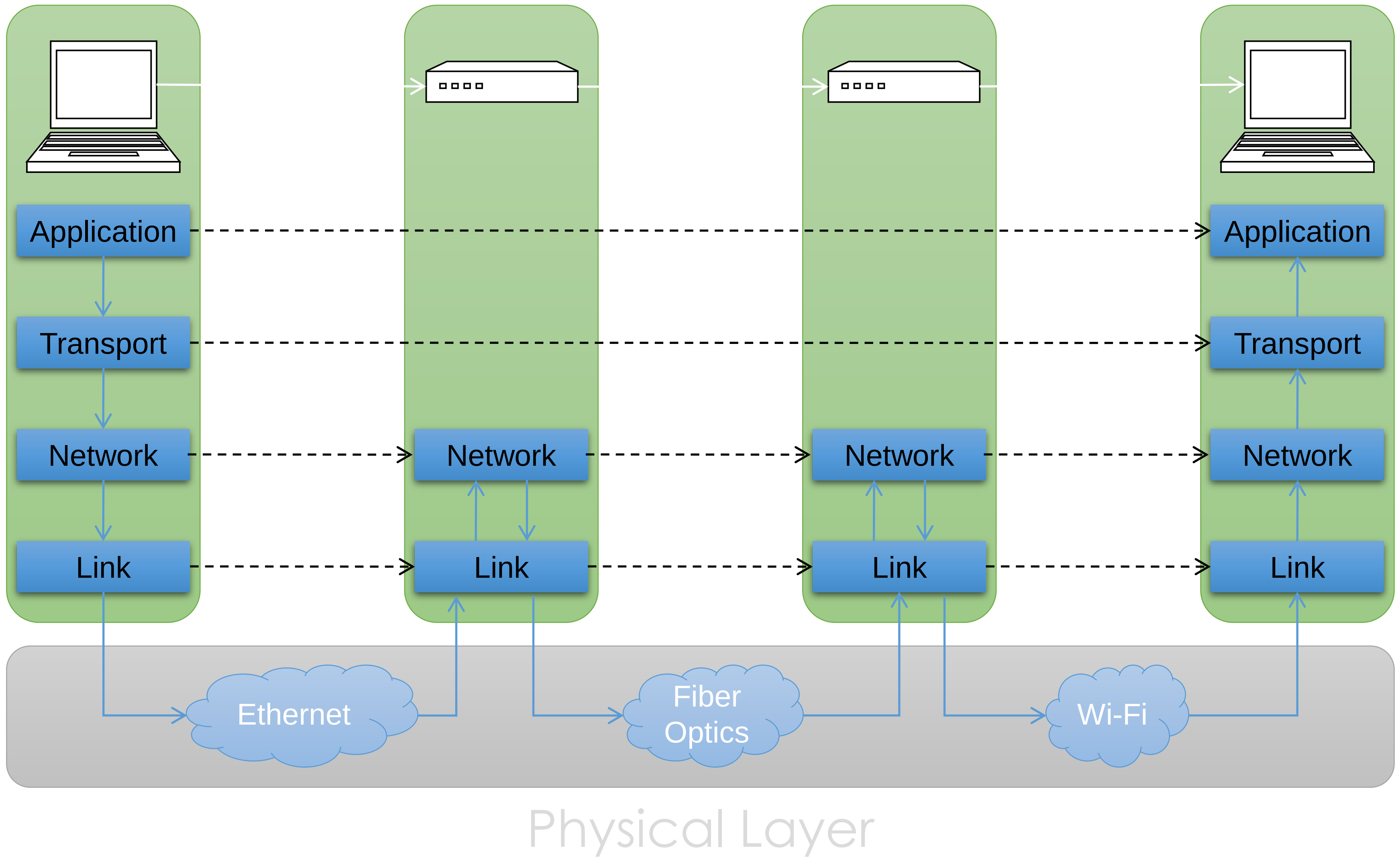
- A packet typically consists of
  - Control information for addressing the packet: header and footer
  - Data: payload
- A network protocol N1 can use the services of another network protocol N2
  - A packet p1 of N1 is encapsulated into a packet p2 of N2
  - The payload of p2 is p1
  - The control information of p2 is derived from that of p1



Network Layers

- Network models typically use a stack of layers
  - Higher layers use the services of lower layers via encapsulation
  - A layer can be implemented in hardware or software
  - The bottommost layer must be in hardware
- A network device may implement several layers
- A communication channel between two nodes is established for each layer
  - Actual channel at the bottom layer
  - Virtual channel at higher layers

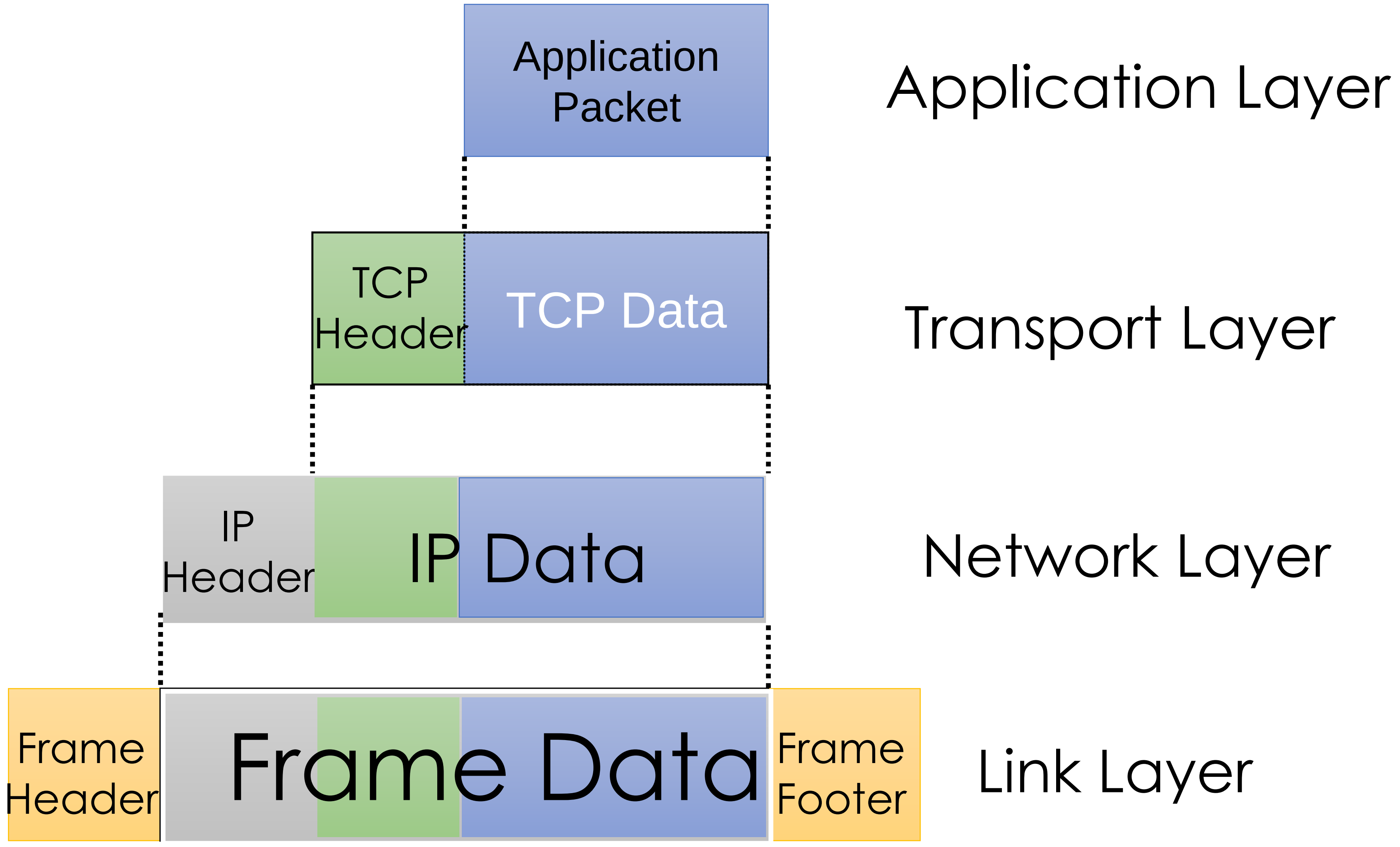
Internet Layers



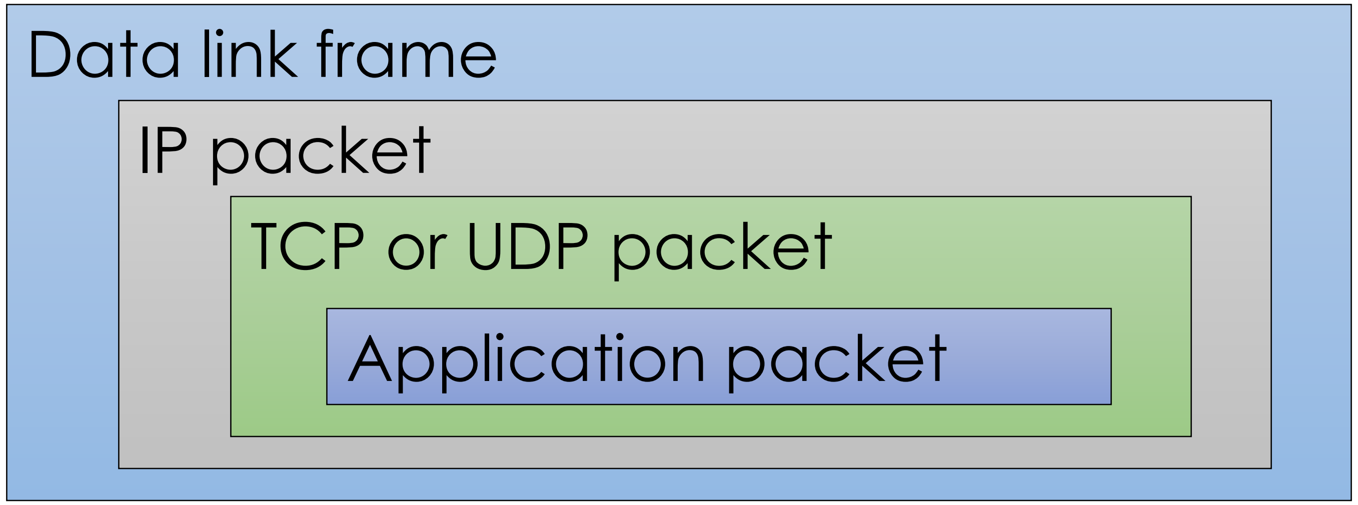
Intermediate Layers

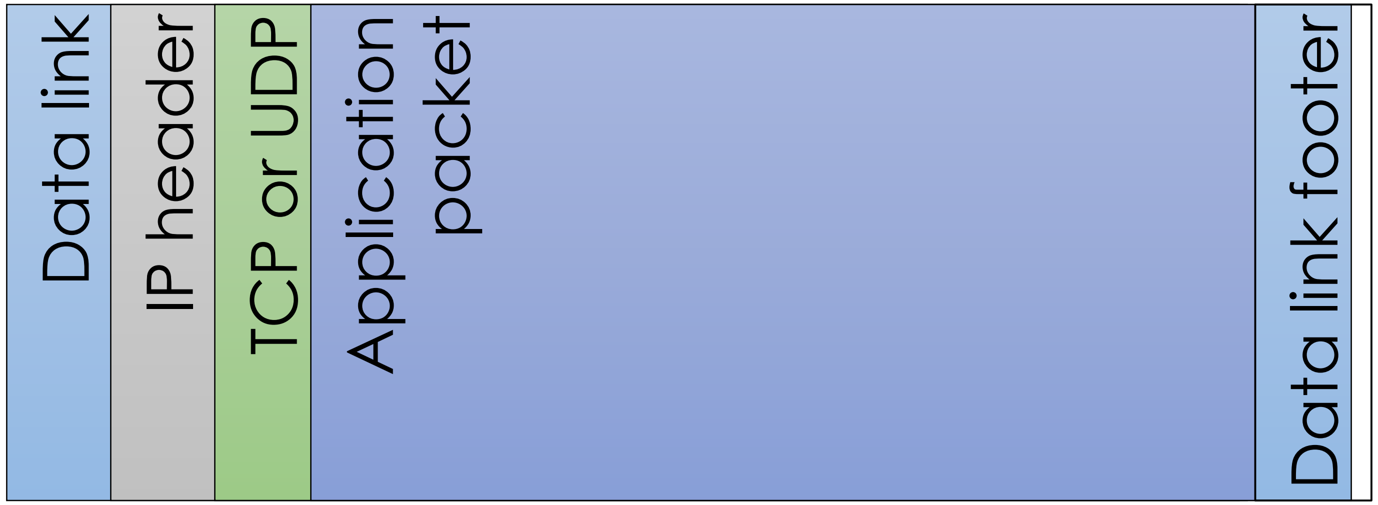
- Link layer
  - Local area network: Ethernet, WiFi, optical fiber
  - 48-bit media access control (MAC) addresses
  - Packets called frames
- Network layer
  - Internet-wide communication
  - Best efforts
  - 32-bit internet protocol (IP) addresses in IPv4
  - 128-bit IP addresses in IPv6
- Transport layer
  - 16-bit addresses (ports) for classes of applications
  - Connection-oriented transmission layer protocol (TCP)
  - Connectionless user datagram protocol (UDP)

Internet Packet Encapsulation



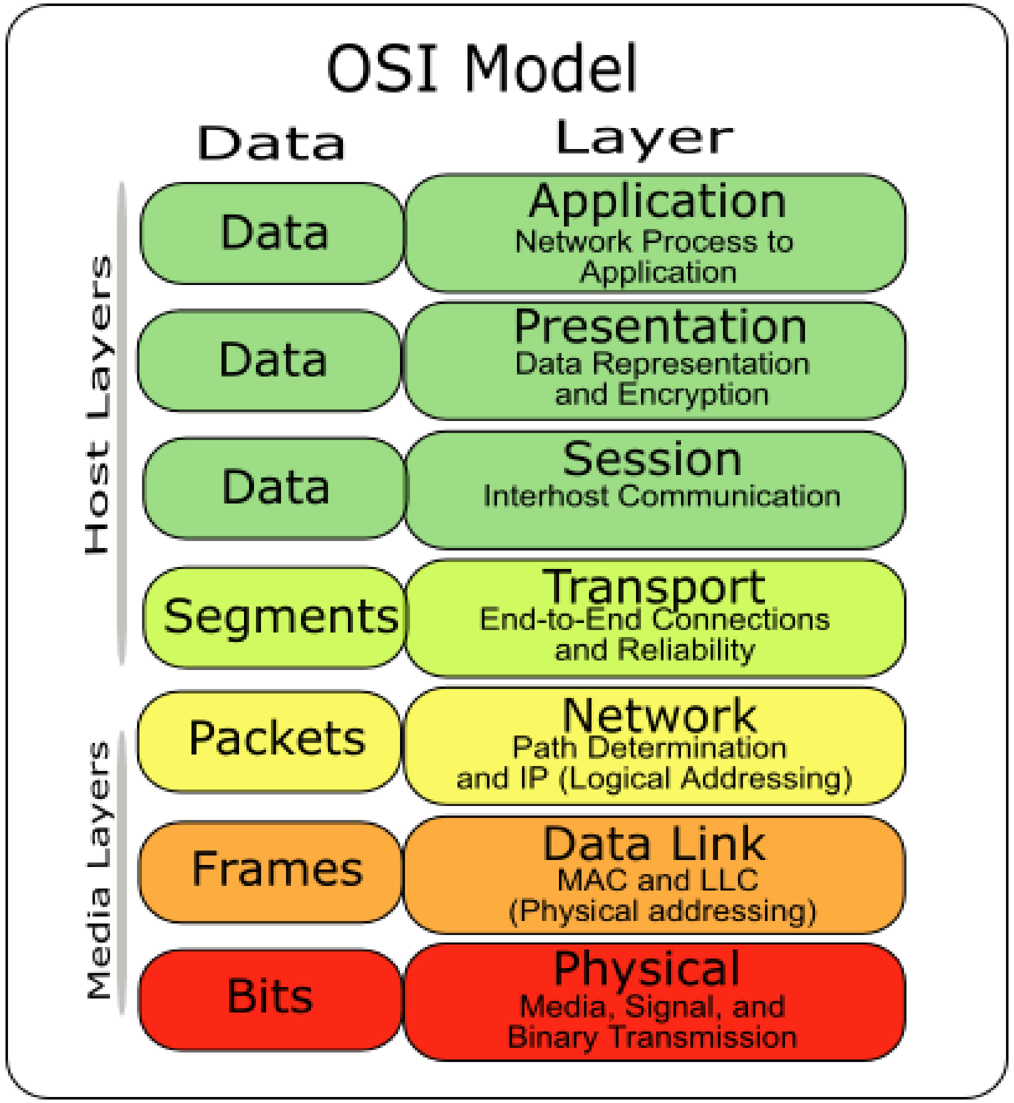
Internet Packet Encapsulation





The OSI Model

- The OSI (Open System Interconnect) Reference Model is a network model consisting of seven layers
- Created in 1983, OSI is promoted by the International Standard Organization (ISO)


OSI Model
Data
Layer
Host Layers
Media Layers
Data
Application
Network Process to Application
Data
Presentation
Data Representation and Encryption
Data
Session
Interhost Communication
Segments
Transport
End-to-End Connections and Reliability
Packets
Network
Path Determination and IP (Logical Addressing)
Frames
Data Link
MAC and LLC (Physical addressing)
Bits
Physical
Media, Signal, and Binary Transmission

Network Interfaces

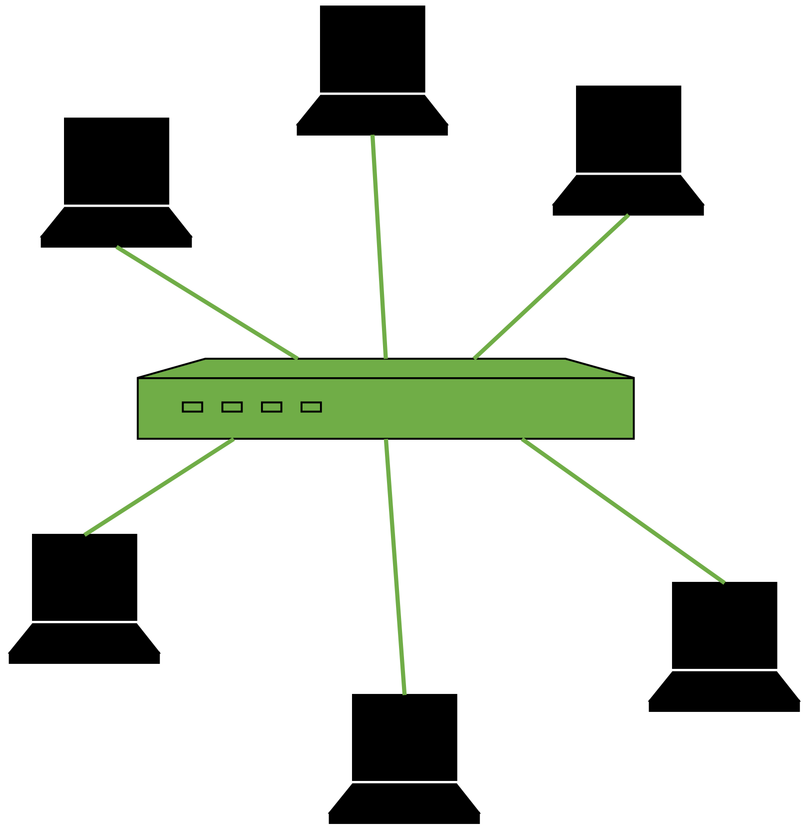
- Network interface: device connecting a computer to a network
  - Ethernet card
  - WiFi adapter
- A computer may have multiple network interfaces
- Packets transmitted between network interfaces
- Most local area networks, (including Ethernet and WiFi) broadcast frames
- In regular mode, each network interface gets the frames intended for it
- Traffic sniffing can be accomplished by configuring the network interface to read all frames (promiscuous mode)

MAC Addresses

- Most network interfaces come with a predefined MAC address
- A MAC address is a 48-bit number usually represented in hex
  - E.g., 00-1A-92-D4-BF-86
- The first three octets of any MAC address are IEEE-assigned Organizationally Unique Identifiers
  - E.g., Cisco 00-1A-A1, D-Link 00-1B-11, ASUSTek 00-1A-92
- The next three can be assigned by organizations as they please, with uniqueness being the only constraint
- Organizations can utilize MAC addresses to identify computers on their network
- MAC address can be reconfigured by network interface driver software

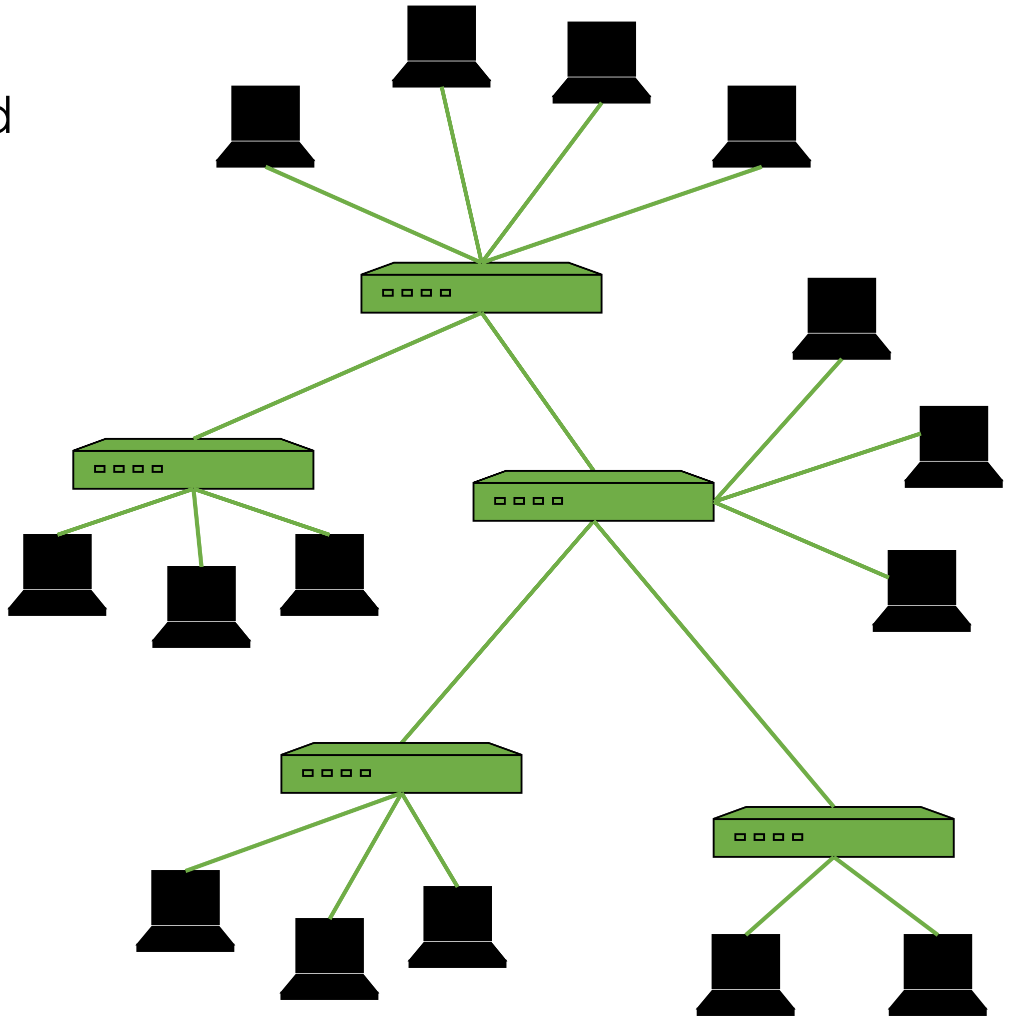
Switch

- A switch is a common network device
  - Operates at the link layer
  - Has multiple ports, each connected to a computer
- Operation of a switch
  - Learn the MAC address of each computer connected to it
  - Forward frames only to the destination computer



Combining Switches

- Switches can be arranged into a tree
- Each port learns the MAC addresses of the machines in the segment (subtree) connected to it
- Fragments to unknown MAC addresses are broadcast
- Frames to MAC addresses in the same segment as the sender are ignored



MAC Address Filtering

- A switch can be configured to provide service only to machines with specific MAC addresses
- Allowed MAC addresses need to be registered with a network administrator
- A MAC spoofing attack impersonates another machine
  - Find out MAC address of target machine
  - Reconfigure MAC address of rogue machine
  - Turn off or unplug target machine
- Countermeasures
  - Block port of switch when machine is turned off or unplugged
  - Disable duplicate MAC addresses

Viewing and Changing MAC Addresses

- Viewing the MAC addresses of the interfaces of a machine
  - Linux: `ifconfig`
  - Windows: `ipconfig /all`
- Changing a MAC address in Linux
  - Stop the networking service: `/etc/init.d/network stop`
  - Change the MAC address: `ifconfig eth0 hw ether <MAC-address>`
  - Start the networking service: `/etc/init.d/network start`
- Changing a MAC address in Windows
  - Open the Network Connections applet
  - Access the properties for the network interface
  - Click "Configure …"
  - In the advanced tab, change the network address to the desired value
- Changing a MAC address requires administrator privileges

ARP

- The address resolution protocol (ARP) connects the network layer to the data layer by converting IP addresses to MAC addresses
- ARP works by broadcasting requests and caching responses for future use
- The protocol begins with a computer broadcasting a message of the form

  who has <IP address1> tell <IP address2>

- When the machine with <IP address1> or an ARP server receives this message, its broadcasts the response

  <IP address1> is <MAC address>

- The requestor's IP address <IP address2> is contained in the link header
- The Linux and Windows command arp - a displays the ARP table

```
Internet Address       Physical Address       Type
128.148.31.1           00-00-0c-07-ac-00      dynamic
128.148.31.15          00-0c-76-b2-d7-1d      dynamic
128.148.31.71          00-0c-76-b2-d0-d2      dynamic
128.148.31.75          00-0c-76-b2-d7-1d      dynamic
128.148.31.102         00-22-0c-a3-e4-00      dynamic
128.148.31.137         00-1d-92-b6-f1-a9      dynamic
```

ARP Spoofing

- The ARP table is updated whenever an ARP response is received
- Requests are not tracked
- ARP announcements are not authenticated
- Machines trust each other
- A rogue machine can spoof other machines

ARP Poisoning (ARP Spoofing)

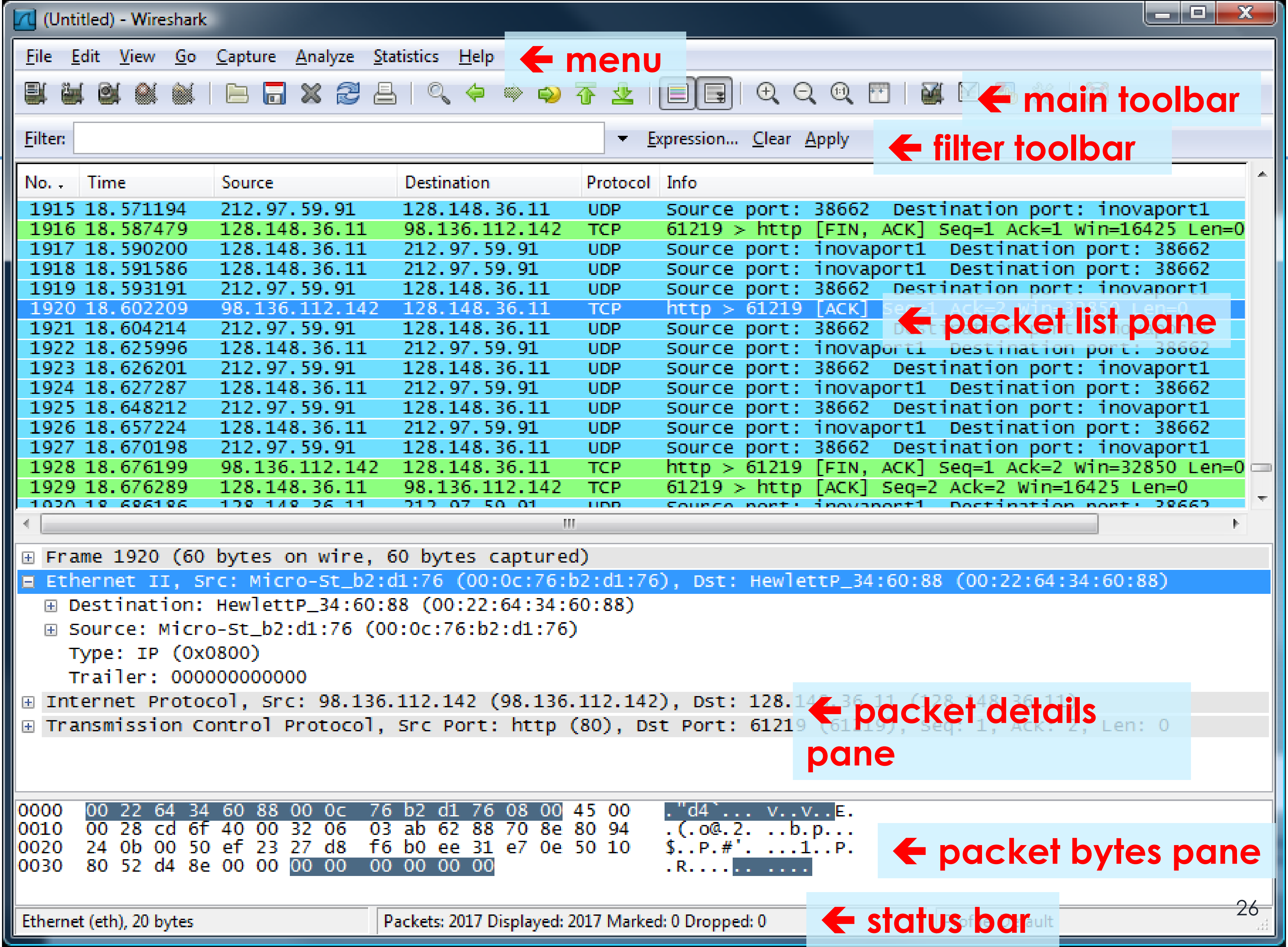
- According to the standard, almost all ARP implementations are stateless
- An arp cache updates every time that it receives an arp reply… even if it did not send any arp request!
- It is possible to "poison" an arp cache by sending gratuitous arp replies
- Using static entries solves the problem but it is almost impossible to manage!

Telnet Protocol (RFC 854)

- Telnet is a protocol that provides a general, bi-directional, not encrypted communication
- `telnet` is a generic TCP client
  - Allows a computer to connect to another one
  - Provides remote login capabilities to computers on the Internet
  - Sends whatever you type
  - Prints whatever comes back
  - Useful for testing TCP servers (ASCII based protocols)

Wireshark

- Wireshark is a packet sniffer and protocol analyzer
  - Captures and analyzes frames
  - Supports plugins
- Usually required to run with administrator privileges
- Setting the network interface in promiscuous mode captures traffic across the entire LAN segment and not just frames addressed to the machine
- Freely available on www.wireshark.org



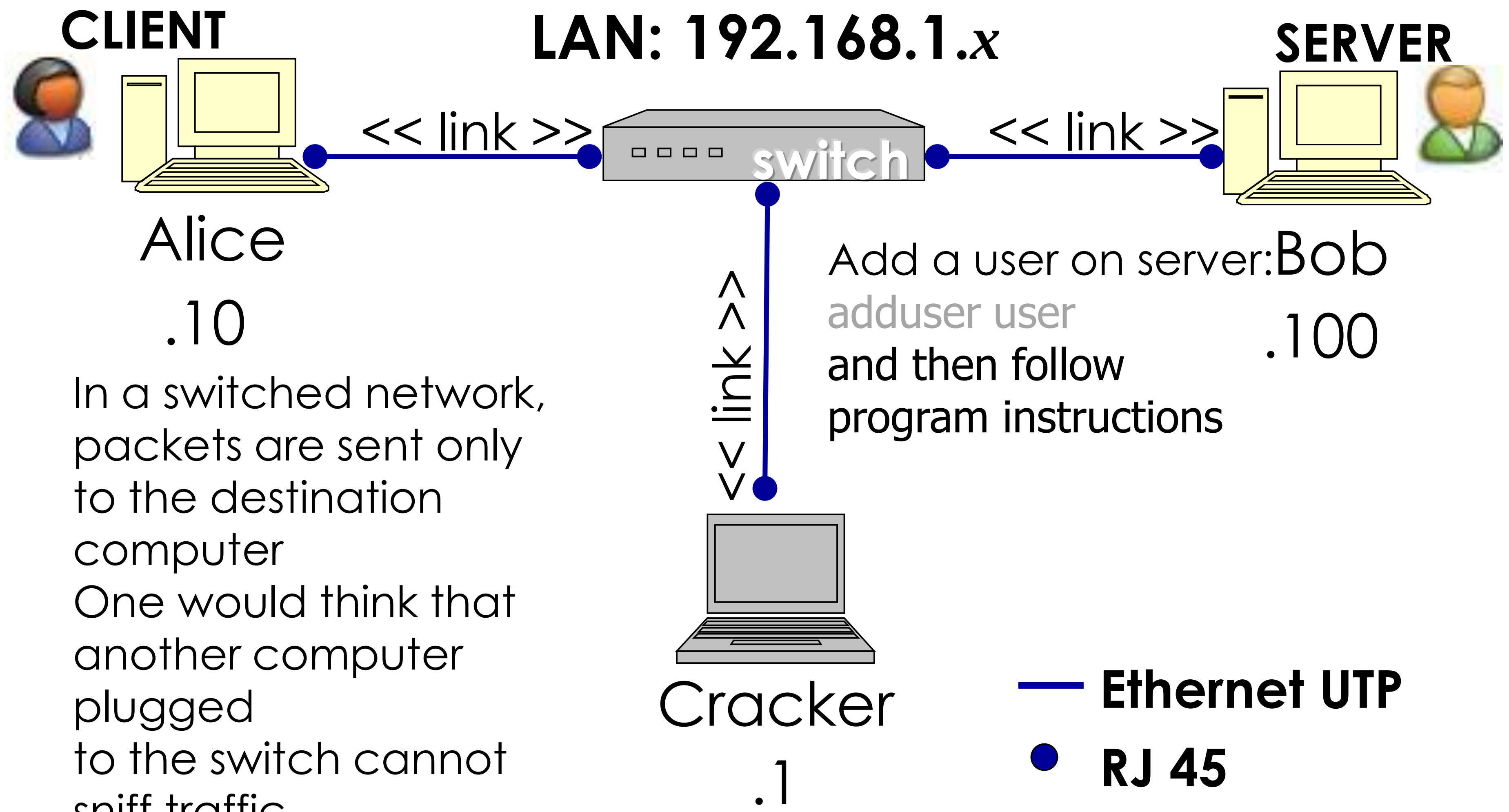
```
No. .  Time       Source          Destination     Protocol  Info
1915 18.571194 212.97.59.91    128.148.36.11   UDP   Source port: 38662  Destination port: inovaport1
1916 18.587479 128.148.36.11   98.136.112.142  TCP   61219 > http [FIN, ACK] Seq=1 Ack=1 Win=16425 Len=0
1917 18.590200 128.148.36.11   212.97.59.91    UDP   Source port: inovaport1  Destination port: 38662
1918 18.591586 128.148.36.11   212.97.59.91    UDP   Source port: inovaport1  Destination port: 38662
1919 18.593191 212.97.59.91    128.148.36.11   UDP   Source port: 38662  Destination port: inovaport1
1920 18.602209 98.136.112.142  128.148.36.11   TCP   http > 61219 [ACK] Seq=1 Ack=2 Win=32850 Len=0
1921 18.604214 212.97.59.91    128.148.36.11   UDP   Source port: 38662  Destination port: inovaport1
1922 18.625996 128.148.36.11   212.97.59.91    UDP   Source port: inovaport1  Destination port: 38662
1923 18.626201 212.97.59.91    128.148.36.11   UDP   Source port: 38662  Destination port: inovaport1
1924 18.627287 128.148.36.11   212.97.59.91    UDP   Source port: inovaport1  Destination port: 38662
1925 18.648212 212.97.59.91    128.148.36.11   UDP   Source port: 38662  Destination port: inovaport1
1926 18.657224 128.148.36.11   212.97.59.91    UDP   Source port: inovaport1  Destination port: 38662
1927 18.670198 212.97.59.91    128.148.36.11   UDP   Source port: 38662  Destination port: inovaport1
1928 18.676199 98.136.112.142  128.148.36.11   TCP   http > 61219 [FIN, ACK] Seq=1 Ack=2 Win=32850 Len=0
1929 18.676289 128.148.36.11   98.136.112.142  TCP   61219 > http [ACK] Seq=2 Ack=2 Win=16425 Len=0
1930 18.686186 128.148.36.11   212.97.59.91    UDP   Source port: inovaport1  Destination port: 38662
```

```
⊞ Frame 1920 (60 bytes on wire, 60 bytes captured)
⊟ Ethernet II, Src: Micro-St_b2:d1:76 (00:0c:76:b2:d1:76), Dst: HewlettP_34:60:88 (00:22:64:34:60:88)
   ⊞ Destination: HewlettP_34:60:88 (00:22:64:34:60:88)
   ⊞ Source: Micro-St_b2:d1:76 (00:0c:76:b2:d1:76)
     Type: IP (0x0800)
     Trailer: 000000000000
⊞ Internet Protocol, Src: 98.136.112.142 (98.136.112.142), Dst: 128.148.36.11 (128.148.36.11)
⊞ Transmission Control Protocol, Src Port: http (80), Dst Port: 61219 (61219), Seq: 1, Ack: 2, Len: 0
```

```
0000  00 22 64 34 60 88 00 0c  76 b2 d1 76 08 00 45 00   ."d4`... v..v..E.
0010  00 28 cd 6f 40 00 32 06  03 ab 62 88 70 8e 80 94   .(.o@.2. ..b.p...
0020  24 0b 00 50 ef 23 27 d8  f6 b0 ee 31 e7 0e 50 10   $..P.#'. ...1..P.
0030  80 52 d4 8e 00 00 00 00  00 00 00 00               .R...... ....
```

Ethernet (eth), 20 bytes | Packets: 2017 Displayed: 2017 Marked: 0 Dropped: 0 | Profile: Default

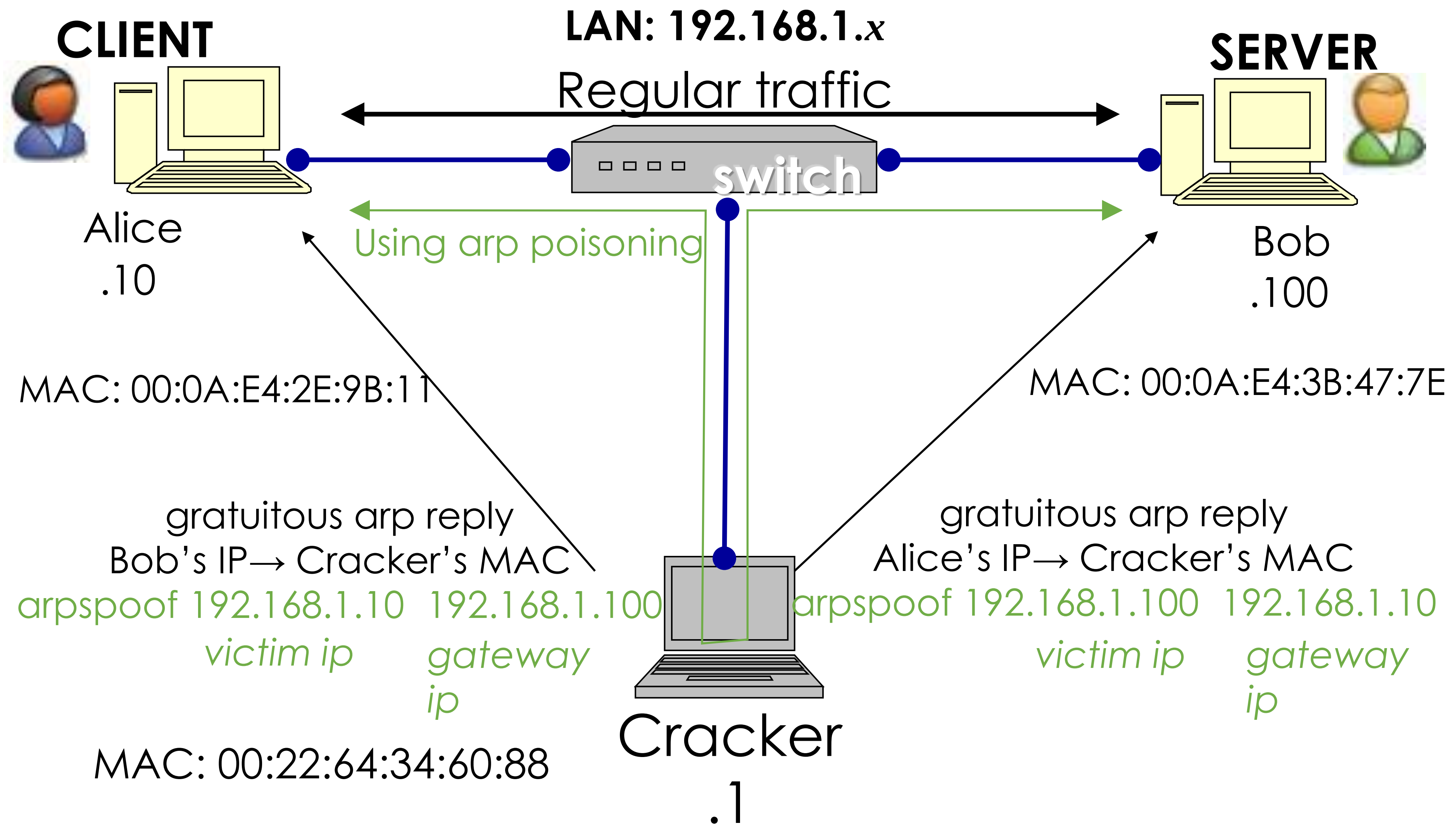
DEMO 1: Configuration using Telnet



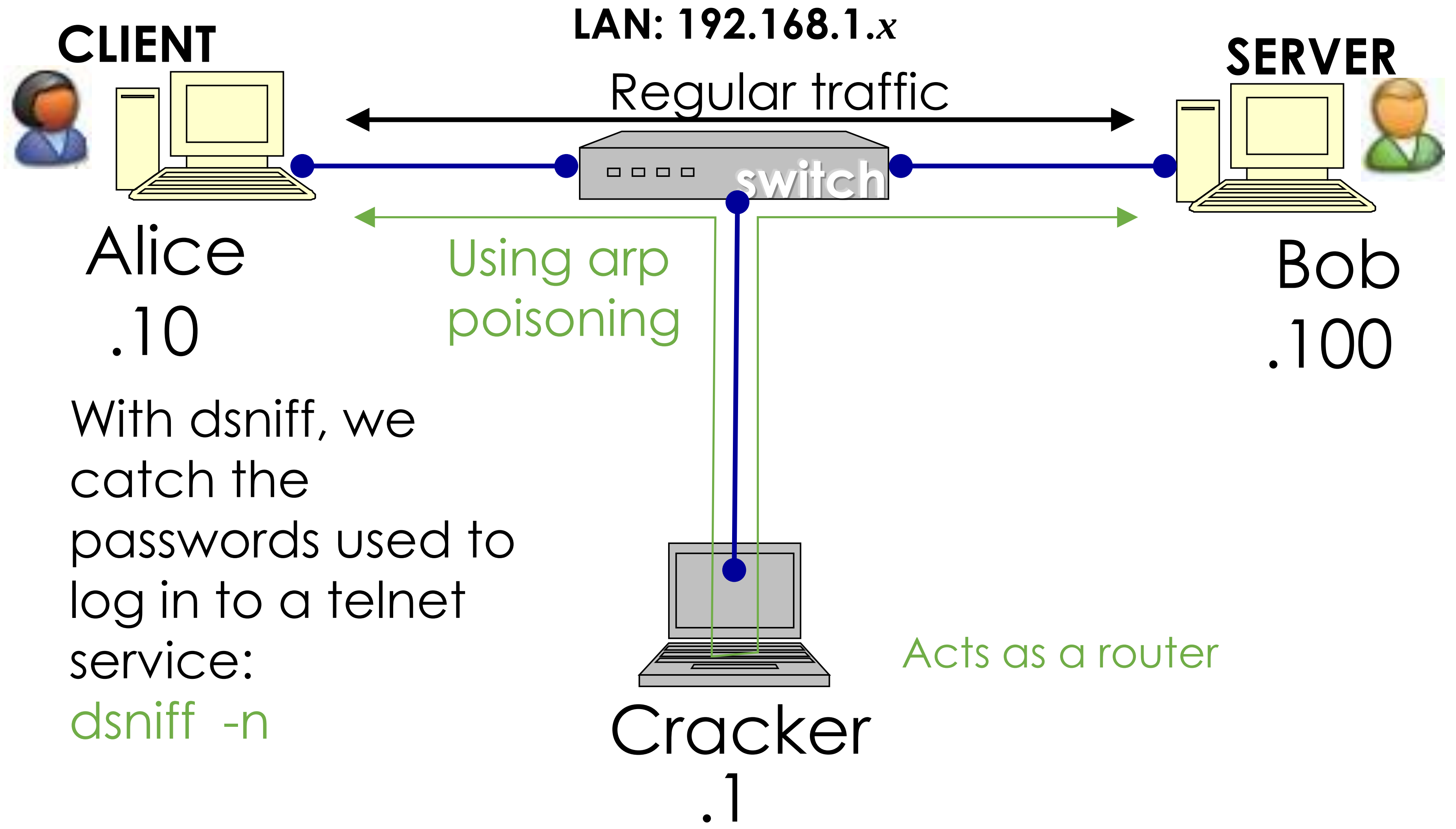
In a switched network, packets are sent only to the destination computer
One would think that another computer plugged
to the switch cannot sniff traffic

Add a user on server:
adduser user
and then follow program instructions

DEMO 1: ARP Spoofing



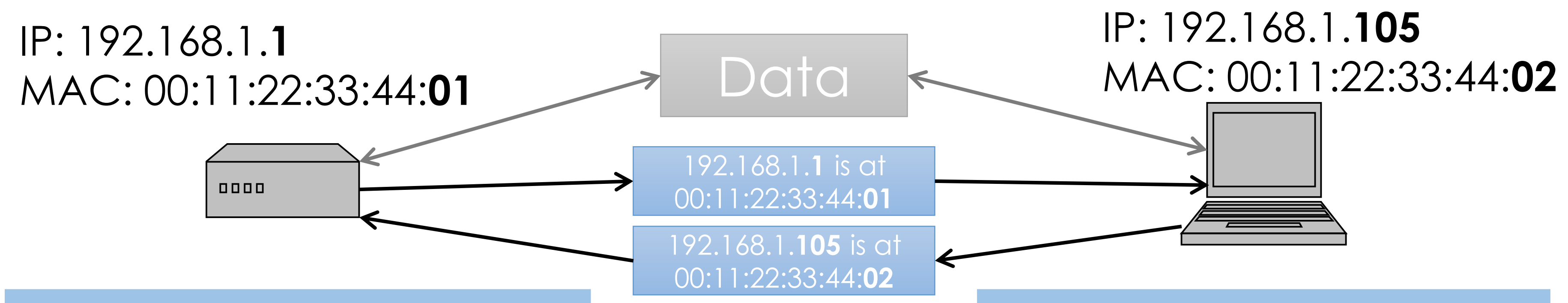
DEMO 1: catch telnet password



With dsniff, we catch the passwords used to log in to a telnet service:

dsniff -n

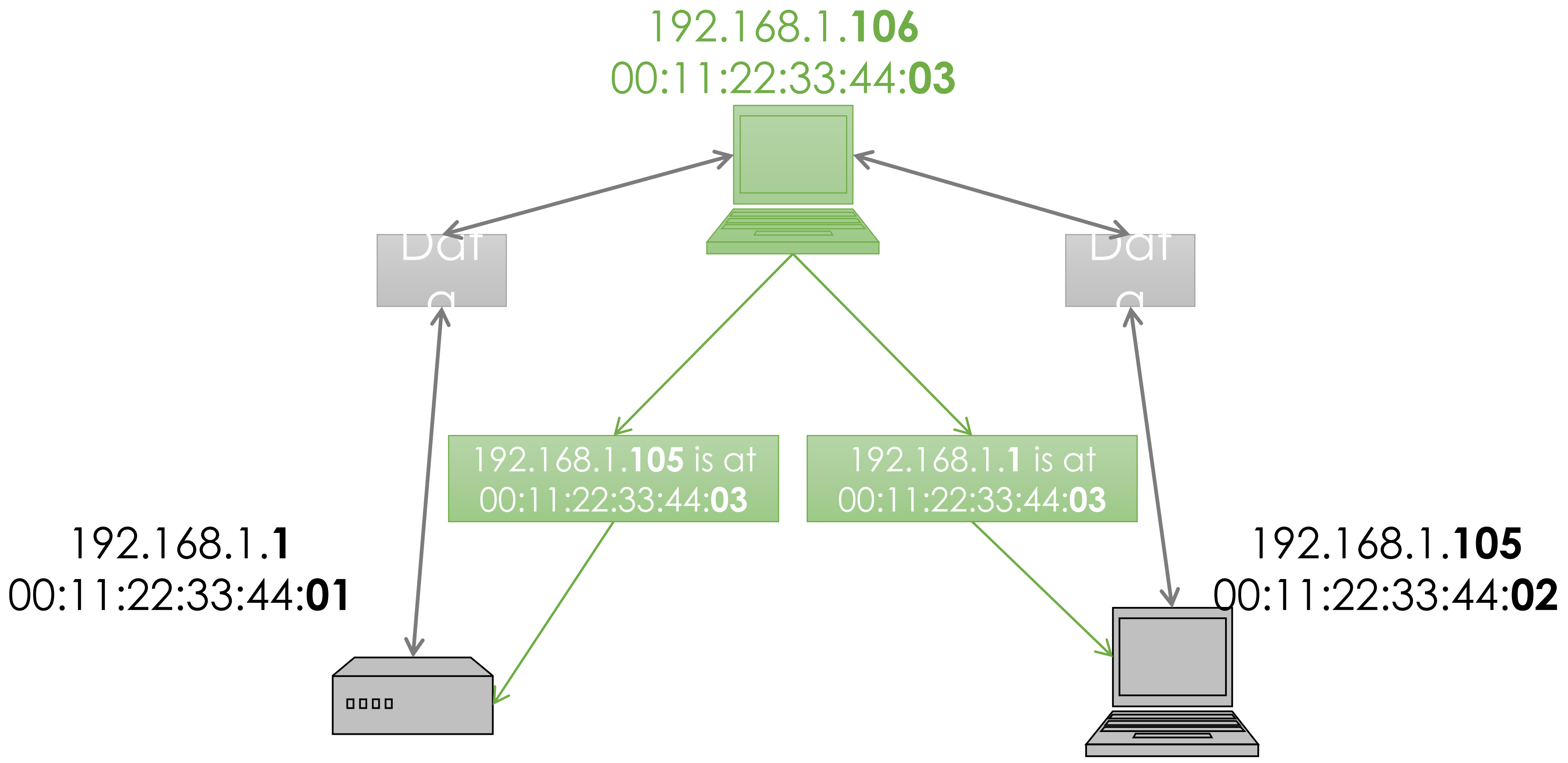
ARP Caches


IP: 192.168.1.**1**
MAC: 00:11:22:33:44:**01**
Data
IP: 192.168.1.**105**
MAC: 00:11:22:33:44:**02**
192.168.1.**1** is at 00:11:22:33:44:**01**
192.168.1.**105** is at 00:11:22:33:44:**02**

ARP Cache	
192.168.1.**105**	00:11:22:33:44:**02**

ARP Cache	
192.168.1.**1**	00:11:22:33:44:**01**

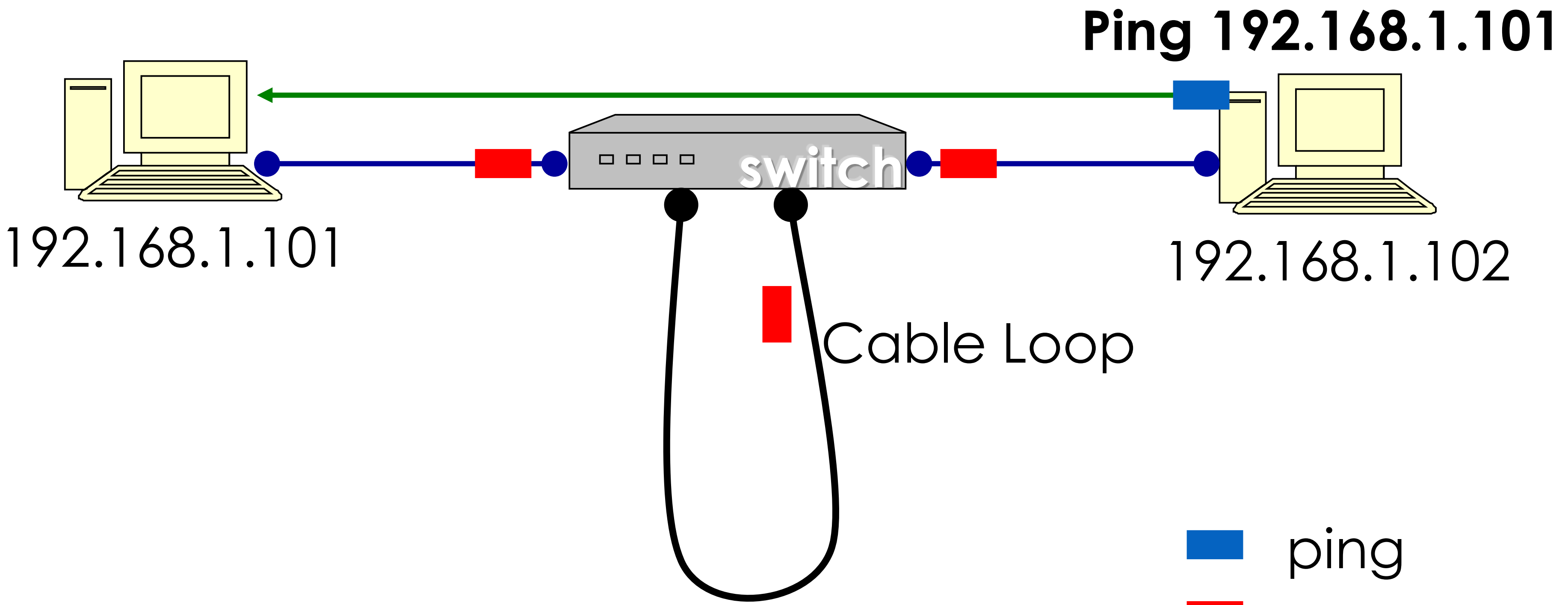
Poisoned ARP Caches



Poisoned ARP Cache	
192.168.1.**105**	00:11:22:33:44:**03**

Poisoned ARP Cache	
192.168.1.**1**	00:11:22:33:44:**03**

DEMO 2: network DOS using ARP



🟦 ping
🟥 arp request

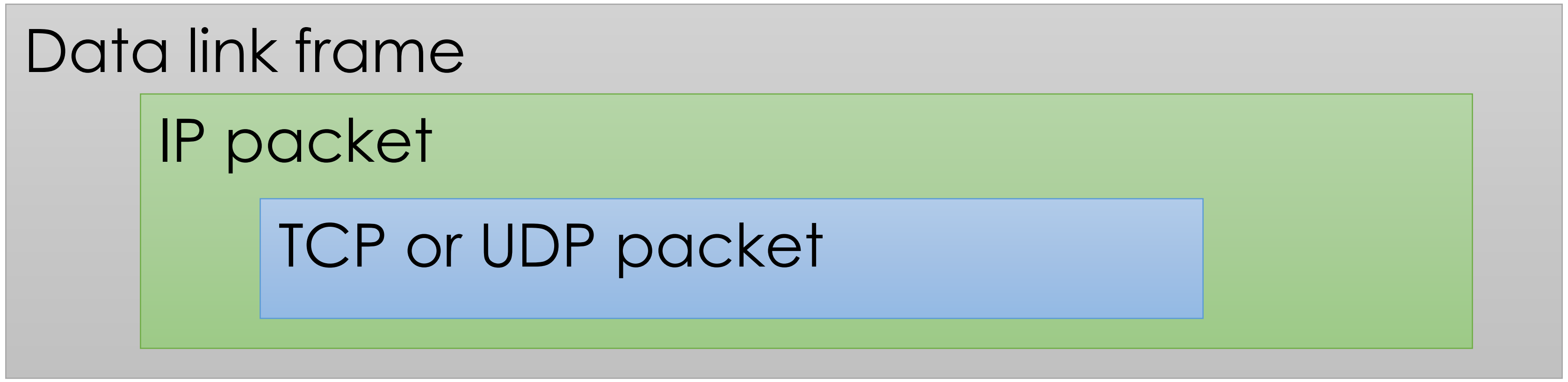
Broadcast storm

How can it be avoided?

Networks: IP and TCP

Internet Protocol

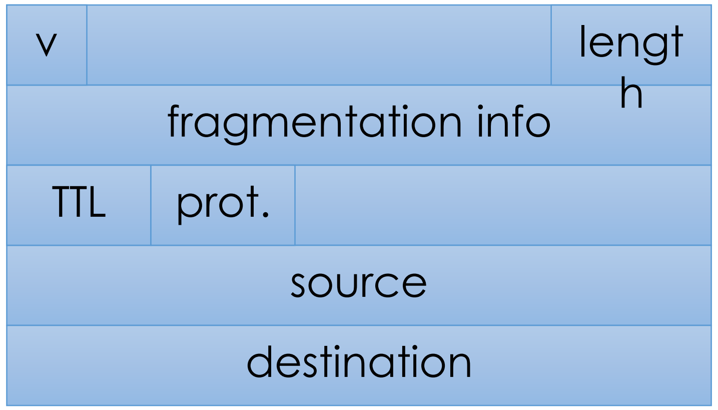
- Connectionless
  - Each packet is transported independently from other packets
- Unreliable
  - Delivery on a best effort basis
  - No acknowledgments
  - Packets may be lost, reordered, corrupted, or duplicated
- IP packets
  - Encapsulate TCP and UDP packets
  - Encapsulated into link-layer frames



IP Addresses and Packets

- IP addresses
  - IPv4: 32-bit addresses
  - IPv6: 128-bit addresses
- Address subdivided into network, subnet, and host
  - E.g., 128.148.32.110
- Broadcast addresses
  - E.g., 128.148.32.255
- Private networks
  - not routed outside of a LAN
  - 10.0.0.0/8
  - 172.16.0.0/12
  - 192.168.0.0/16

- IP header includes
  - Source address
  - Destination address
  - Packet length (up to 64KB)
  - Time to live (up to 255)
  - IP protocol version
  - Fragmentation information
  - Transport layer protocol information (e.g., TCP)



IP Address Space and ICANN

- Hosts on the internet must have unique IP addresses
- Internet Corporation for Assigned Names and Numbers
  - International nonprofit organization
  - Incorporated in the US
  - Allocates IP address space
  - Manages top-level domains
- Historical bias in favor of US corporations and nonprofit organizations

- Examples

003/8	May 94	General Electric
009/8	Aug 92	IBM
012/8	Jun 95	AT&T Bell Labs
013/8	Sep 91	Xerox Corporation
015/8	Jul 94	Hewlett-Packard
017/8	Jul 92	Apple Computer
018/8	Jan 94	MIT
019/8	May 95	Ford Motor
040/8	Jun 94	Eli Lily
043/8	Jan 91	Japan Inet
044/8	Jul 92	Amateur Radio Digital
047/8	Jan 91	Bell-Northern Res.
048/8	May 95	Prudential Securities
054/8	Mar 92	Merck
055/8	Apr 95	Boeing
056/8	Jun 94	U.S. Postal Service

A Typical University's IP Space

- Most universities separate their network connecting dorms and the network connecting offices and academic buildings
- Dorms
  - Class B network 138.16.0.0/16 (64K addresses)
- Academic buildings and offices
  - Class B network 128.148.0.0/16 (64K addresses)
- CS department
  - Several class C (/24) networks, each with 254 addresses

IP Routing

- A router bridges two or more networks
  - Operates at the network layer
  - Maintains tables to forward packets to the appropriate network
  - Forwarding decisions based solely on the destination address
- Routing table
  - Maps ranges of addresses to LANs or other gateway routers

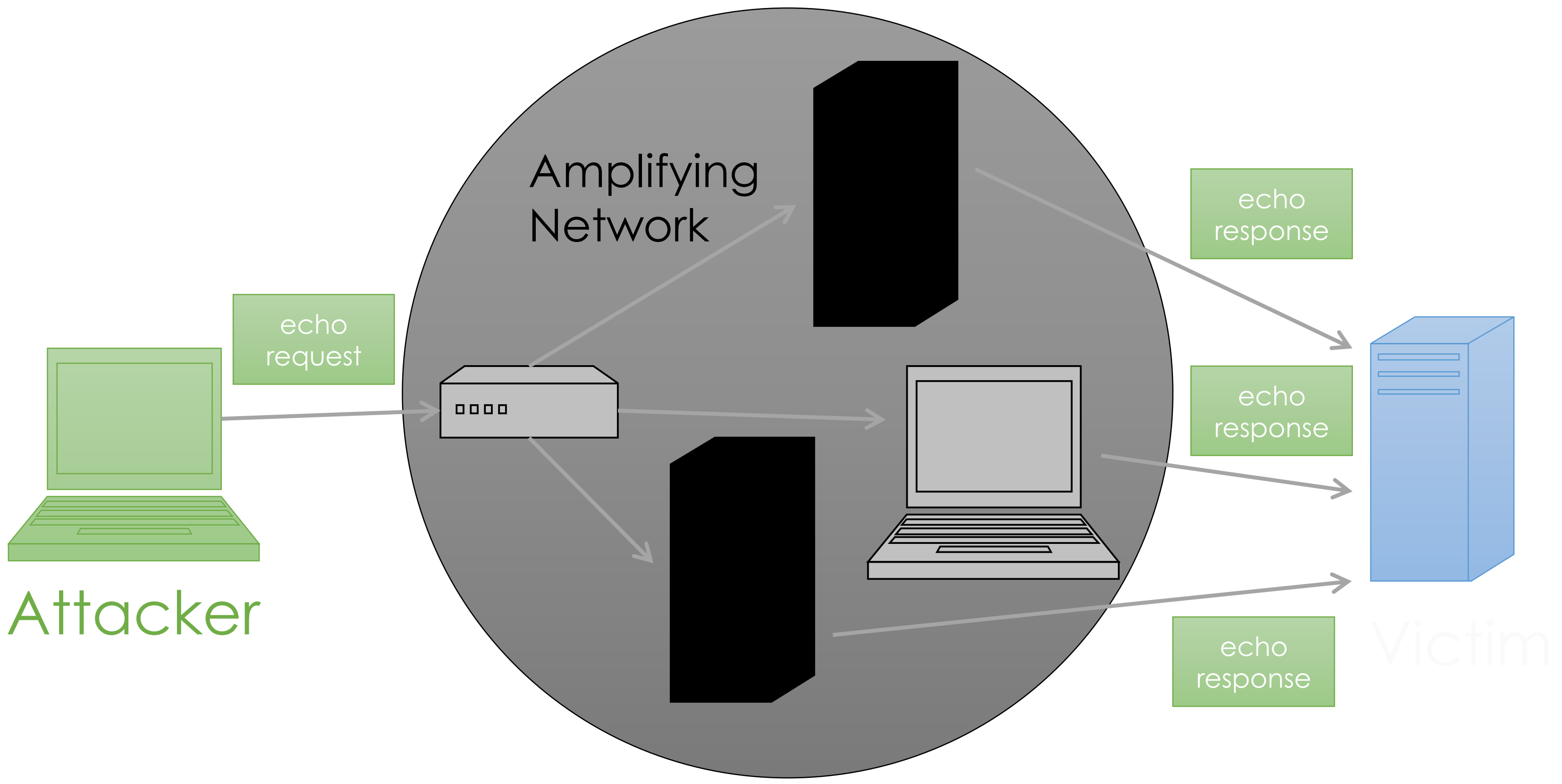
Internet Routes

- Internet Control Message Protocol (ICMP)
  - Used for network testing and debugging
  - Simple messages encapsulated in single IP packets
  - Considered a network layer protocol
- Tools based on ICMP
  - Ping: sends series of echo request messages and provides statistics on roundtrip times and packet loss
  - Traceroute: sends series ICMP packets with increasing TTL value to discover routes

ICMP Attacks

- Ping of death
  - ICMP specifies messages must fit a single IP packet (64KB)
  - Send a ping packet that exceeds maximum size using IP fragmentation
  - Reassembled packet caused several operating systems to crash due to a buffer overflow
- Smurf
  - Ping a broadcast address using a spoofed source address

Smurf Attack



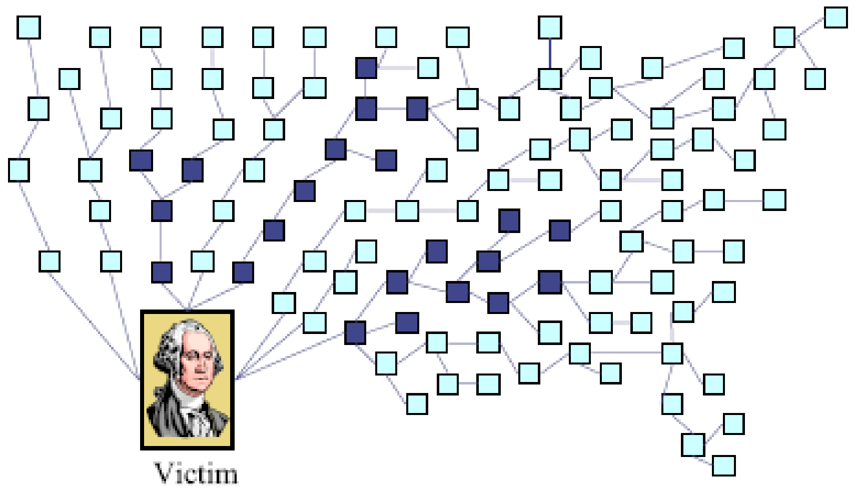
IP Vulnerabilities

- Unencrypted transmission
  - Eavesdropping possible at any intermediate host during routing
- No source authentication
  - Sender can spoof source address, making it difficult to trace packet back to attacker
- No integrity checking
  - Entire packet, header and payload, can be modified while en route to destination, enabling content forgeries, redirections, and man-in-the-middle attacks
- No bandwidth constraints
  - Large number of packets can be injected into network to launch a denial-of-service attack
  - Broadcast addresses provide additional leverage

Denial of Service Attack

- Send large number of packets to host providing service
  - Slows down or crashes host
  - Often executed by botnet
- Attack propagation
  - Starts at zombies
  - Travels through tree of internet routers rooted
  - Ends at victim
- IP source spoofing
  - Hides attacker
  - Scatters return traffic from victim

Source:
M.T. Goodrich, Probabalistic Packet Marking for Large-Scale IP Traceback, IEEE/ACM Transactions on Networking 16:1, 2008.



IP Traceback

- Problem
  - How to identify leaves of DoS propagation tree
  - Routers next to attacker
- Issues
  - There are more than 2M internet routers
  - Attacker can spoof source address
  - Attacker knows that traceback is being performed
- Approaches
  - Filtering and tracing (immediate reaction)
  - Messaging (additional traffic)
  - Logging (additional storage)
  - Probabilistic marking

Probabilistic Packet Marking

- Method
  - Random injection of information into packet header
  - Changes seldom used bits
  - Forward routing information to victim
  - Redundancy to survive packet losses
- Benefits
  - No additional traffic
  - No router storage
  - No packet size increase
  - Can be performed online or offline

Transmission Control Protocol

- TCP is a transport layer protocol guaranteeing reliable data transfer, in-order delivery of messages and the ability to distinguish data for multiple concurrent applications on the same host
- Most popular application protocols, including WWW, FTP and SSH are built on top of TCP
- TCP takes a stream of 8-bit byte data, packages it into appropriately sized segment and calls on IP to transmit these packets
- Delivery order is maintained by marking each packet with a sequence number
- Every time TCP receives a packet, it sends out an ACK to indicate successful receipt of the packet.
- TCP generally checks data transmitted by comparing a checksum of the data with a checksum encoded in the packet

Ports

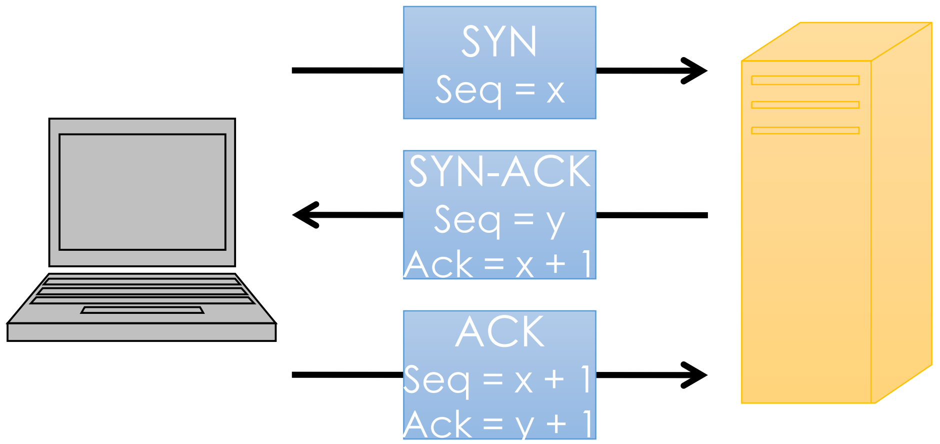
- TCP supports multiple concurrent applications on the same server
- Accomplishes this by having ports, 16 bit numbers identifying where data is directed
- The TCP header includes space for both a source and a destination port, thus allowing TCP to route all data
- In most cases, both TCP and UDP use the same port numbers for the same applications
- Ports 0 through 1023 are reserved for use by known protocols.
- Ports 1024 through 49151 are known as user ports, and should be used by most user programs for listening to connections and the like
- Ports 49152 through 65535 are private ports used for dynamic allocation by socket libraries

TCP Packet Format

Bit Offset	0-3	4-7	8-15	16-18	19-31
0	Source Port			Destination Port	
32	Sequence Number				
64	Acknowledgment Number				
96	Offset	Reserved	Flags	Window Size	
128	Checksum			Urgent Pointer	
160	Options				
>= 160	Payload				

Establishing TCP Connections

- TCP connections are established through a three way handshake.
- The server generally has a passive listener, waiting for a connection request
- The client requests a connection by sending out a SYN packet
- The server responds by sending a SYN/ACK packet, indicating an acknowledgment for the connection
- The client responds by sending an ACK to the server thus establishing connection



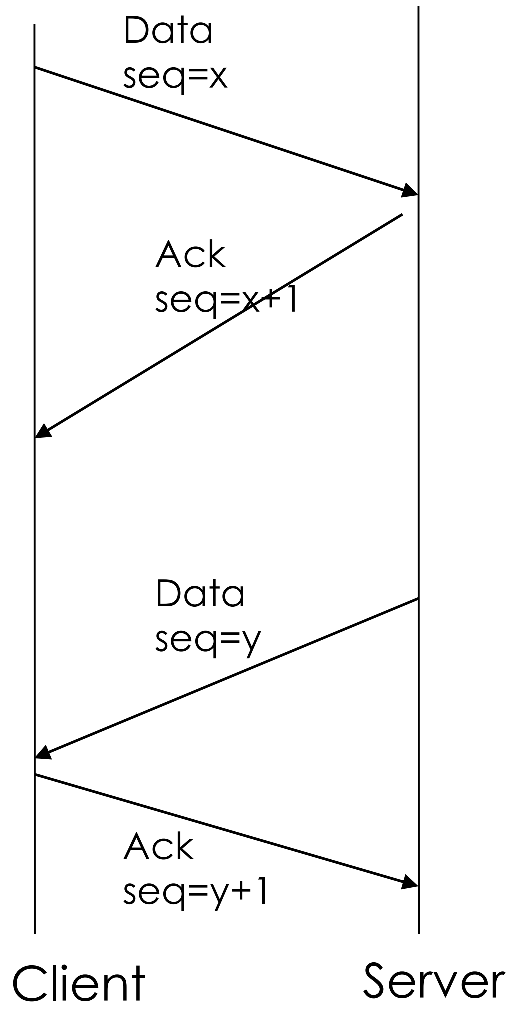
SYN Flood

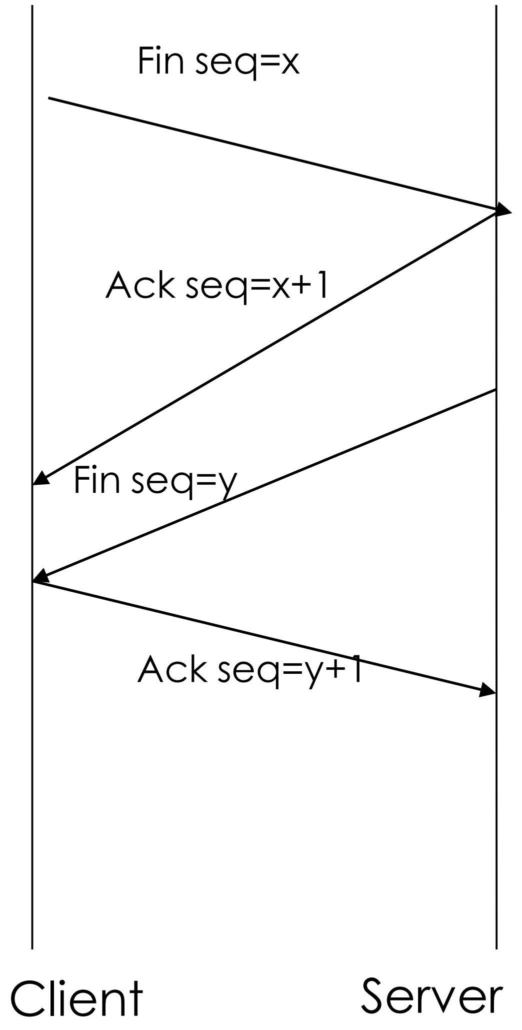
- Typically DOS attack, though can be combined with other attack such as TCP hijacking
- Rely on sending TCP connection requests faster than the server can process them
- Attacker creates a large number of packets with spoofed source addresses and setting the SYN flag on these
- The server responds with a SYN/ACK for which it never gets a response (waits for about 3 minutes each)
- Eventually the server stops accepting connection requests, thus triggering a denial of service.
- Can be solved in multiple ways
- One of the common way to do this is to use SYN cookies

TCP Data Transfer

- During connection initialization using the three way handshake, initial sequence numbers are exchanged
- The TCP header includes a 16 bit checksum of the data and parts of the header, including the source and destination
- Acknowledgment or lack thereof is used by TCP to keep track of network congestion and control flow and such
- TCP connections are cleanly terminated with a 4-way handshake
  - The client which wishes to terminate the connection sends a FIN message to the other client
  - The other client responds by sending an ACK
  - The other client sends a FIN
  - The original client now sends an ACK, and the connection is terminated

TCP Data Transfer and Teardown





TCP Congestion Control

- During the mid-80s it was discovered that uncontrolled TCP messages were causing large scale network congestion
- TCP responded to congestion by retransmitting lost packets, thus making the problem was worse
- What is predominantly used today is a system where ACKs are used to determine the maximum number of packets which should be sent out
- Most TCP congestion avoidance algorithms, avoid congestion by modifying a congestion window (cwnd) as more cumulative ACKs are received
- Lost packets are taken to be a sign of network congestion
- TCP begins with an extremely low cwnd and rapidly increases the value of this variable to reach bottleneck capacity
- At this point it shifts to a collision detection algorithm which slowly probes the network for additional bandwidth
- TCP congestion control is a good idea in general but allows for certain attacks.

Optimistic ACK Attack

- An optimistic ACK attack takes advantage of the TCP congestion control
- It begins with a client sending out ACKs for data segments it hasn't yet received
- This flood of optimistic ACKs makes the servers TCP stack believe that there is a large amount of bandwidth available and thus increase cwnd
- This leads to the attacker providing more optimistic ACKs, and eventually bandwidth use beyond what the server has available
- This can also be played out across multiple servers, with enough congestion that a certain section of the network is no longer reachable
- There are no practical solutions to this problem

Session Hijacking

- Also commonly known as TCP Session Hijacking
- A security attack over a protected network
- Attempt to take control of a network session
- Sessions are server keeping state of a client’s connection
- Servers need to keep track of messages sent between client and the server and their respective actions
- Most networks follow the TCP/IP protocol
- IP Spoofing is one type of hijacking on large network

IP Spoofing

- IP Spoofing is an attempt by an intruder to send packets from one IP address that appear to originate at another
- If the server thinks it is receiving messages from the real source after authenticating a session, it could inadvertently behave maliciously
- There are two basic forms of IP Spoofing
    - Blind Spoofing
        - Attack from any source
    - Non-Blind Spoofing
        - Attack from the same subnet

Blind IP Spoofing

- The TCP/IP protocol requires that “acknowledgement” numbers be sent across sessions
- Makes sure that the client is getting the server’s packets and vice versa
- Need to have the right sequence of acknowledgment numbers to spoof an IP identity

Non-Blind IP Spoofing

- IP Spoofing without inherently knowing the acknowledgment sequence pattern
  - Done on the same subnet
  - Use a packet sniffer to analyze the sequence pattern
    - Packet sniffers intercept network packets
    - Eventually decodes and analyzes the packets sent across the network
    - Determine the acknowledgment sequence pattern from the packets
    - Send messages to server with actual client's IP address and with validly sequenced acknowledgment number

Packet Sniffers

- Packet sniffers “read” information traversing a network
  - Packet sniffers intercept network packets, possibly using ARP cache poisoning
  - Can be used as legitimate tools to analyze a network
    - Monitor network usage
    - Filter network traffic
    - Analyze network problems
  - Can also be used maliciously
    - Steal information (i.e. passwords, conversations, etc.)
    - Analyze network information to prepare an attack
- Packet sniffers can be either software or hardware based
  - Sniffers are dependent on network setup

Detecting Sniffers

- Sniffers are almost always passive
  - They simply collect data
  - They do not attempt “entry” to “steal” data
- This can make them extremely hard to detect
- Most detection methods require suspicion that sniffing is occurring
  - Then some sort of “ping” of the sniffer is necessary
  - It should be a broadcast that will cause a response only from a sniffer
- Another solution on switched hubs is ARP watch
  - An ARP watch monitors the ARP cache for duplicate entries of a machine
  - If such duplicates appear, raise an alarm
  - Problem: false alarms
    - Specifically, DHCP networks can have multiple entires for a single machine

Stopping Packet Sniffing

- The best way is to encrypt packets securely
  - Sniffers can capture the packets, but they are meaningless
    - Capturing a packet is useless if it just reads as garbage
  - SSH is also a much more secure method of connection
    - Private/Public key pairs makes sniffing virtually useless
  - On switched networks, almost all attacks will be via ARP spoofing
    - Add machines to a permanent store in the cache
    - This store cannot be modified via a broadcast reply
    - Thus, a sniffer cannot redirect an address to itself
- The best security is to not let them in in the first place
  - Sniffers need to be on your subnet in a switched hub in the first place
  - All sniffers need to somehow access root at some point to start themselves up

Port Knocking

- Broadly port knocking is the act of attempting to make connections to blocked ports in a certain order in an attempt to open a port
- Port knocking is fairly secure against brute force attacks since there are 65536^k combinations, where k is the number of ports knocked
- Port knocking however if very susceptible to replay attacks. Someone can theoretically record port knocking attempts and repeat those to get the same open port again
- One good way of protecting against replay attacks would be a time dependent knock sequence.

User Datagram Protocol

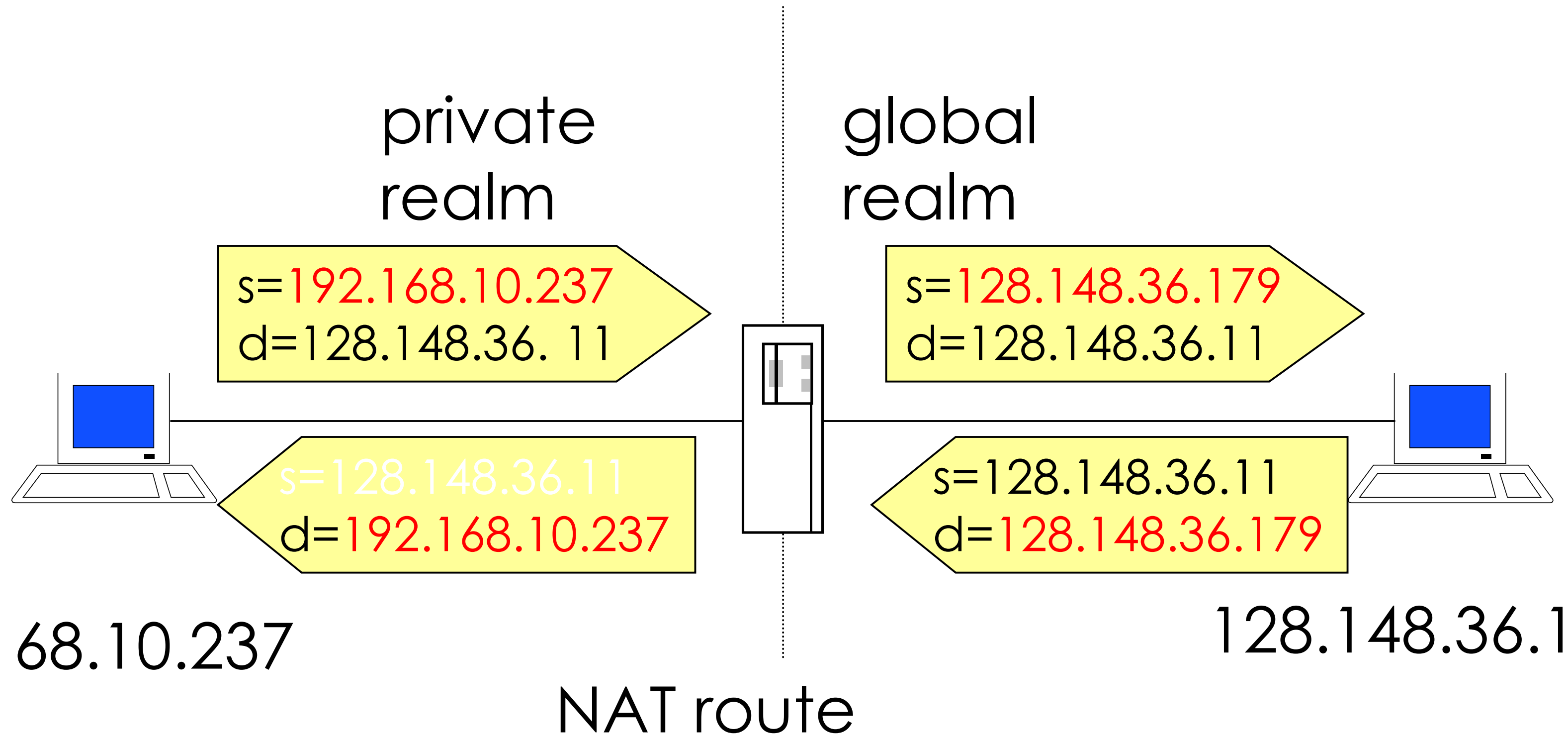
- UDP is a stateless, unreliable datagram protocol built on top of IP, that is it lies on level 4
- It does not provide delivery guarantees, or acknowledgments, but is significantly faster
- Can however distinguish data for multiple concurrent applications on a single host.
- A lack of reliability implies applications using UDP must be ready to accept a fair amount of error packages and data loss. Some application level protocols such as TFTP build reliability on top of UDP.
    - Most applications used on UDP will suffer if they have reliability. VoIP, Streaming Video and Streaming Audio all use UDP.
- UDP does not come with built in congestion protection, so while UDP does not suffer from the problems associated with optimistic ACK, there are cases where high rate UDP network access will cause congestion.

Network Address Translation

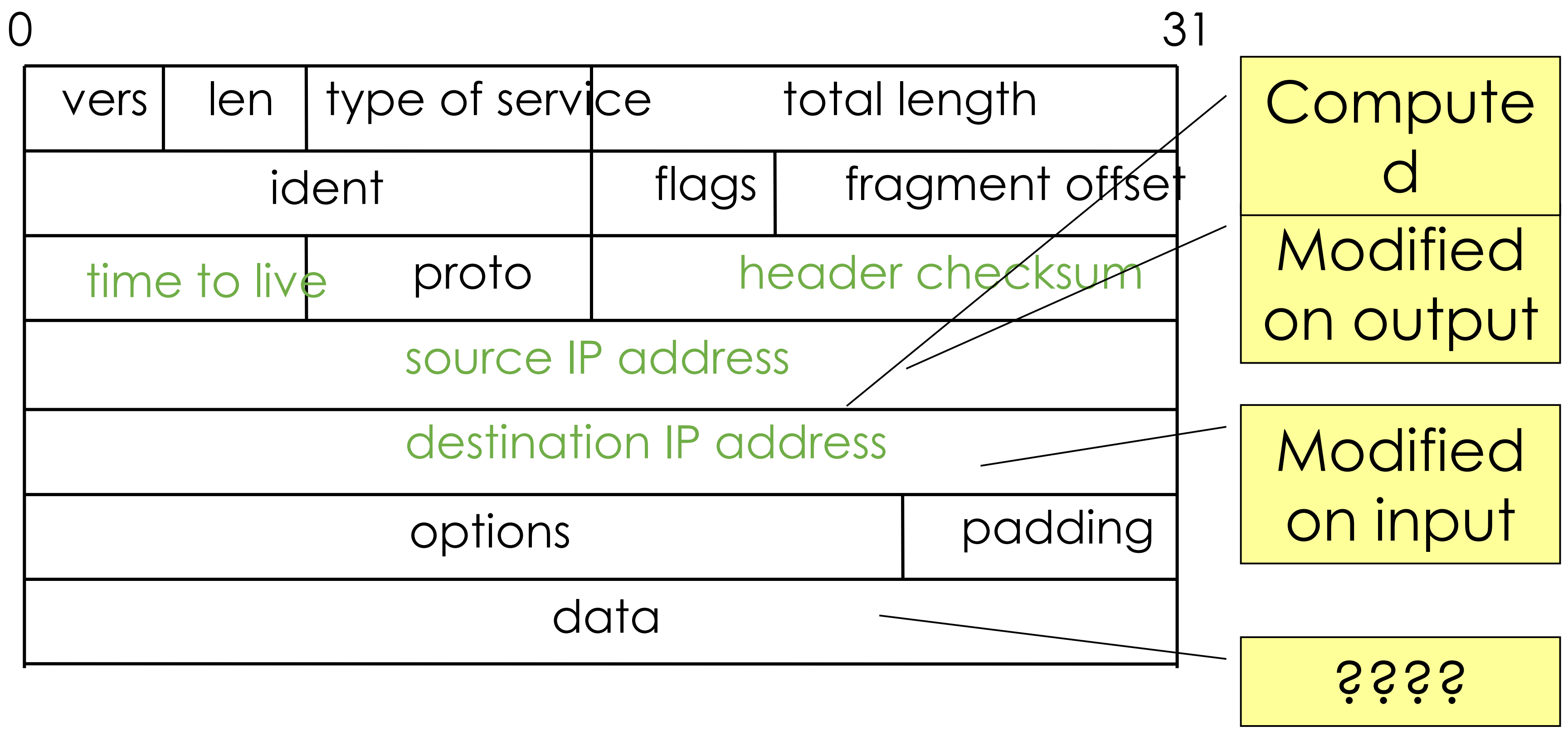
- Introduced in the early 90s to alleviate IPv4 address space congestion
- Relies on translating addresses in an internal network, to an external address that is used for communication to and from the outside world
- NAT is usually implemented by placing a router in between the internal private network and the public network.
- Saves IP address space since not every terminal needs a globally unique IP address, only an organizationally unique one
- While NAT should really be transparent to all high level services, this is sadly not true because a lot of high level communication uses things on IP

Translation

- Router has a pool of private addresses 192.168.10.0/24



IP Packet Modifications


0
31
vers
len
type of service
total length
ident
flags
fragment offset
time to live
proto
header checksum
source IP address
destination IP address
options
padding
data
Compute
d
Modified on output
Modified on input
????