CS 4910: Intro to Computer Security

Network Security I: Computer Network Concepts & Network Attacks

Instructor: Xi Tan

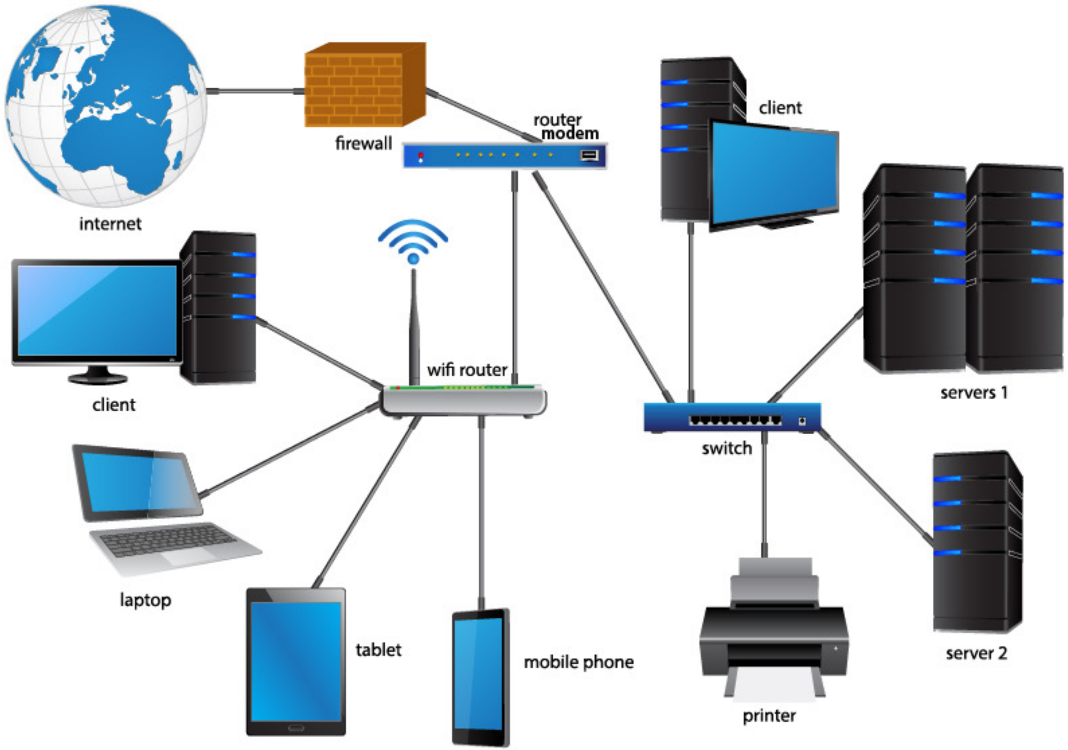
Updates

- Lab 2:
  - Task 1: Packet sniffing and spoofing
  - Task 2: Not required
  - Deadline: 3/31
- Homework 3
  - Deadline: ~~04/02~~ 04/07
- Research Paper:
  - Research Paper Topic Selection
  - Deadline: 03/19

Network Security

- Computer Network Concepts
- Network Attacks
- Network Security

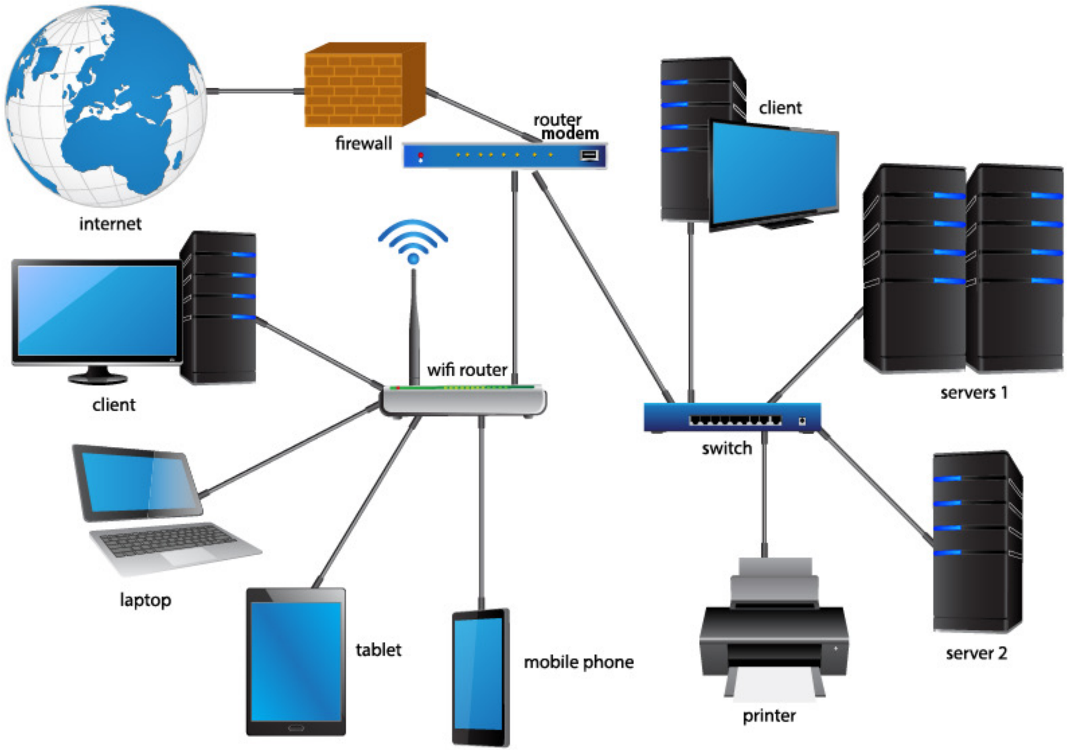
What is A Computer Network?



A computer network is a collection of computers and other devices connected together to communicate and share resources.

Figure is from: https://integrinetit.com/what-is-a-computer-network/

Important Components for A Network



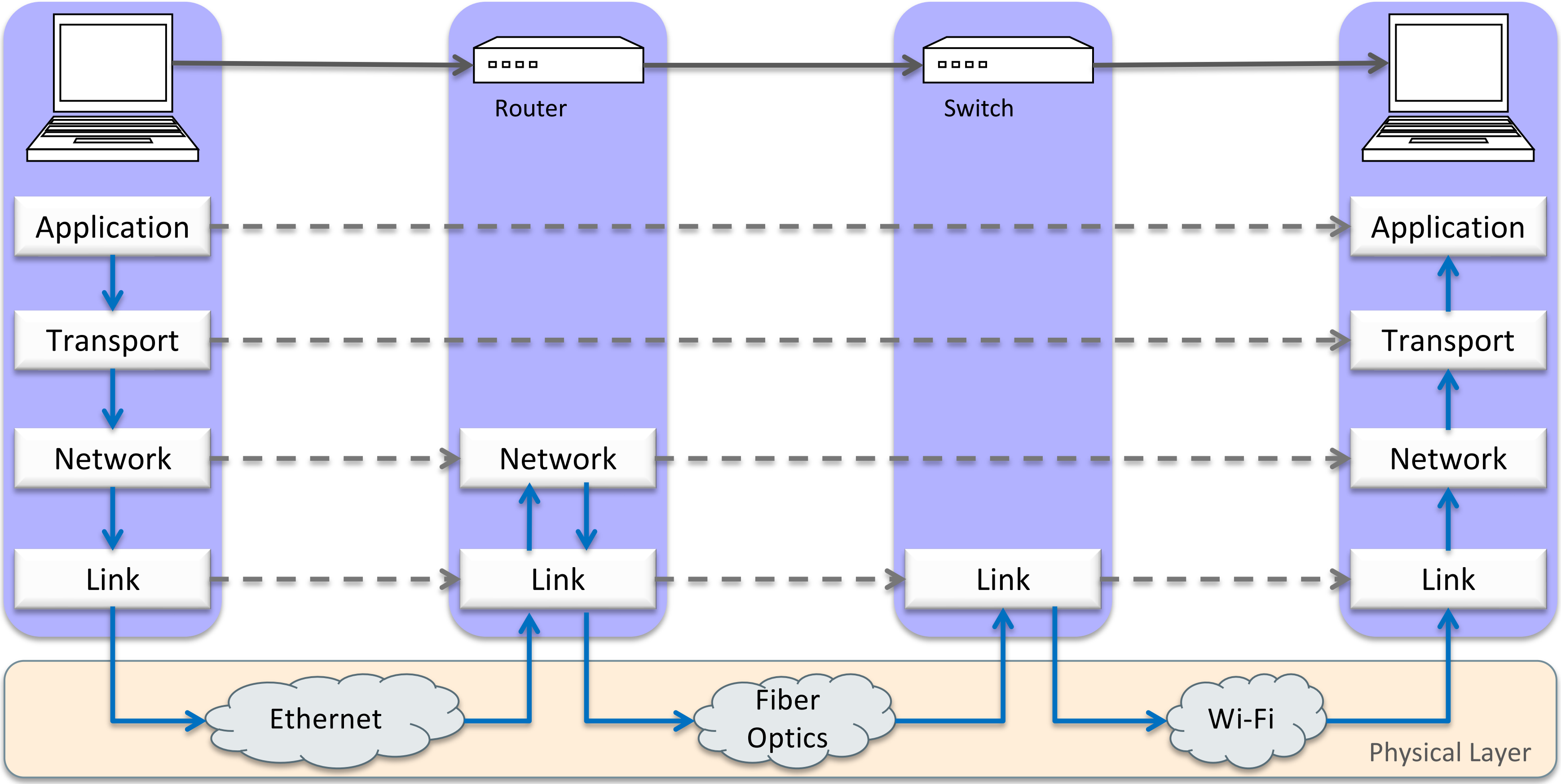
How are those devices connected?

How is the data (frame/packet) transmitted through the connection?

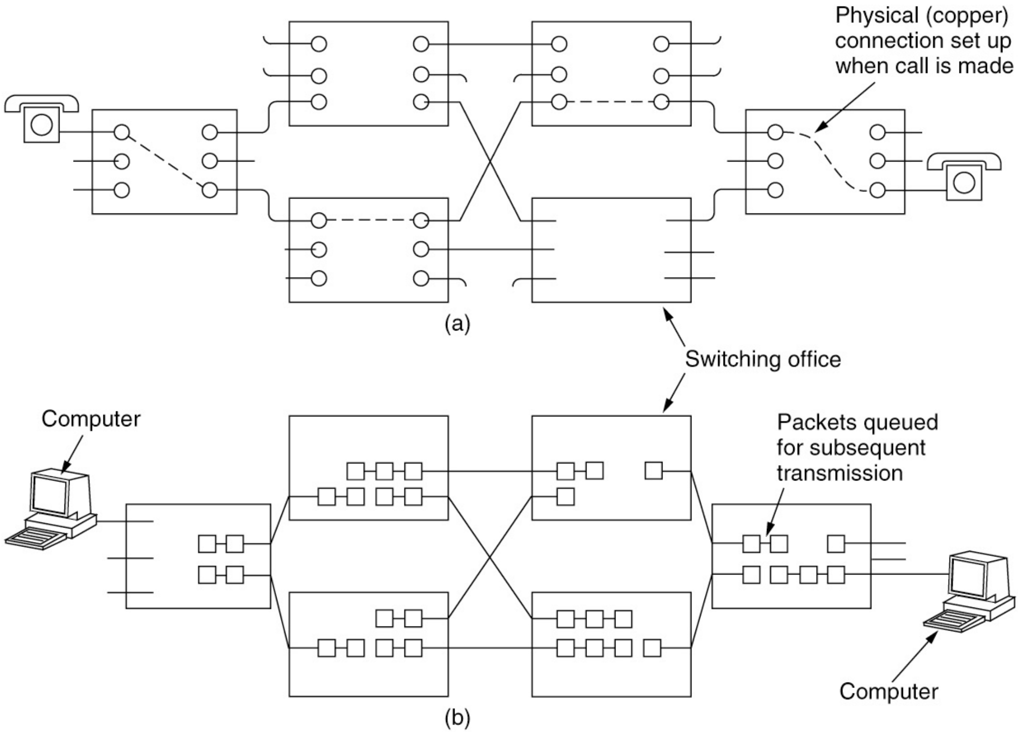
How do those devices interpret the data?

Figure is from: https://integrinetit.com/what-is-a-computer-network/

How are those devices connected? -- Network Layers



How is the data transmitted through the connection?



(a) Circuit switching (b) Packet switching

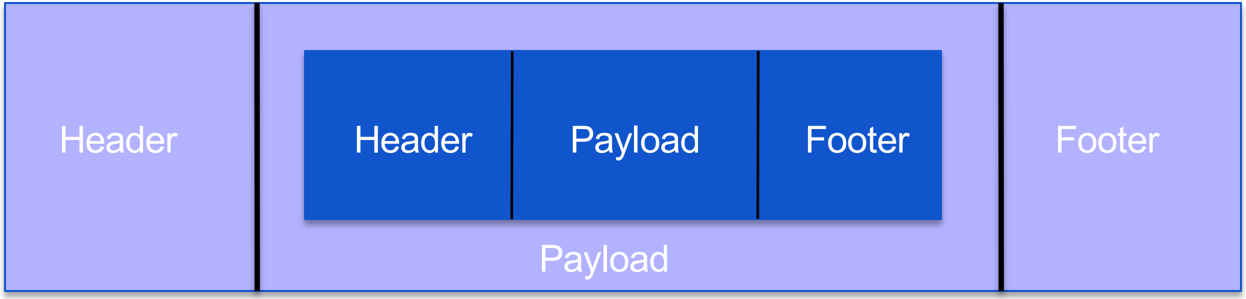
How do those devices interpret the data?

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



Internet Communication

Layers	Data Encapsulation (frame/packet)	Protocols
Application	HTTP Header · HTTP Data	DNS
Transport	TCP Header · HTTP Header · HTTP Data	UDP, TCP
Network	IP Header · TCP Header · HTTP Header · HTTP Data	IP, ICMP
Data Link	MAC Header · IP Header · TCP Header · HTTP Header · HTTP Data	MAC, ARP

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

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

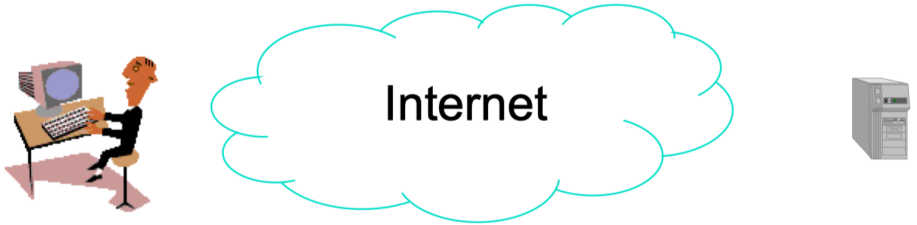
Packet Sniffers

- What can we get from packet sniffers?
  - Packet header
  - Payload data
  - Unencrypted sensitive data
  - Protocols in use
- Tools?
  - Wireshark, tcpdump, etc.

Detecting Sniffers

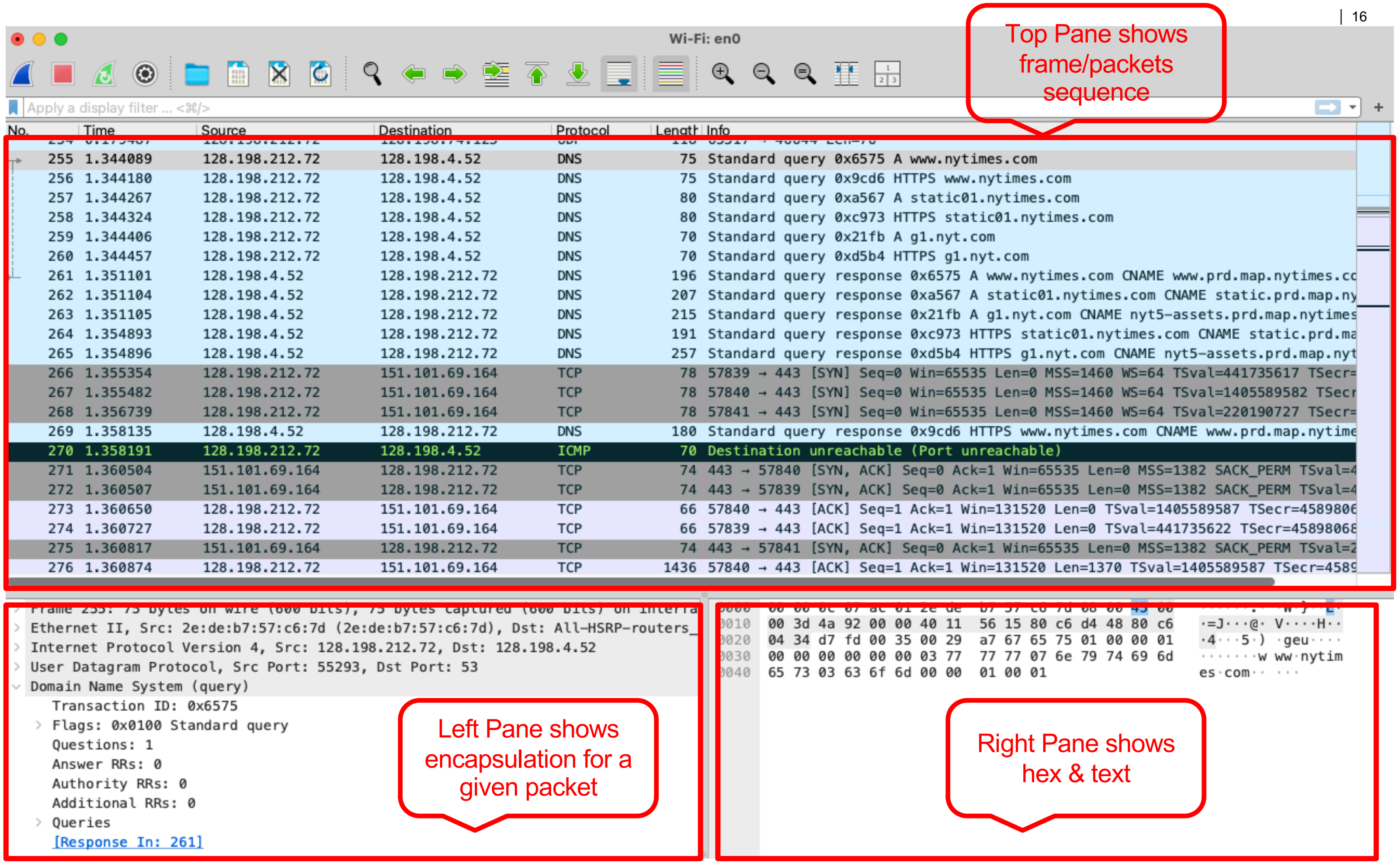
- Sniffers are almost always passive
  - They simply collect data
  - They do not attempt “entry” to “steal” data
- This can make them extremely hard to detect
- To reduce the impact of packet sniffing, encryption mechanisms should be utilized in higher-level protocols to prevent attackers from recovering sensitive data

Network Analyzer -- Wireshark

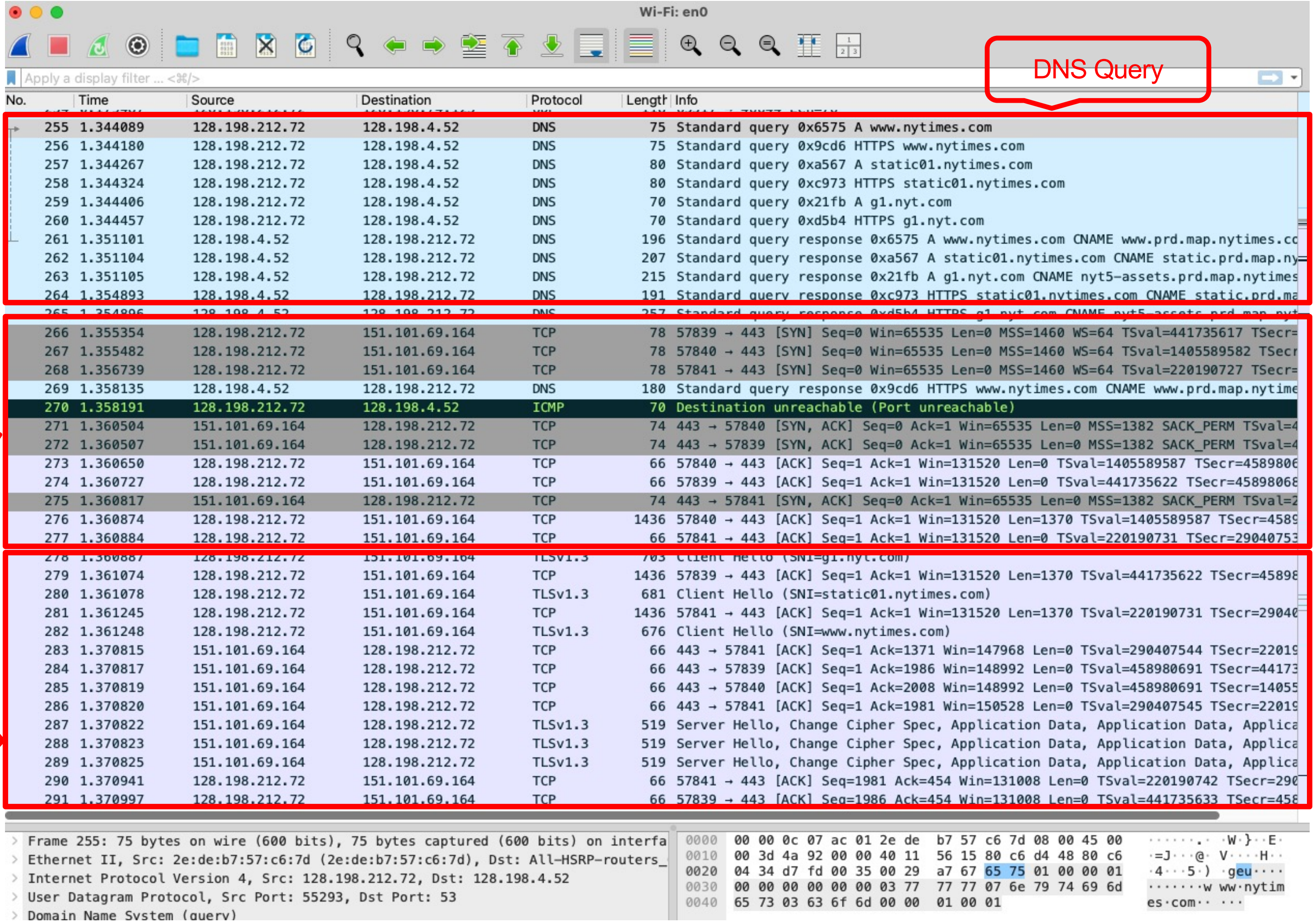


- User clicks on http://www.nytimes.com/
- Network analyzer captures all frames observed by its NIC
- Sequence of frames and contents of frame can be examined in detail down to individual bytes

Wireshark Windows



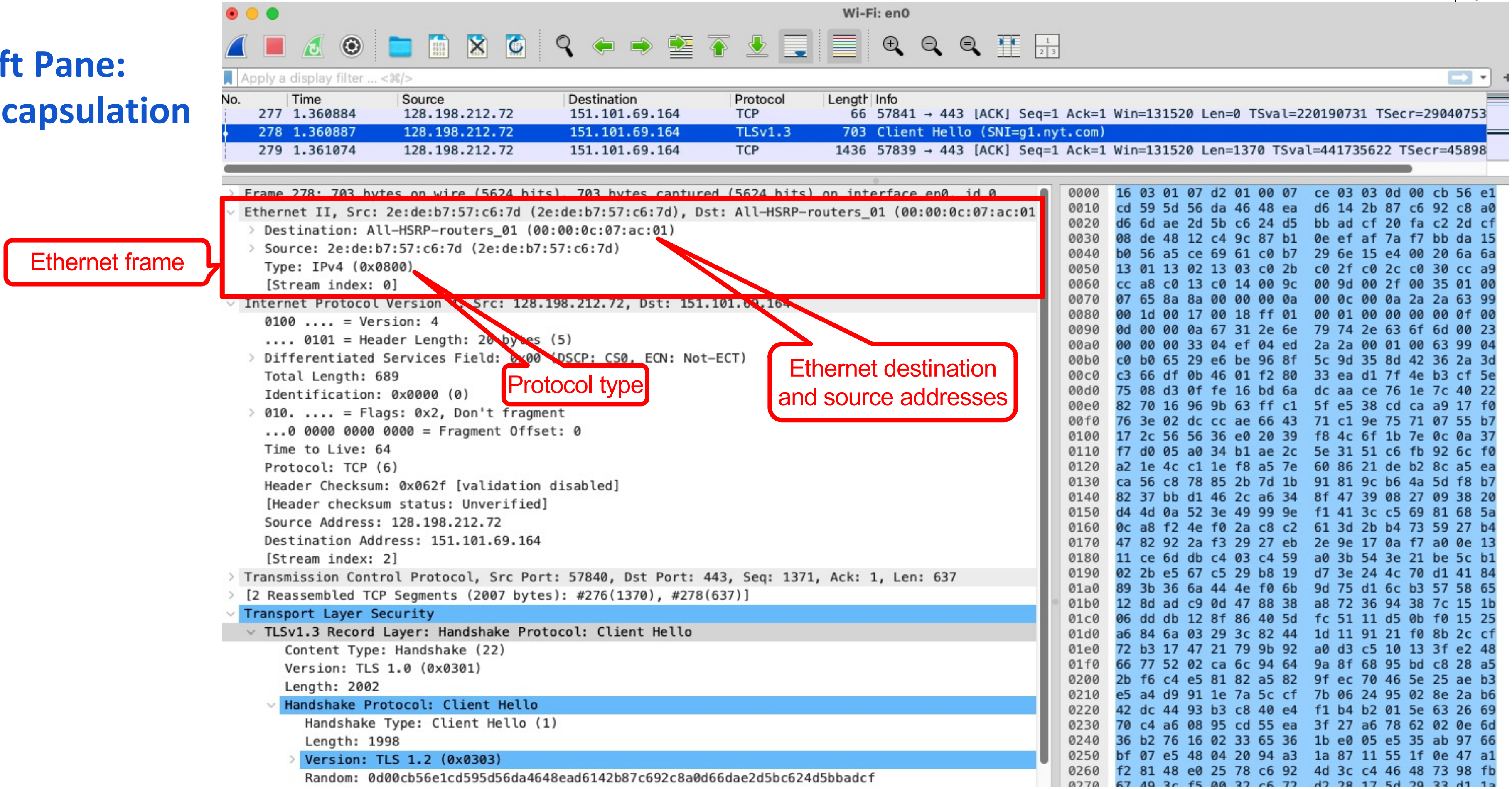
Top pane: Frame Sequence



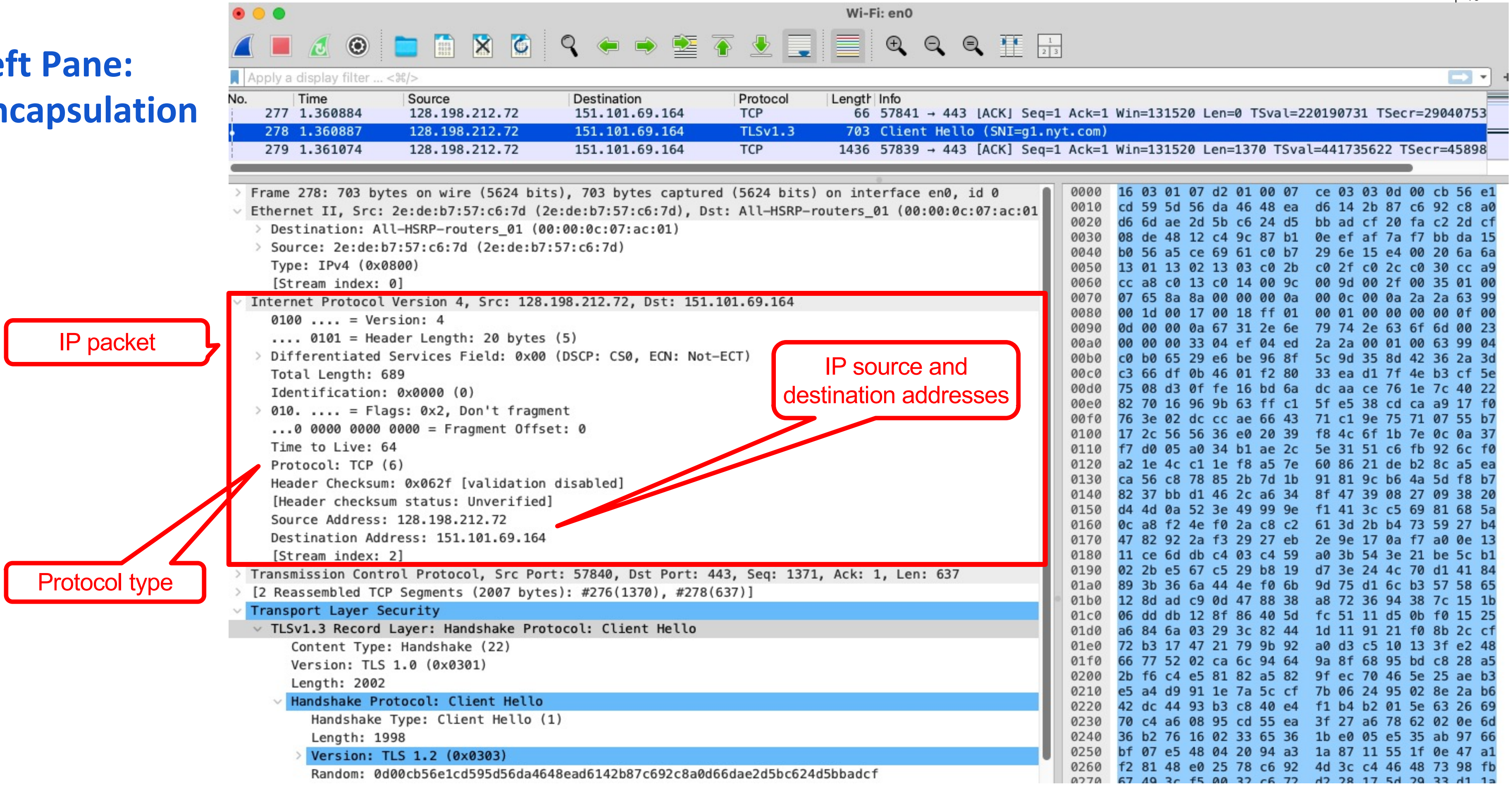
TCP connection establishment: SYN, SYN-ACK, ACK

TLS handshake: Client hello, server hello, key exchange and final handshake (Required when using HTTPS)

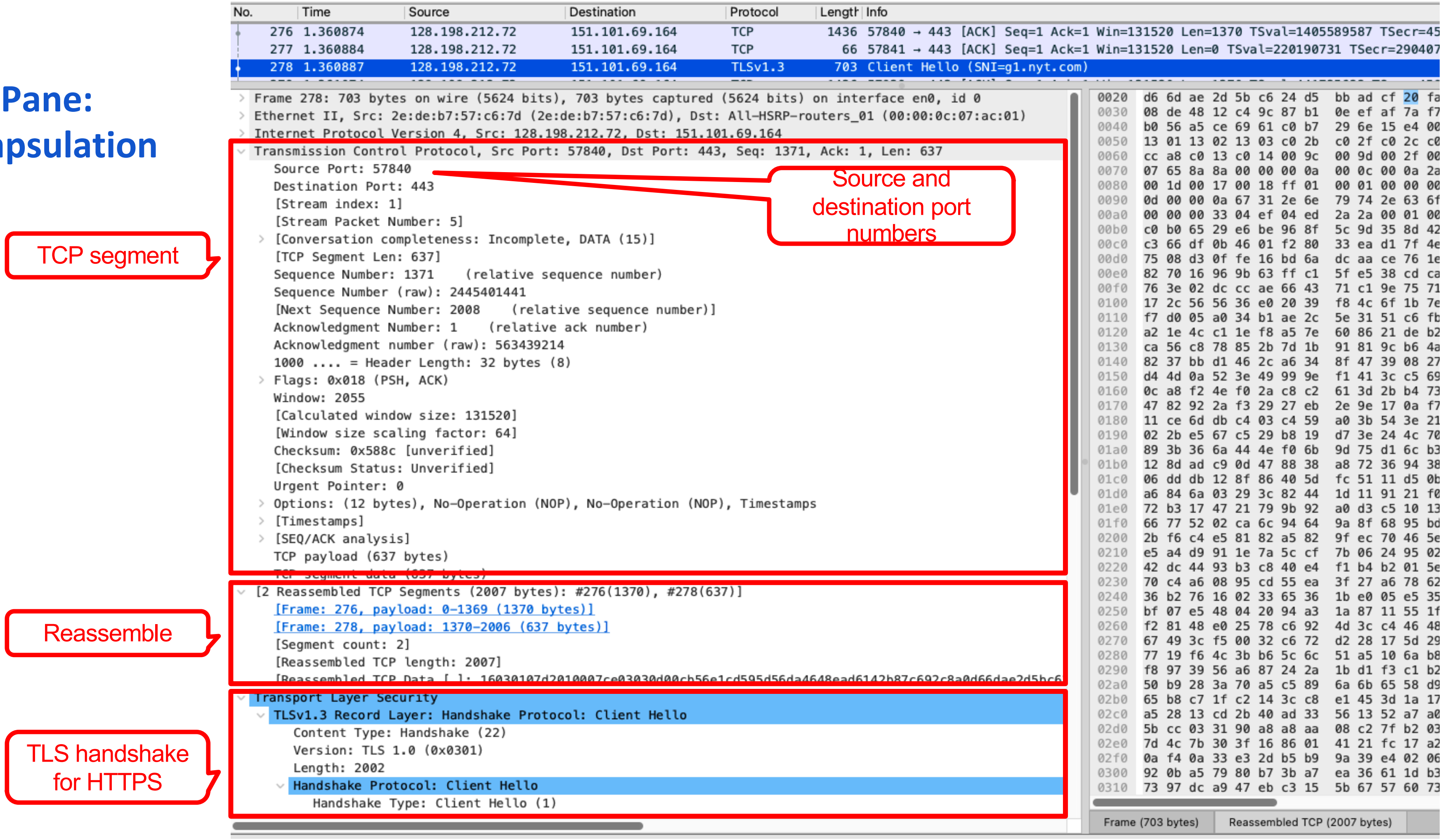
Left Pane: Encapsulation



Left Pane: Encapsulation



Left Pane: Encapsulation



Network Attacks

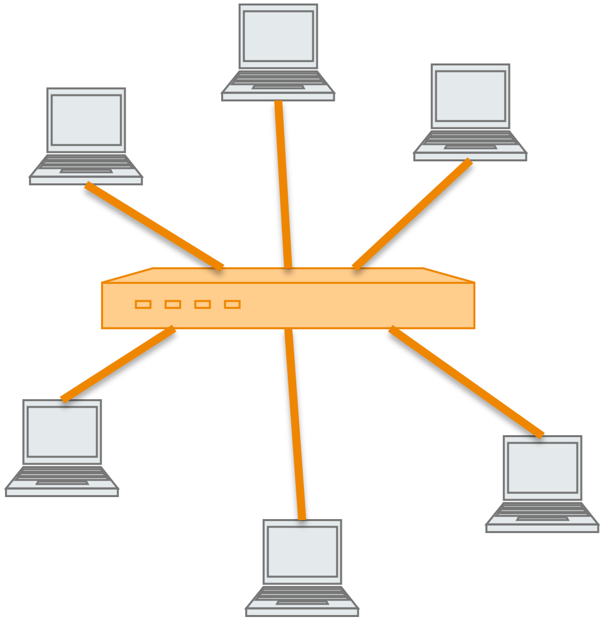
- MAC Spoofing, ARP Spoofing
- IP Spoofing
- Denial of Service
- DNS Cache Poisoning

MAC Addresses

- Most network interfaces come with a predefined MAC (Media Access Control) 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

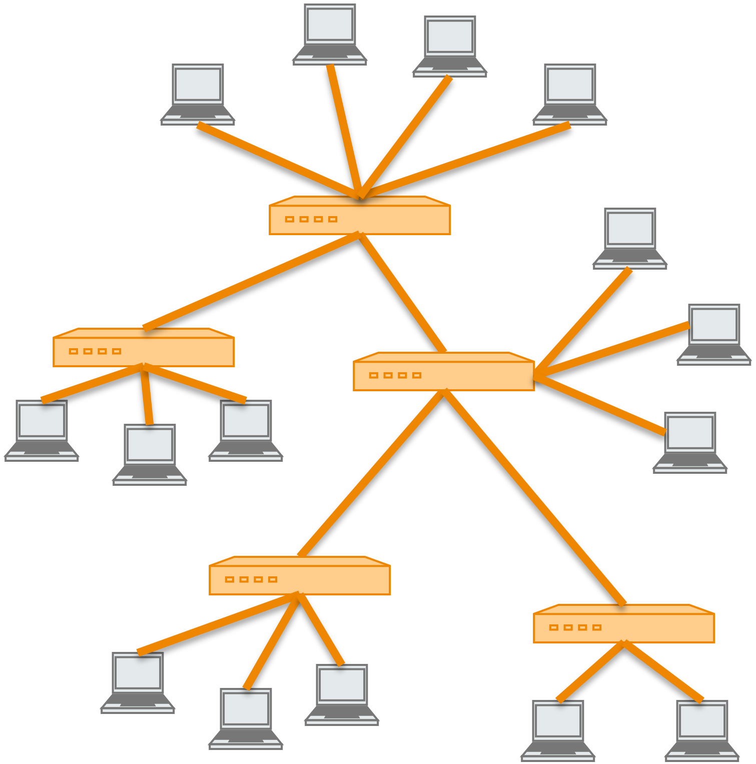
Switch

- A switch is a common network device
  - Operates at the link layer
  - Has multiple ports, each connected to a computer
  - Connects computers in an organization’s internal Lan (local area network)
- 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 Spoofing

- 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) is a link-layer protocol that connects the network layer to the link 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>
- Then the machine with <IP address1> responds the requestor with an ARP reply as

  <IP address1> is <MAC address>
- 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

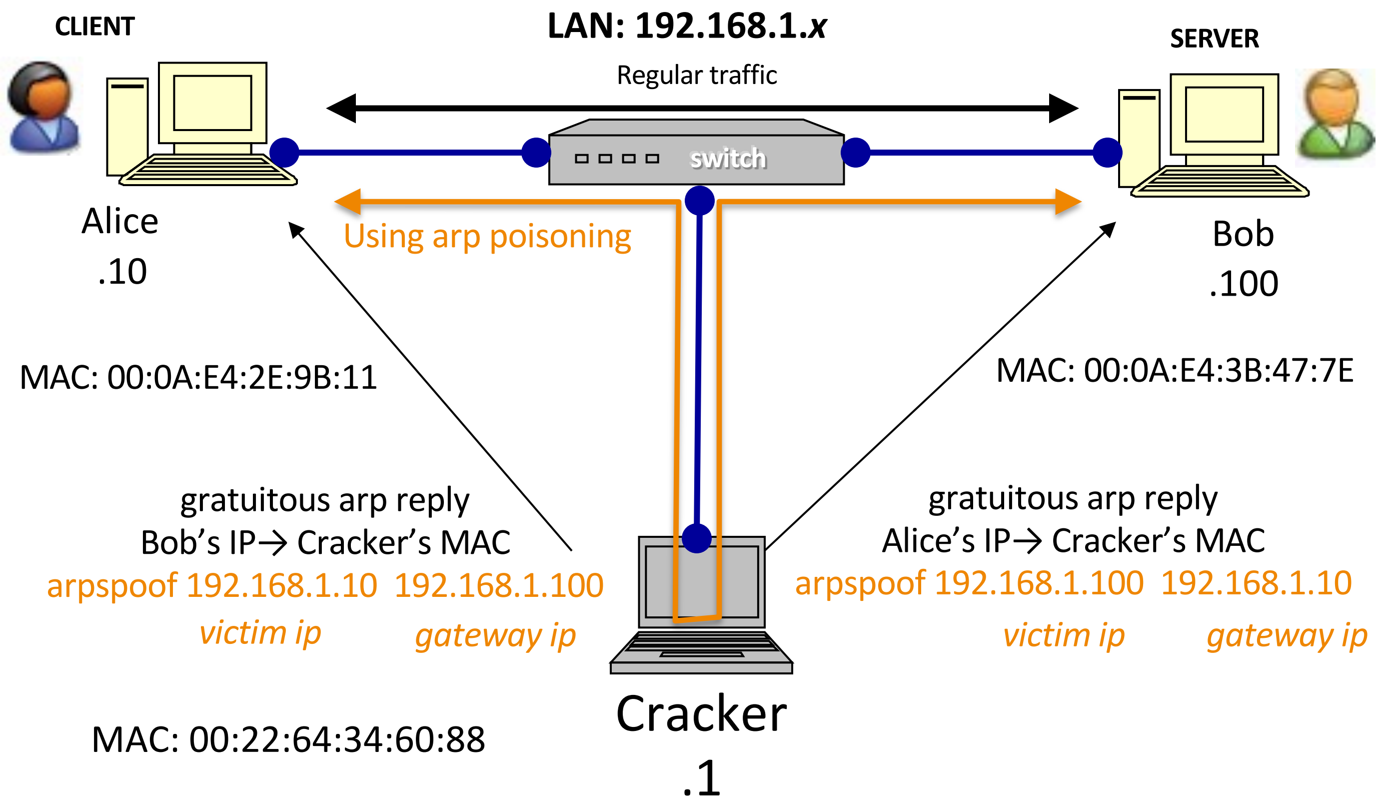
- The ARP table is updated whenever an ARP response is received
- Requests are not tracked
- ARP announcements are not authenticated
- Machines trust each other

- A rogue machine can spoof other machines

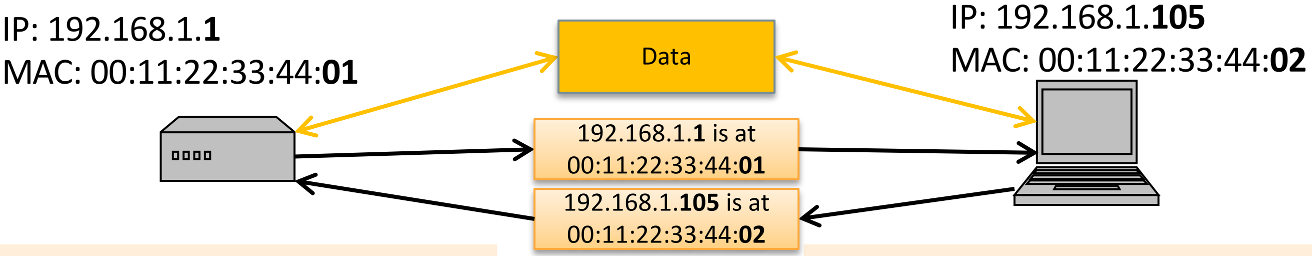
ARP Spoofing (ARP Poisoning)

- 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!

Example 1: ARP Spoofing



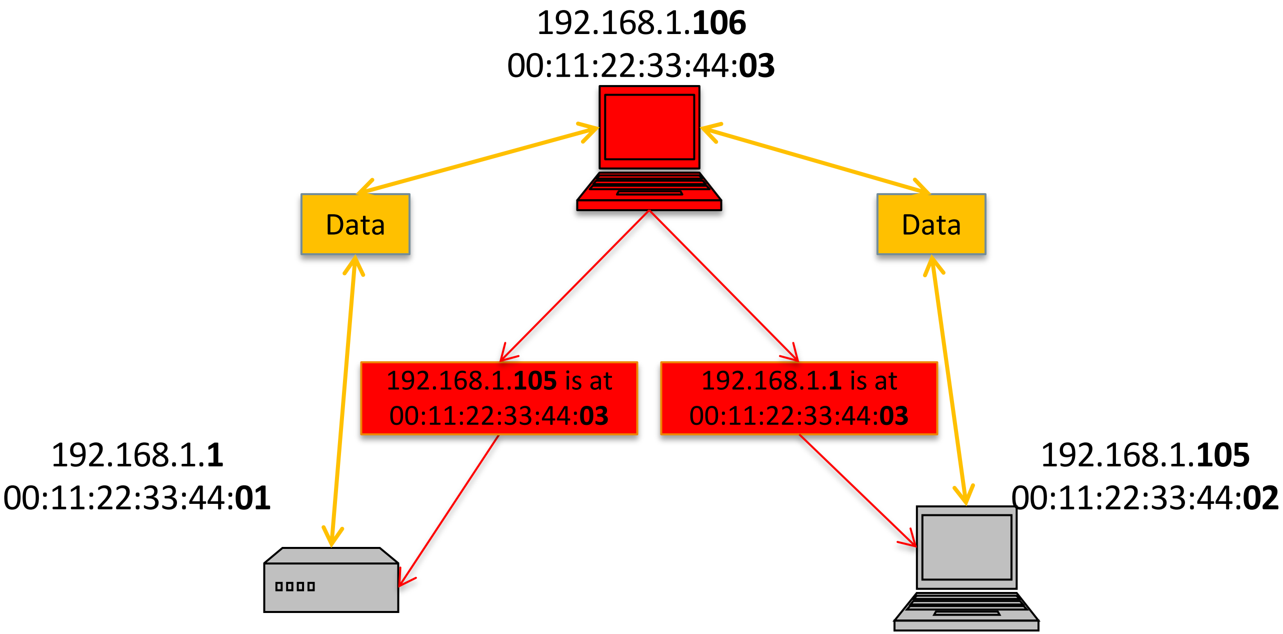
ARP Caches



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

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

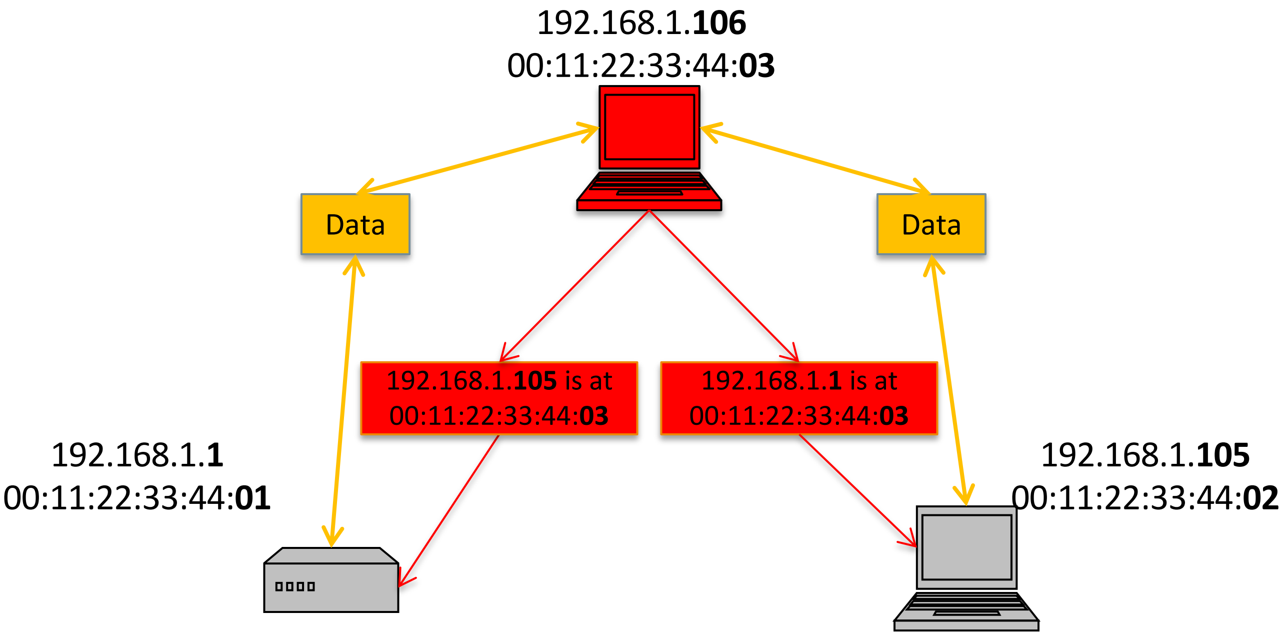
Example 2: Poisoned ARP Caches



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

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

Example 2: 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**

ARP Spoofing (or ARP poisoning)

- Send fake ARP messages to an Ethernet LAN (no authentication)
  - this causes other machines to associate IP addresses with attacker’s MAC

- Defenses
  - static ARP table
  - DHCP snooping (access control based on IP, MAC, and port)
  - detection: Arpwatch, reverse ARP

Next

Network Attacks

- MAC Spoofing, ARP Spoofing
- IP spoofing
- Denial of service
- DNS cache poisoning