CSE 127 Computer Security
Alex Gantman, Spring 2018, Lecture 16

Network Security I
Objectives

- Setting aside exploitation of vulnerabilities in packet parsing software, what protocol logic attacks are possible?
- Our goal is to understand:
  - Architecture of the Internet protocol suite (TCP/IP)
  - Common weaknesses in networking protocols
  - Available mitigations and their limitations
Review: Internet Protocol Suite

- **Application Layer**
  - Examples: SMTP, FTP, SSH, HTTP, etc.

- **Transport Layer**: Port-addressed host-to-host communications (on LAN or WAN).
  - User Datagram Protocol (UDP): single packet transmission with no reliability or ordering mechanisms.

- **Internet Layer (IP)**: Fragmentation, reassembly, and end-to-end (across network boundaries) routing of data packets.
  - Provides a uniform interface that hides the underlying network topology.

- **Link Layer**: Transmission of data frames within a local network (without intervening routers).
  - Example: Ethernet

- **Physical Layer**: Transmission of raw bits (rather than logical data packets) over a physical data link connecting network nodes.
  - Example: 100BASE-T
  - [Technically not part of the Internet Protocol Model, but is still there]
Review: Internet Protocol Suite

**Internet protocol suite**

**Application layer**
- BGP
- DHCP
- DNS
- FTP
- HTTP
- IMAP
- LDAP
- MGCP
- MQTT
- NNTP
- NTP
- POP
- ONC/RPC
- RTP
- RTSP
- RIP
- SIP
- SMTP
- SNMP
- SSH
- Telnet
- TLS/SSL
- XMPP
- more...

**Transport layer**
- TCP
- UDP
- DCCP
- SCTP
- RSVP
- more...

**Internet layer**
- IP (IPv4, IPv6)
- ICMP
- ICMPv6
- ECN
- IGMP
- IPsec
- more...

**Link layer**
- ARP
- NDP
- OSPF
- Tunnels (L2TP)
- PPP
- MAC (Ethernet, DSL, ISDN, FDDI)
- more...

---

https://en.wikipedia.org/wiki/Internet_protocol_suite
Review: Internet Protocol Suite

https://en.wikipedia.org/wiki/Internet_protocol_suite
TCP/IP Protocol Stack by Example

- ROUGHLY, what happens when I click on a URL while UCSD’s network?
Application Layer (HTTP)

- Turn click into HTTP GET request
Application Layer (Name Resolution)

- Where is www.yahoo.com?

What’s the address for www.yahoo.com

Oh, you can find it at 64.58.76.177
Transport Layer (TCP)

- Break message into packets (TCP segments)
- Should be delivered reliably & in-order

```plaintext
GET http://www.yahoo.com/r/mp HTTP/1.1
Host: www.yahoo.com
Connection:keep-alive
...
```

“and let me know when they got there”
Network layer: IP Addressing

- Address each packet so it can traverse network and arrive at host
- Addresses are generally globally unique

My computer: 132.239.9.64

www.yahoo.com: 64.58.76.177

Destination: 64.58.76.177
Source: 132.239.9.64
Data: GET http://www.yahoo.com
Network Layer: IP Routing

- Each router forwards packet towards destination
Datalink layer (Ethernet)

- Break message into frames
- Media Access Control (MAC)
  - Can I send now? Can I send now?
- Send frame
TCP/IP Security (1970’s)

▪ Original TCP/IP design: Trusted network and hosts
  – Administered by mutually trusted parties

▪ End-to-end Principle
  – Intelligence is at the edges
  – Network is simple
    ▪ Optimized for speed and simplicity, maintains no state

▪ Robustness Principle
  – “In general, an implementation must be conservative in its sending behavior, and liberal in its receiving behavior. That is, it must be careful to send well-formed datagrams, but must accept any datagram that it can interpret (e.g., not object to technical errors where the meaning is still clear).”
  ▪ https://www.ietf.org/rfc/rfc0791.txt

https://en.wikipedia.org/wiki/ARPANET
TCP/IP Security (1980’s)

- Wait ... what if we can’t trust everyone?
  - “When describing such attacks, our basic assumption is that the attacker has more or less complete control over some machine connected to the Internet. This may be due to flaws in that machine’s own protection mechanisms, or it may be because that machine is a microcomputer, and inherently unprotected. Indeed, the attacker may even be a rogue system administrator.”

- Can’t trust the hosts
  - Compromised hosts
  - Untrusted insiders on internal networks
  - Anyone can connect to public Internet [next decade]

- But network is still trusted
TCP/IP Security (today)

- Can’t trust the network either
  - Network equipment can be compromised
  - Untrusted network operators
  - Anyone can access the physical channel of wireless networks
TCP/IP Security

- Built-in trust assumptions:
  - Hosts controlled by trusted administrators
    - Random people can't get onto the network
    - Correct information reported by hosts
    - Protocols implemented correctly
  - Network protocols used only as intended
    - Correct packet headers
    - Consideration of others’ resources
      - Rate limiting of costly operations
Attacker Models

- Man in the middle: can see, block, and modify traffic
  - Attacker controls wifi access point

- Passive: Eavesdrop on traffic
  - Attacker has passive tap or recorded traces

- Off-path: attacker can inject traffic into network
  - Anyone with access to network
No Confidentiality

- Who can see the packets you send?
  - Network (routers, switches, access points, etc.)
  - Unprotected WiFi network: everyone within range
  - WPA2 Personal (PSK): everyone on same network
  - Non-switched Ethernet: everyone on same network
  - Switched Ethernet: maybe everyone on same network
No Authentication

- TCP/IP offers no authentication of packets
  - Source address in IP header set by sender

- Attacker with direct access to network (including MitM) can spoof source address
  - Spoof: forge, set to arbitrary value

- Connectionless protocols (UDP) especially vulnerable
Link Layer (MAC)

- Physical channel is often shared by multiple hosts on the local network.
  - Examples: open WiFi, non-switched Ethernet

- Link layer controls access to the physical medium.
  - Also known as the Media Access Control (MAC) layer.
  - Acronym collision

- How to make sure each host only gets frames addressed to it?

- Each host is responsible for picking up frames addressed to it and ignoring the others.
  - Honor system!

- Filtering typically happens on the network card (or equivalent).
  - Only frames addressed to this host are parsed and passed on to the layer above.

- Many support “promiscuous” mode – all frames are picked up.
Network Routing

- Say I want to send packet to 8.8.8.8 ...

- Step 1: Is host on local network?
  - Check subnet masks of local networks
Status: Connected

Ethernet 1 is currently active and has the IP address 132.239.17.19.

Configure IPv4: Manually

IP Address: 132.239.17.19
Subnet Mask: 255.255.255.0
Router: 132.239.17.1
DNS Server: 132.239.0.252, 128.54.16.2, 8.8.8.8
Search Domains: ucsd.edu

802.1X: WPA: UCSD-P...
Dynamic Host Configuration Protocol (DHCP)

- How does a computer know it’s own IP address?
  - Or its subnet mask?
  - Or the IP address of its default gateway?

- Dynamic Host Configuration Protocol (DHCP)

- Protocol sketch:
  - Alice joins the local network and would like an IP address
  - She broadcasts a DHCP Discovery message: “I would like an IP address, please”
  - Router(s) reply with DHCP Offers: “You can use x.x.x.x for a day”
  - Alice picks one of the offers and broadcasts a DHCP Request: “I’ll take x.x.x.x”
  - Router responds with DHCP Acknowledgement “Ok, x.x.x.x is yours for a day”

- Offer and Acknowledgement also contain subnet, router, and DNS addresses
Status: **Connected**

Ethernet 1 is currently active and has the IP address 132.239.17.19.

**Configure IPv4:** Manually

**IP Address:** 132.239.17.19

**Subnet Mask:** 255.255.255.0

**Router:** 132.239.17.1

**DNS Server:** 132.239.0.252, 128.54.16.2, 8.8.8.8

**Search Domains:** ucsd.edu

**802.1X:** WPA: UCSD-P...
Network Routing

- Say I want to send packet to 8.8.8.8 ...

- Step 1: Is host on local network?
  - Local: send directly
  - Not local: send via default gateway
Status: **Connected**

Ethernet 1 is currently active and has the IP address 132.239.17.19.

- **Configure IPv4:** Manually
- **IP Address:** 132.239.17.19
- **Subnet Mask:** 255.255.255.0
- **Router:** 132.239.17.1
- **DNS Server:** 132.239.0.252, 128.54.16.2, 8.8.8.8
- **Search Domains:** ucsd.edu

**802.1X:** WPA: UCSD-
Network Routing

- Say I want to send packet to 8.8.8.8 ...
- Step 1: Is host on local network?
  - Local: send directly
  - Not local: send via default gateway
- Step 2: Create IP packet
- Step 3: Create and send link layer (e.g. Ethernet) frame
Network Routing

- Ethernet frame:
  - Host needs to fill in Ethernet destination address
    - MAC address of host on local network
    - MAC address of gateway for host not on local network
  - How to find Ethernet address from an IP address?
Address Resolution Protocol (ARP)

- Address Resolution Protocol (ARP)
  - used to query hosts on local network to get link-layer address for an IP address

- Protocol sketch
  - Alice (looking for Bob’s IP) broadcasts an ARP request:
    - “What is the MAC address of 10.0.0.3?”
  - Bob sees broadcast and replies:
    - “The MAC address of 10.0.0.3 is 01:02:03:04:05:06.”
  - Alice sends IP packet for 10.0.0.3 in an Ethernet frame to 01:02:03:04:05:06.
Address Resolution Protocol (ARP)

Looking for physical address of a node with IP address 141.23.56.23

The MAC address of destination is broadcast address: 0xFFFFFFFFFFFFFF

The node physical address is A46EF45983AB

IP = 141.23.56.23

a. ARP request is broadcast
b. ARP reply is unicast
Address Resolution Protocol (ARP)

- ARP messages are Ethernet frames
- ARP requests are broadcast (on the local subnet)
- Anyone can send an ARP reply
ARP Spoofing

- Attacker on the network can impersonate any other host

- **Mitigation**
  - Fixed ARP tables
    - Impractical for all but small fixed networks
  - Port binding on switch
    - Restrict MAC and IP addresses allowed on a physical port switch
  - Higher level host authentication
    - E.g. SSH or TLS
Problems With Addressing

- This problem repeats at every protocol layer:
  - Source needs to send something to destination.
  - How to know which address corresponds to name?
    - Domain name to IP address
    - IP routing
    - IP address to Ethernet address
    - ...
Network Routing

- Say I want to send packet to 8.8.8.8 ...

- Step 1: Is host on local network?
  - Local: send directly
  - Not local: send via default gateway

- Step 2: Create IP packet

- Step 3: Create and send link layer (e.g. Ethernet) frame

- Step 4: Gateway picks next router in path and forwards the IP packet
  - Repeat until destination is reached
  - How to know which router to forward to next?
Border Gateway Protocol (BGP)

- Border Gateway Protocol (BGP) is used to manage IP routing information between networks on the Internet.
- Each BGP node maintains connections to a set of trusted neighbors.
  - Connections between neighbors may be (weakly) authenticated.
- Neighbors share routing information.
- No authorization.
  - Malicious (or malfunctioning) BGP nodes may provide incorrect routing information that redirects IP traffic.
BGP Hijacking

- 2008 Pakistan tried to block YouTube within the country
  - Pakistan Telecom claimed ownership of YouTube’s IP block via BGP
  - BGP nodes forwarded this routing information
  - YouTube sinkholed globally

- 2018 MyEtherWallet.com compromised, $100,000’s reported stolen
  - Attackers used BGP hijacking to claim ownership of a chunk of Amazon Route 53 (DNS) addresses
  - Used hijacked DNS traffic to direct MyEtherWallet.com-bound traffic to attackers’ servers in Russia

- BGP errors (accidental or malicious) are constantly being made
  - See [https://twitter.com/bgpstream](https://twitter.com/bgpstream) or [https://bgpstream.com/](https://bgpstream.com/)
IP Spoofing Attacks

- There is no authentication in Link or Internet layers

- Even if routing is correct, Eve can still spoof Alice’s IP address
  - Eve can send IP packets claiming to be from Alice
  - Eve may not be able to receive IP packets addressed to Alice

- UDP: trivial
  - Stateless protocol, each datagram is independent of others

- TCP: more complicated, but still possible
  - Two endpoints maintain a shared state
  - Attacker must be able to guess it
TCP Primer

- TCP provides reliable, ordered delivery of bytes
- Establishes a stateful session between two IP:port endpoints
- Each side maintains:
  - Sending sequence number: sending base + count of bytes sent
  - Receiving sequence number: receiving base + count of bytes received
TCP Primer

- Session established with a three-way handshake
  - x: client sequence base
  - y: server sequence base

TCP Connection Spoofing

- Eve wants to connect to Bob, pretending that she is Alice (spoofing Alice’s IP address).
- Eve can’t see traffic between Alice and Bob
  - “TCP off-path attack”
- Eve needs to complete the TCP three-way handshake without seeing the middle message
  - Must guess $y$
TCP Connection Spoofing

- The sequence number field is 32 bits

- Early implementations just incremented a global counter used to initialize sequence numbers for TCP connections
  - RFC 793 requires counter incrementing every 4 µs (250 kHz)
  - Early BSD kernels incremented by a large constant every second

- Later pseudo-random number generators were used
  - PRNGs were still global, weaknesses allowed guessing
TCP Connection Spoofing

- In 2007 Comcast was injecting forged TCP reset messages to interfere with customers’ peer-to-peer file sharing traffic
  - FCC ordered it to stop
TCP Primer

- Note: Server needs to maintain state (x and y) after receiving initial SYN
TCP SYN Flood

- Denial of Service (DoS) attack.
- Attacker “floods” the server with SYN messages.
  - Never responds to SYN-ACK with ACK
  - Source IP usually spoofed
- Each received SYN uses up server resources to store state.
- Eventually all resources are exhausted.
- All new connection attempts are ignored.

TCP SYN Flood Mitigations

- More server resources
  - Larger connection queues
- Discarding unfinished handshakes
- SYN Cookies
  - Encoding state information in the ACK sequence number
IPSec

- Internet layer alternative to IP that provides encryption and authentication

- Two modes
  - Transport Mode
    - Payload protected, but headers are in the clear
  - Tunnel Mode
    - Entire IP packet is protected and encapsulated inside another IP packet for routing
    - Used for VPNs
Onion Routing

- Anonymous communication via a network of proxies
- Communication between endpoints is encapsulated in layers of encryption
  - Get it? Like an onion.
- Each node only knows identity of next node
  - Peels of (or puts on) its layer of encryption and passes the message on to the next node
Onion Routing: Tor
Additional Resources

- **Wireshark**
  - [https://www.wireshark.org/](https://www.wireshark.org/)

- **Attacking Network Protocols**
  - By James Forshaw
  - [https://nostarch.com/networkprotocols](https://nostarch.com/networkprotocols)

- **Tor Protocol Specification**
  - by Roger Dingledine and Nick Mathewson
  - [https://gitweb.torproject.org/torspec.git/tree/tor-spec.txt](https://gitweb.torproject.org/torspec.git/tree/tor-spec.txt)
    - and rest of [https://gitweb.torproject.org/torspec.git/tree/](https://gitweb.torproject.org/torspec.git/tree/)
Review

- TCP/IP protocol stack was not designed for modern threat models
- Ethernet, IP, UDP, TCP, ARP, BGP, DHCP do not protect integrity or confidentiality
- Mapping names to addresses is a persistent problem in protocols
- Must rely on application-layer security mechanisms
Homework

- Assignment 5 is due next week (6/4 @ 10pm)
- Read Chapter 5 from *The Craft of System Security*
Next Lecture...

Network Security II