CompTIA Security+ study notes

SY0-401

Exam breakdown

Knowledge domain	Percent of exam
Network security	20%
Compliance and operational security	18%
Threats and vulnerabilities	20%
Application, data and host security	15%
Access control and identity management	15%
Cryptography	12%

Index

- Secure shell (SSH)
- Domain name server (DNS)
- Transport layer security (TLS)
- Transmission control protocol (TCP/IP)
- File transfer protocol (FTP)
- Hypertext transfer protocol (HTTP)
- Secure file copy (SCP)
- IPv4 and IPv6
- Fibre channel
- Internet small computer system interface (iSCSI)
- Telnet
- NetBIOS

- Secure wireless networking
  - Service set identifier (SSID)
  - Wired Equivalent Privacy (WEP)
  - Wi-fi protected access (WPA)
  - Wi-fi protected access 2 (WPA2)
  - Wi-fi protected setup (WPS)
  - Extensible authentication protocol (EAP)
  - Protected extensible authentication protocol (PEAP)
  - Lightweight extensible authentication protocol (LEAP)
  - CCMP
  - Wireless operations
    - MAC filter
    - Antenna placement
    - Power level controls
    - Antenna types
    - Captive portal

- Compliance and operational security
  - Risk concepts
    - Risk
    - Risk management
    - Risk assessment
    - Asset
    - Mitigation
    - Control types
    - False positives and negatives

- Security program
- Privacy policy
  - Safe harbour
- Information classification and handling
  - Data labeling, handling and disposal
- Acceptable use policy (AUP)
- Security policy
- Qualitative risk assessment
- Quantitative risk assessment
- Risk calculation
  - Single loss expectancy (SLE)
  - Annualized loss expectancy (ALE)
  - Impact
  - Availability
  - Recovery time objective (RTO)
  - Recovery point objective (RPO)
- Quantitative vs. qualitative risk
- Vulnerabilities
- Threat vectors
- Cloud and virtualization risk management

System integration processes

- Onboarding and offboarding business partners
- Interoperability agreements

Risk management

- Change management
- Incident management
- Routine audits
- Data loss or theft
- Business continuity concepts
  - Continuity of operations
  - Disaster recovery planning
- High availability
  - Redundant array of independent disks (RAID)
  - Clustering and load balancing
- Disaster recovery concepts
  - Backup plans and policies
  - Backup strategies

Network security

Network device configuration

Firewall

- Allow and block network traffic based on security policies
- Can control network traffic by machine, port and application-level details (principle of least access applies)
- Flood guard against DoS/DDoS attacks

How firewalls enforce security policies

- **Network address translation (NAT)** - masks internal network addresses
- Basic packet filtering - fast and efficient
- Stateful packet filtering - smarter but uses more resources
- Access control lists (ACLs) - check source address against rules before allowing (e.g. DMZ)
- Application layer proxies - forward packets to application for processing (e.g. SMTP)

Router

- Operate at the network layer (OSI layer 3)

- Connect different network segments together
- Physical or remote access is a major security issue:
    - Simple Network Management Protocol (SNMP) and Telnet both send passwords in the clear
    - Out-of-band remote management via separate NIC is more secure

Switch

- Operates at the data link layer (OSI layer 2) but some have routing (OSI layer 3)
- Each port has a separate collision domain
- Sniffers can't see as much compared to hub-based networks
- Enforces security policies by inspecting packets (MAC addresses)
- Security issues:
    - Commonly administered using SNMP and Telnet (cleartext passwords) and ships with a default password

Exam tip: SNMP v1 and v2 authenticate using cleartext passwords. SNMPv3 adds cryptographic protections. Switches should be secured by disabling all access protocols other than a secure serial line or SSH.

Load balancer

- Uses health checks and schedulers to spread work across servers
- Best used in stateless systems where subsequent requests can be handled by any server (e.g. websites)

Proxy

- Takes requests from the client and forwards to destination on behalf of the client
- Transparent proxies are called *gateways* or *tunneling proxies*

Proxy types

- Anonymizing proxies - Tor
- Caching proxy - reduces bandwidth and improves performance
- Content-filtering proxy - checks against Acceptable Use Policy (e.g. URLs)
- Open proxy - similar to anonymizing proxy, often used to circumvent corporate proxies
- Reverse proxy - sits in front of web servers, filters traffic, caches graphics, load balancing (e.g. nginx)
- Web proxy - specialized to handle web traffic, aka *web cache*

Web security gateways

- Combined proxy and content-filtering

Capabilities

- Real-time malware protection - scan all incoming and outgoing web traffic
- Content monitoring - check against Acceptable Use Policy
- Productivity monitoring - how much and how often certain resources are accessed
- Data protection and compliance - protect against exfiltration

VPN concentrators

- VPNs terminate at a specific point in the network, the *concentrator*
- VPN can encrypt data in a packet, or encrypt the entire packet and send it across the internet in a tunnel
- IPSec is the most common protocol

Intrusion detection systems (IDS)

- Detect, log and respond to unauthorized network or host use
- Usually software, but sometimes hardware appliances
- Two primary methods: signature-based and anomaly based
- Should be tuned based on org needs / threat model

Types

- Host-based (HIDS) - installed on a device, can't see network activity
- Network-based (NIDS) - sniffs the network, can't see device activity

Components

- Traffic collector/sensor - collects events for the IDS to examine
- Analysis engine - compares traffic sample to known suspicious patterns
- Signature database - collection of malicious patterns
- User interface and reporting - alerts dashboard, interaction with IDS

Intrusion detection models

Behaviour based

- Relies on a configured "normal" baseline and watches for suspicious patterns

- Could potentially detect 0days (not in signature database) but also has many false positives

Signature based

- Relies on predefined set of patterns (signatures)
- Fast and precise, but requires accurate and timely signature database

Anomaly based

- Similar to behaviour based, IDS goes through a training phase when first installed to develop a "normal" profile, then looks for deviations

Heuristic

- Uses AI (machine learning) algorithms to determine if a traffic pattern is malicious
- Extrapolates based on rules (e.g. if 10 attempts is bad, 20 is worse) - fuzzy logic

Intrusion prevention system (IPS)

- Like an IDS, but can automatically respond without operator intervention

Protocol analyzers

- Also known as a *packet sniffer*, *network analyzer* or *network sniffer* (e.g. Wireshark)
- Need to be able to place a NIC in *promiscuous mode* (process all network packets, regardless of destination)
- Promiscuous mode was easier with hub-models, for switches need to use *port mirroring* or *SPAN ports* which see all traffic
- *Cyclic redundancy check (CRC)* is used to determine if a block of data has been corrupted, used in ethernet and wifi packets

Security uses

- Detecting intrusions or suspicious traffic
- Capturing traffic during incident response or handling
- Looking for evidence of botnets, Trojans or infected systems
- Looking for traffic that exceeds certain thresholds
- Testing encryption between systems

Network uses

- Analyzing network problems
- Detecting misconfigured/misbehaving applications
- Gathering network usage and traffic statistics

- Debugging client/server communications

Spam filter

- Most spam filtering is centralized, more efficient than individual desktops
- Anti-spam software can be installed on SMTP servers or as a hardware appliance
- US CAN-SPAM Act (2003) regulates commercial emailers and protects consumers

Popular methods

- Blacklists
- Content or keyword filtering
- Trusted servers (Whitelists)
- Delay-based filtering - stops spambots that ignore the 'welcome banner'
- PTR and reverse DNS checks - check origin domain of email sender
- Callback verification - validate email address with mail server (can be spoofed)
- Statistical content filtering - learn when people mark messages as spam
- Rule-based filtering - keyword matches (e.g. "get rich")
- Egress filtering - scanning outbound mail for spam
- Hybrid filtering - a combination of the above techniques

Unified Threat Management (UTM) appliances

- Integrated firewall, IDS/IPS, antivirus, VPN, anti-spam, web traffic filtering, anti-spyware, content filtering, etc.
- A single appliance could be more efficient than integrating multiple components

URL filtering

- Blocks connections to websites based on a blacklist
- Web is volatile, blacklist needs constant updating
- Appliances can also inspect content and filter based on keywords, music/video content or anything that violates an AUP
- Can also be used to detect and block malware centrally

Web application firewall vs. network firewall

- Network firewall enforces policies based on network address rules
- Web application firewall (WAF) blocks based on HTTP/HTTPS traffic (content filters)

- WAFs can detect and block disclosure of critical data (e.g. credit card numbers) and protect websites from common attack vectors (e.g. cross-site scripting, fuzzing, buffer overflow)

Application-aware devices

- Network security was developed before application-level attacks were a concern
- Modern devices are application-aware and can do stateful inspection of traffic to block network-level and application-level attacks

Next-generation firewalls

- Firewalls capable of content-level filtering (application-level monitoring)

Application-aware IDS/IPS

- Application-aware IDS/IPS can detect malicious attacks to applications based on the application, not just the address

Application-level proxy

- Specialized, relay specific application traffic (e.g. HTTP conversations)

Secure network administration

Rule-based management

- Desired operating conditions are represented as *rules* (policy) enforced by *controls*

Firewall rules

- All firewall rulesets should deny any traffic that isn't explicitly permitted
- Rules are executed top down, processing stops if the conditions are met for a rule - so the order of rules matters
- The last rule should always be a "deny all" - block any traffic that gets to the last rule and still isn't explicitly allowed

VLAN management

- Some switches can enable virtual local area networks (VLANs), which act as partitioned, isolated broadcast domains even though they are on the same physical network
- VLANs allow network administrators to group hosts together even if they aren't connected to the same switch

- VLAN membership is established via *tagging*, *trunk port* and *physical address* (aka MAC address)
- *Trunking* is the process of spanning VLANs across multiple switches - packets from a single VLAN can travel between switches (hosts on different VLANs must use switches to communicate)
- VLAN segregation increases throughput and network security
    - Unused switch ports can be configured as empty VLANs that don't connect to the network (not open)
    - Traffic used to manage network devices can be segregated (e.g. remote access to a switch) but VLANs are not as secure as VPNs
- An attacker could reconfigure VLANs to gain access to secure portions of a network

Secure router configuration

- Restrict physical access
- Avoid passing cleartext administrative passwords (e.g. via Telnet, SNMP)
- Reset default passwords
- Configure via serial control interface port / out-of-band management

Exam tip: When connecting to a router for configuration, always use a secure connection to prevent eavesdropping.

Access control lists (ACLs)

- List of users and permissions (e.g. user ID, token, network address)
- Commonly used for evaluating access to filesystems based on user IDs
- Allowed traffic must be explicit, all other requests are denied (as with firewalls, order matters)

Port security

- Switches can control which devices and how many of them are allowed to connect via each port, using MAC addresses
- Provides some network security, even though MAC addresses can be spoofed

Variants

- Static learning - fixed MAC address assigned to a port, good for defined connections
- Dynamic learning - switch learns when devices connect, good for a small + limited number of machines
- Sticky learning - allows multiple devices to a port, but remembers them even after rebooting (attacker can't change settings through power cycling)

802.1x

- Also known as **port-based network access control** / authentication
- Authentication standard that handles communications between a user and an authorization device (e.g. edge router)
- Prevents unauthorized users from accessing publicly available ports on a switch
- *Extensible authentication protocol (EAP)* is a general protocol that can support multiple auth methods (e.g. one-time passwords, Kerberos, public keys)

Flood guards

- Flood guards monitor broadcast, multicast and unicast traffic and detect when to block traffic to manage flooding

Types

- Ping
- SYN
- Internet Control Message Protocol (ICMP) - aka smurf attacks
- Traffic flooding

Exam tip: Flood guards are implemented in firewalls and IDS/IPS systems to prevent DoS/DDoS attacks

Loop protection

- Open Shortest Path First (OSPF) is a routing protocol that creates IP tables for the Internet layer, and updates routing when a new device is added
- Spanning Tree Protocol (STP) prevents loops, which can happen when new devices are added

Implicit deny

- If an action isn't explicitly permitted, then the action is denied
- In rule-based systems, the last rule is often *deny all*

Network separation

- Prevents sensitive traffic from being sniffed through separation
- Examples include bridges, switches, VLANs
- Development, testing and production environments are often separated

Secure network design

DMZ security zones

- Layered defense - outer zones provide basic defense while innermost ones are the most protected
- Outermost zone is the internet - between the internet and the corporate network is the DMZ
- DMZs have a firewall on either side, machines within need to be *hardened* - this can limit their capabilities
- Any server accessed from the internet needs to reside in the DMZ (e.g. web, email) where traffic and behavior can be monitored

Subnet

- Subnetting allows you to organize a network and partition it for security purposes
- A subnet mask is a **32-bit number** that masks an IP address and divides it into a network address and a host address <network><host>
- For example, a Class A IP address consists of 8 bits identifying the network and 24 bits identifying the host. This is because the default subnet mask for a Class A IP address is 8 bits long (or, written in dotted decimal notation, 255.0.0.0)

Class	Addresses per network	Number of networks	Subnet mask
Class A (8-bit)	16,777,214	128	255.0.0.0
Class B (16-bit)	65,534	16,384	255.255.0.0
Class C (24-bit)	254	2,097,152	255.255.255.0

CIDR notation

- An IPv4 address contains **32 bits**
- CIDR notation makes it possible to specify in the IP address the number of bits that make the <network ID> portion of the address
- Before the implementation of CIDR, IPv4 networks were represented by the starting address and the subnet mask, both written in dot-decimal notation. Thus, 192.168.100.0/24 was often written as 192.168.100.0/255.255.255.0

Determine # hosts from CIDR notation

- 192.168.12.0**/24** tells you that the first **24 bits** of the IP address are used for *network routing*, which means there are **8 bits** left for calculating *host IDs*
- With 8 bits, there are **256** possible *values* (when you write out in binary: 128 + 64 + 32 + 16 + 8 + 4 + 2 + 1 and the 0 value)
- **Subtract 2** to get the total number of hosts

Determine CIDR from subnet mask

1. Convert last octet to binary
2. Identify remaining bits
3. Subtract remaining bits from **32** to get the CIDR notation

Determine # hosts from a subnet mask

1. Convert the last octet to binary
2. Identify remaining bits
3. Add the decimal values of those bits together + 1 (for the 0 value)
    - This gives you the total number of IP addresses
    - **Subtract 2** to get the total number of hosts

See also:

- Convert large numbers from decimal to binary
- How to calculate a subnet mask

Supernet

- A supernetwork, or supernet, is an IP network that is formed, for routing purposes, from the combination of two or more networks (or subnets) into a larger network.
- Supernetting within the Internet helps avoid topological fragmentation of the IP address space by delegating control address space segments to regional network service providers (regional route aggregation)

NAT

- NAT uses two sets of IP addresses for resources, internal and external
- It was developed to mitigate the depletion of IPv4 addresses and is used to translate between 2 addressing schemes (usually at the firewall/router level)
- Non-routable IP addresses will not route across the internet:
    - **Class A:** 10.0.0.0 - 10.255.255.255
    - **Class B:** 172.16.0.0 - 172.31.255.255

    - **Class C:** 192.168.0.0 - 192.168.255.255
- Conceals internal network structure from machines outside the firewall
- One of the methods used to enforce perimeter security because it requires users to access resources through firewalls and gateways
- *Static NAT* is a 1:1 binding used when external sources reference internal ones (e.g. web server contacting a database)
- *Dynamic NAT* is used for DMZ resources that need to access outside resources, an edge device manages the table and translation
- Finer grained control can be obtained through *Port Address Translation (PAT)* - allows an external address to serve 2 internal IPs through different ports (e.g. a web server and an email server)

Remote access

- *Remote Access Service (RAS)* is part of the Windows OS that allows remote connections via dial-up (includes authentication and maybe callback for added security)
- Can also mean *Remote Access Server* which regulates remote access
- Today remote access is mostly via VPN or desktop software

Telephony

- A *private branch exchange* (PBX) uses computer-based switching to connect business telephones to the local phone system
- Can be compromised by phone hackers (*phreakers*)
- Telecommunications firewalls are needed to protect PBX and data connections from unauthorized use (e.g. restricting use to office hours)

Network access control (NAC)

- NAC is the practice of managing endpoints on a case-by-case basis as they connect
- Two main competing methodologies
    - Microsoft Network Access Protection (NAP) - based on measuring the system health of the machine (e.g. patch levels)
    - Cisco Network Admission Control (NAC) - an appliance that enforces policies and allows 3rd party integration to verify client security posture
- Not widely deployed, but BYOD policies have increased interest in NAC

Virtualization

- Virtualization can improve security because it makes it easier to move, copy and backup systems, but securing the OS host and hypervisor is critical
- *Snapshots* make it easier to roll back a system if there is a bad change or incident
- Vulnerabilities exploiting the layer between virtual machines and/or hardware host are rare
- *Elasticity* is the ability of a system to expand/contract as requirements dictate

Cloud computing

- You already know what it is and why it's good
- Main security challenge is control over data that resides with a 3rd party - encryption is the main way to mitigate this risk

Types

- Platform as a service (PaaS) - ElasticSearch
- Software as a Service (SaaS) - Office 365
- Infrastructure as a Service (IaaS) - AWS

- Private cloud - reserved for an organization
- Public cloud - open to any tenant
- Community - shared by a group of like minded organizations (e.g. BPS)
- Hybrid cloud - combines elements of public/private cloud (e.g. data stored on prem)

Protocols

IPSec

- An open framework (vendor agnostic) for securely exchanging packets at Layer 3 via tunnel (e.g. VPN)
- In *transport mode*, packet data is encrypted and handled by endpoint computers
- In *tunnel mode*, the **entire packet** (including source/dest IP) is encrypted, placed inside another packet and passed between tunnel endpoints
- Both methods can be used at the same time (e.g. remote access into network)
- Traffic can be *unidirectional* or *bidirectional* (needs 2 security associations)
- Key management is collectively referred to as *Internet Key Management Protocol (IKMP)* or *Internet Key Exchange (IKE)*, and covers:
    - Internet Security Association and Key Management Protocol (ISAKMP)
    - Oakley

    - Secure Key Exchange Mechanism for Internet (SKEMI)
- Allows several technologies for protecting CIA:
    - Diffie-Hellman key exchange between peers on a public network
    - Public key signing of Diffie-Hellman key exchanges to guarantee identity
    - Bulk encryption algorithms like IDEA and 3DES for encrypting data
    - Keyed hash algorithms like HMAC, MD5 and SHA-1 for packet authentication
    - Digital certificates between 2 parties
- A *security association* is a algorithm + key for protecting a channel - two protocols collectively make up the *security association*:
    - *Authentication Header (AH)* - ensures integrity of data and authenticity of origin, including IP address
    - *Encapsulating Security Payload (ESP)* - provides security for the packet
    - Using both at the same time ensures integrity and privacy

Simple network management protocol (SNMP)

- See above, an application layer protocol for managing network devices
- A group that consists of SNMP devices and one or more SNMP managers is called an *SNMP community*

Secure shell (SSH)

- Encrypted remote terminal connection program, uses public key encryption and designed as a secure version of Telnet
- Uses **port 22**

Domain name server (DNS)

- Protocol for translating domains into IP addresses
- An *AAAA record* maps IPv6 address records
- Uses **UDP port 53** for standard queries and **TCP for large transfers** (e.g. zones)

Transport layer security (TLS)

- SSL uses public key encryption for confidentiality and integrity protection over the internet, but the current version (v3) is out of date and has been replaced by TLS

Transmission control protocol (TCP/IP)

- Maintains a connection between two endpoints and guarantees packet delivery, unlike UDP which is stateless and requires a reliable connection
- Initiates a connection via 3-way handshake: SYN, SYN-ACK, ACK

File transfer protocol (FTP)

- An insecure, plaintext method of file transfer using TCP port 21
- FTPS is the implementation of FTP over **SSL/TLS**, using TCP ports 989, 990
- SFTP is the implementation of FTP over **SSH**, using TCP port 22
- TFTP is for trivial file transfers with no security, using UDP port 69

Hypertext transfer protocol (HTTP)

- Forms the basis of the web with TCP/IP and uses TCP port 80
- HTTPS is the secure version and uses TCP port 443

Secure file copy (SCP)

- SSH-based file transfer, uses TCP port 22

Internet control message protocol (ICMP)

- ICMP is used for diagnostic, control and error messaging (e.g. ping)
- Typically blocked at IPv4 network boundaries because it can be used in attacks (e.g. ping floods), cannot be blocked in IPv6

IPv4 and IPv6

- Used for routing packets through associated protocols like TCP, UDP, HTTP,
- IPv4 is more common, uses 32-bit notation (x.x.x.x where x = 0-255)
- IPv6 replaces IPv4 and uses 128-bit notation (x:x:x:x:x:x:x:x, 8 groups of 4 hexadecimal digits)
- IPv6 offers benefits like autoconfiguration, enhanced security but presents challenges for network admins (i.e. does not need to use NAT)
- The loopback address for IPv6 is **::1**

Fibre channel

- High-speed network technology (up to 16 Gbps) used to connect storage to computer systems
- Requires special cables and can be expensive
- **Fibre channel over Ethernet (FCoE)** encapsulates fibre channel frames and can be used over 10 Gigabit ethernet networks

Internet small computer system interface (iSCSI)

- *iSCSI* is an adaptation of SCSI, a protocol for block storage, designed to work with TCP/IP Ethernet

- Used to send data over existing network infrastructures, creating Storage Area Networks (SANs)
- Seen as a low-cost alternative to *fibre channel*

Telnet

- Garbage cleartext transfer protocol, uses TCP port 23
- Largely replaced by SSH

NetBIOS

- A separate naming scheme from DNS, for file and printer sharing on old Windows networks
- Largely obsolete except for older, smaller networks

Secure wireless networking

- 802.11 protocol has been standardized for wireless networks, 5 are in production: 802.11a, b, g, n, ac

Service set identifier (SSID)

- 802.11 has rudimentary authentication that forces a client to perform a 'handshake' at the access point (AP) before associating with it, called the *SSID*
- The SSID is a 32-bit character identifier attached to the packet header and sent in plaintext - easily sniffed by attackers

Exam tip: Even though it's not that secure, renaming the SSID and disabling SSID broadcast are important to remember

Wired Equivalent Privacy (WEP)

- WEP is insecure because the *initialization vector* is sent in plaintext and contains the reused encryption key, allowing an attacker to compare ciphertexts to learn the key

Exam tip: WEP alone should not be used to provide confidentiality. If there's no other option, it should be placed outside the corporate firewall and require VPN

Wi-fi protected access (WPA)

- Intended to replace WEP, uses *Temporal Key Integrity Protocol (TKIP)* to assign 128-bit keys on a per-packet basis, improving WEP AP security without a hardware upgrade

- TKIP is flawed and has been deprecated with the release of WPA2

Wi-fi protected access 2 (WPA2)

- Also known as IEEE 802.11i, the standard for wi-fi security
- Uses 802.1x to provide authentication and AES + CCMP for encryption

Setup options

- Choose WPA2-Personal or WPA2-Enterprise as the security framework
- Choose AES for encryption (TKIP is deprecated)
- For network security key, WPA2-Enterprise (not shared, each user gets a key) is typically used in business environments

Wi-fi protected setup (WPS)

- Designed for home and small business, WPS allows devices to connect with an 8-digit pin and Extensible Authentication Protocol (EAP)
- Can be brute forced to reveal the PIN and the WPA/WPA2 passphrase, and should be disabled

Extensible authentication protocol (EAP)

- EAP is used for authentication
- EAP-TLS and variants (EAP-TTLS) are used to securely pass credentials and allow for the use of older authentication methods (e.g. PAP - password authentication protocol)

Protected extensible authentication protocol (PEAP)

- Designed jointly by Cisco, Microsoft and RSA
- PEAP encapsulates EAP with a TLS tunnel and is widely supported

Lightweight extensible authentication protocol (LEAP)

- Proprietary Cisco product but susceptible to offline password guessing, etc.
- Deprecated in favour of PEAP and EAP-TLS

CCMP

- Stands for *Counter Mode with Cipher Block - Chaining Message Authentication Codes Protocol* (or *Counter Mode with CBC-MAC Protocol*) - what a garbage acronym
- Used to provide message integrity for AES encryption, and requires new hardware

Wireless operations

MAC filter

- Selective admission of packets based on *Media Access Control (MAC)* addresses, which can be easily intercepted and spoofed
- *MAC limiting* limits the number of addresses a switch or port can learn, prevents a variety of flooding attacks by ignoring large numbers of "new" MAC addresses

Antenna placement

- Placement can have a big effect on radio frequency (RF)
- Trade-off: high-gain antennas are good with weaker signals but have less coverage while wide-coverage omnidirectional antennas have lower gain

Exam tip: Adjusting radiated power through power controls allows for help keep wireless signals from being broadcast beyond areas of physical access control

Power level controls

- Too much power invites interference, too little limits range
- Useful in enterprise environments to limit reach and create overlapping zones

Antenna types

- Omnidirectional antennas are not good with corners, boundaries or other hard to reach areas
- Directional antennas are needed to complete coverage: *panel* antennas are good for rooms with no bleed behind the antenna, *yagi* antennas are more like a beam over a long distance (higher sniffing risk)

Captive portal

- Using an HTTP client to handle authentication to a wifi network, often in public hotspots
- Intercepts all packets and returns the login page (which can be in a walled-off section of the network)
- Often used in NAC implementations, forcing a check prior to connecting

Site surveys

- A site survey involves mapping the floor plan, testing for RF interference, coverage and analysis (suggested placement of APs)

Compliance and operational security

Risk concepts

Risk

- The possibility of suffering harm or loss

Risk management

- The process of identifying threats and vulnerabilities, potential impact, cost of mitigating and deciding which actions are cost-effective for controlling these risks

Risk assessment

- The process of analyzing threats, vulnerabilities, impacts and mitigating actions

Asset

- Resource or information an org needs to conduct business

Mitigation

- Action taken to reduce the likelihood of a threat occuring (or reduce its impact?)
- Risks can be *avoided, transferred, mitigated or accepted*
- A risk that remains after controls are implemented is called *residual risk*

Control types

- **Technical** - technology interventions (e.g. passwords, antivirus software)
- **Management** - rules (e.g. policies, regulations, laws, planning, risk assessments)
- **Operations** - procedures (e.g. incident response, configuration, education, training)

For each class, there are 7 types of controls:

- Preventative
- Detective
- Corrective
- Deterrent
- Recovery
- Compensating

False positives and negatives

- False positive: when a test result indicates a condition that does not actually exist
- False negative: the failure of a system to detect a condition
- False positives can add to workload and create data fatigue, while false negatives can have significant consequences

Security program

- A security program is the sum of its technology, personnel, metrics, training, policies, procedures, standards, guidelines, etc
- Policies are high-level broad statements of what the organization wants to accomplish
- Standards are mandatory elements regarding the implementation and can be externally driven, like legal compliance:
    - Payment Card Industry Data Security Standard (PCI-DSS)
    - Gramm Leach Bliley Act (GLBA)
    - Health Insurance Portability Accountability Act (HIPAA)
- Guidelines are recommendations relating to a policy (non-mandatory)

The operational process:

1. Plan (adjust) for security
2. Implement the plans
3. Monitor the implementation
4. Evaluate the effectiveness (e.g. vuln assessment and/or pen test)

Privacy policy

- Organizations are obligated (sometimes by law) to describe how they handle customer information, including how it is secured - notice, choice, consent
- *Personally identifiable information (PII)* is any data that can be used to uniquely identify an individual

Safe harbour

Safe harbor is a term describing how US organizations opt to comply with much stricter EU privacy rules.

7 key principles:

- Notice

- Choice
- Onward transfer - disclosures of PII are consistent with previous principles
- Security
- Data integrity
- Access
- Enforcement

Information classification and handling

Exam tip: Information classification categories you should be aware of for the exam include: high, medium, low, confidential, private and public

Data labeling, handling and disposal

- *Wiping* works by going into a data structure and replacing it with an alternate pattern; modern *journaling* operating systems can undo wiping
- When deleting files, make sure to overwrite the drives, not just remove the pointers

Acceptable use policy (AUP)

- Outlines what the organization considers acceptable use of its resources, in order to promote productivity and limit liability
- Similarly, an AUP describes appropriate use by the organization (e.g. monitoring employees and expectation of privacy)
- Often closely linked with an organization's internet usage policy

Security policy

- A security policy is a high-level statement that outlines what security means to the organization and describes its goals and accountability structure, such as:
    - "This organization will exercise the principle of least privilege"
    - "Information will be provided on a need to know basis"
- Because they are high-level goals, policies should be updated less frequently than the procedures that implement them
- Some policies, like mandatory "use it or lose it" vacation, are intended to detect and prevent nefarious behaviour
- Job rotation and separation of duties serve a similar function, in addition to ensuring that more than one person has key information

Qualitative risk assessment

- Subjectively determining how an event might affect the business by considering probability and impact
- Threats can be organized using different levels of complexity (e.g. 3 levels would be low, medium, high)

Quantitative risk assessment

- Objectively determining impact through metrics and modelling, using historical data and trends (difficult to do well)
- It can be as simple as assigning scores to qualitative assessments or more complex (multiplying weight/importance of the factor by its assessed impact)
- Models are based on assumptions, which can be flawed - there is no substitute for experience and expertise

Risk calculation

Single loss expectancy (SLE)

SLE = asset value x exposure factor

- Example: A small building and its contents is worth $2m. It has a call center, which if destroyed would reduce business capability by 50%. The SLE = $2 million x 0.5 = $1 million

Annualized loss expectancy (ALE)

ALE = SLE x ARO

- ALE is calculated by multiplying the SLE by how many times the event might happen in a year, called the *annualized rate of occurrence (ARO)*.
- Example: If the ARO is once every 20 years, it is represented as 1/20. The ALE = $1 million x 1/20 = $50,000
- Ergo a countermeasure to protect the business should cost no more than $50,000

Impact

- Impact levels need to be defined according to business context (what is bad?)
- *Mean time to repair (MTTR)* is a measure of how long it takes to repair a given failure
- *Mean time between failure (MTBF)* is the average time between failures

MTBF = **Σ** (start of downtime - start of uptime) / number of failures

Example: 3 identical systems: first fails at 10h, second fails at 11h, third fails at 12h. The MBTF is 11h

- *Mean time to failure (MTTF)* means the same thing as MTBF

Availability

Availability = MTTF / (MTTF + MTTR)

- Assuming that a system has an MTTF of 6 months and the repair takes 30 minutes, Availability = 6 months / (6 months + 30 minutes) = 99.9884% [you can't do this without converting everything to minutes, and multiplying the answer by 100 to get %]

Recovery time objective (RTO)

- Describes the target time to resume operations after an incident (business continuity - shorter is typically more expensive)

Recovery point objective (RPO)

- Recovery point objective (RPO) represents the maximum allowable data loss (determines frequency of backup)

Quantitative vs. qualitative risk

- Purely quantitative risk management is impossible, but purely qualitative risk management can be done
- How this is balanced depends on management style and how difficult it is to quantify certain factors (e.g. asset value, threat frequency)

Vulnerabilities

- Vulnerabilities are characteristics of an asset that can be exploited to cause harm
- They can be fixed, removed and/or mitigated

Threat vectors

- A threat is any circumstance or event with the potential to cause harm
- A *threat vector* is the path or tool used by an attacker to cause harm

Popular examples:

- The web
- Unsecured wireless hotspot
- Mobile devices
- USB (removable) media
- Email (links, attachments, malware)
- Social engineering (deception, hoaxes)

Cloud and virtualization risk management

Challenges:

- Responsibility: who is responsible for what, and to what degree? (cloud provider, service provider, user)
- Data: safeguarding data that isn't under the organization's direct control (e.g. using encryption)
- Virtualization doesn't have these issues (on prem) but the ability to make complete snapshots and replicate entire systems exists in both paradigms

System integration processes

Onboarding and offboarding business partners

- When a contract with a 3rd party ends, data retention and destruction needs to be considered (and enforced within the contract)
- Social media is used for business but typically does not allow negotiation of terms, so legal exposure in this area should be considered

Interoperability agreements

Common agreements include:

- Service level agreement (SLA) - service expectations between business and supplier, including service definitions, performance and how changes/issues will be handled
- Business partnership agreement (BPA) - terms between partners
- Memorandum of understanding (MOU) - bilateral
- Interconnection security agreement (ISA) - security requirements between systems
- Agreements can also cover handling of personal/personally identifiable information:
    - Collection and storage
    - Purpose and use

    - Security controls and storage time
- Control over data (ownership, backups, security controls) is also an issue, especially when it comes to delineating what can be shared by 3rd parties

Risk management

Change management

- Ensures that proper procedures are followed when modifying IT infrastructure
- Normal for an enterprise to have a Change Control Board to approve all production changes and ensure that change management procedures are followed
- *Configuration control* is the process of controlling changes to items that have been baselined (how much of this actually applies in Agile/test-driven environment?)

Incident management

- Organizations should have a plan for handling interruptions arising from cyber attacks
- A *Computer incident response team (CIRT)* or *Computer Emergency Response teams (CERT)* can be part of this plan
- CIRTs usually have permanent members and ad hoc specialists, including legal staff
- *Incident response* is the range of actions performed in response to triggering events

Routine audits

Verifies compliance with security policy and identifies anomalies (e.g. logins at night).

Examples:

- User access
- User rights (e.g. after a job change)
- Storage (e.g. size, appropriateness of materials stored)
- Retention (e.g. how long should a record be kept, legal compliance)
- Firewall rules (e.g. ensuring that temporary rules don't become permanent)

Data loss or theft

Policies and procedures to prevent data loss and theft:

- Firewalls
- Network segregation

- Access controls + audit logs
- *Data Loss Prevention (DLP)* products - scanning packets for indicators (e.g. account numbers), often thwarted by encryption

Business continuity concepts

- *Business continuity plan (BCP)* - keep an org running after disruption
    - An IT contingency plan is typically part of a BCP and used more frequently
    - May also include succession planning to ensure continuity of personnel
- *Disaster recovery plan (DRP)* - recover and rebuild after a disruption, also includes safety procedures for staff (e.g. evacuation)

Exam tip: The above terms terms are often used interchangeably but there are subtle differences

Continuity of operations

- **Identify critical systems and components** - the "crown jewels"
- **Business impact analysis** - what the loss of critical functions would mean to an org
- **Remove single points of failure** - people or products (e.g. a firewall)
- **BCP planning and testing**
- **Continuity of operations** - what functions need to continue during a disruption

Disaster recovery planning

Once there is an inventory of critical business functions, a DRP should include the following:

- Who is responsible for the operation of a function?
- What do these individuals need to perform the function?
- When should this function be accomplished relative to other functions?
- Where will it be performed?
- How will it be performed?
- Why is the function critical to our organization?

High availability

- High availability requires that both data and services be available
- *Redundancy* increases reliability (e.g. having spare parts, extra internet connections)
- *Fault tolerance* involves mirroring data and systems so that disruptions don't affect the user
- An *active-active architecture* processes data on both the primary and backup systems, which allows instant failover but requires bidirectional replication

Exam tip: High availability is about maintaining data and services in an operational state, while fault tolerance is a design goal to achieve high availability even if a fault occurs

Redundant array of independent disks (RAID)

- **RAID0** (striped disks) spreads data across many disks to speed up retrieval but does not increase reliability (if one disk blows, the data will be lost)
- **RAID1** (mirrored disks) copies the data from one disk to 2+ other disks to speed up retrieval and improve reliability, but can be expensive
- **RAID2** (bit-level error correcting code) stripes data at the bit level and recovers data using error correcting techniques
- **RAID3** (byte-striped with error checks) stripes data at the byte level with one disk dedicated to parity (which is a bottleneck)
- **RAID4** (dedicated parity drive) larger stripes than RAID3 with one disk dedicated to parity (similar bottleneck to RAID3)
- **RAID5** (block-striped with error check) spreads parity data across drives, requires a minimum of 3 drives and improves both speed and reliability - ***this is the most common method***

Exam tip: Knowledge of basic RAID structures will be tested

Clustering and load balancing

- A group of systems linked and working together is known as a *cluster*
- *Load balancing* distributes processing across 2+ systems, increasing fault tolerance

Disaster recovery concepts

Backup plans and policies

Backups are important for disasters, but also minor issues (e.g. user error) and rolling back to a pre-intrusion state after security incidents (e.g. ransomware)

Considerations:

- How frequent?
- How extensive?
- What is the process and who is responsible?
- Where will they be stored?
- How long will they be retained?

- How many copies will be maintained?

Types:

	Full	Differential	Incremental	Delta
Amount of space	Large	Medium	Medium	Small
Restoration	Simple	Simple	Involved	Complex

- A *differential backup* saves changes to files and software since the last full backup
- Recovering from a differential backup involves loading the full backup, then applying the differential backup
- *Incremental backup* is a variation of differential, it saves changes since the last full *or incremental* backup
- Recovering from an incremental backup involves loading the full backup and every incremental backup after that
- *Delta* or *transactional backups* store only the portions of files that have changed (e.g. individual database records) - recovery usually requires an application
- An *archive bit* is a file attribute set whenever a file is modified, turned off after

Exam tip: You'll be tested on this for sure

Backup strategies

- **Rule of three** - keep the 3 most recent backups, overwriting the oldest when a new one is created
- Keep the most recent backups for various time intervals (e.g. roll back to different weeks)
- Calculating the cost of a backup strategy:
    - Cost of media for a single backup
    - Storage cost and retention policy
    - Labour costs to perform a single backup
    - Frequency of backups

Backup processing facilities

- A *cold site* has the basic environmental controls to operate but no hardware or data backups, can take weeks to operationalize
- A *warm site* is somewhere in between with recently data backups, can be made operational in days

- A *hot site* is a fully configured environment with mirrored/near-mirrored data that can be made operational in minutes or hours

Digital forensics and incident response

Forensics refers to the application of scientific knowledge to legal problems and typically covers:

- Investigating systems relating to a violation of laws
- Investigating systems relating to compliance with an organization's policies
- Investigating systems that have been remotely attacked

Forensic procedures

The 4-step process:

- **Collection** - searching for, recognizing and collecting evidence
- **Examination** - facilitating visibility, revealing hidden information
- **Analysis** - structuring evidence for use in future proceedings
- **Reporting** - creating a written report of findings

Evidence

Evidence can be documents, verbal statements and material items admissible in court. Evidence handling can be challenged in court, so it must be properly acquired, identified, protected against tampering, transported and stored.

Standards for evidence

- **Sufficient** - convincing, unquestionable
- **Competent** - legally qualified and reliable
- **Relevant** - relates to the case at hand

Types of evidence

- **Direct** - oral testimony (e.g. eyewitness statement)
- **Real** - tangible objects that prove/disprove a fact (e.g. link suspect to crime scene)
- **Documentary** - business records, manuals, logs (often used in computer crimes)
- **Demonstrative** - model, experiment, chart used to aid the jury

3 rules regarding evidence

- **Best evidence rule** - originals are preferred but copies (e.g digital records) may be accepted in unusual circumstances
- **Exclusionary rule** - evidence collection must be lawful (e.g. network sniffing is not permitted unless stated in an org's policy and employee has consented)
- **Hearsay rule** - typically computer evidence is "second-hand" because the computer itself can't be interrogated

Handling evidence

- **Capture system image** - Imaging or dumping the physical memory of a computer (RAM) can identify evidence not available on the hard drive (e.g. rootkit disabling event logging)
- **Hash values** - to quickly verify that files haven't been tampered with, hashing algorithms are used to create *message digests.* MD5 and SHA are common, but have flaws, so SHA-2 / 3 are preferred
- **Record time offset** - know the difference between a system timestamp and actual time (e.g. timezone)
- **Network traffic and logs** - firewall and IDS logs, network flow data, event logs
- **Multimedia** - photos and videos are helpful (e.g. objects, serial numbers) but should be taken with external cameras because of tampering risk

Examination

- Pulling the plug will freeze a computer, but lose everything on RAM and there may be "bombs" in the system - important to document actions taken and rationale, ideally with a witness

Order of volatility

1. CPU, cache and register contents (collect first)
2. Routing tables, ARP cache, process tables, kernel statistics
3. Live network connections and data flows
4. Memory (RAM)
5. Temporary file system/swap space
6. Data on hard disk
7. Remotely logged data
8. Data stored on archival media/backups

Exam tip: A common data element in the forensics process is accurate system time wrt an external time source. A *record time offset* is calculated by measuring system time against an

external clock such as a **Network Time Protocol (NTP)** server. It is best collected while the system is still running.

Chain of custody

Steps:

1. Record each item collected as evidence
2. Record who collected it along with date and time
3. Write a description of the evidence
4. Put the evidence in containers and tag them with case number, collector, date, time
5. Record all hash values in documentation
6. Securely transport and store the evidence in a protected facility
7. Obtain a signature from the person who accepts evidence at the facility
8. Provide controls to prevent access/tampering while stored
9. Securely transport evidence to court for proceedings

Exam tip: Never analyze seized evidence directly, secure it within a chain of custody and examine a forensic copy instead. Using a *write blocker* allows the disk to be read during copying, but prevents any writing. A good forensics process can prove that the forensic copy is identical to the original at the start and end of the examination.

Incident response procedures

There are 5 key steps in an incident response cycle:

- **Discover and report** - have a process for reporting and resolving incidents
- **Confirm** - specialists should review the report to confirm whether or not a security incident has occured
- **Investigate** - a response team should investigate the incident in detail and devise a recovery plan
- **Recover** - complete the investigation and take steps to return systems to normal operations
- **Lessons learned** - a post-mortem to correct weaknesses and suggest improvements

Roles and activities

- A first responder must do as much as possible to control damage or loss of evidence (e.g. capture logs, take photos)
- Multi-disciplinary incident response teams are needed to prepare for and respond to incidents

- Once an incident is identified, response, escalation and notification processes are triggered based on severity (e.g. port scans vs. port flooding)
- Affected components should be isolated to prevent further damage (e.g. quarantine or remove infected machines)
- Mitigation strategies vary based on data *state*:
    - Time - data spends more time in storage relative to transit or processing, and is susceptible to breach/compromise over long periods
    - Quantity - there is likely more data residing in storage than in any other state
    - Access - different protection mechanisms exist in each domain (wtf)
- Recovery can be a 2-step process: essential business functions first, followed by complete restoration of operations (without the original vulnerability)

Data breach

- A *data breach* occurs when sensitive information is copied, transmitted, viewed, stolen or accessed by an unauthorized party
- The primary mitigation steps in most cases are *data minimization* (don't keep what you don't need) and *encryption* to reduce the value of stolen data
- Data should be encrypted at rest and in transit

Security awareness and training

- User habits
- Acceptable use (see above)
- Password behaviours (enforced by a password policy)
- Data handling
- Shoulder surfing
- Clean desk policies
- Tailgating (following an employee with authorized access)
- Personally owned devices
- Zero-day exploits
- Social networking

Physical security and environmental controls

Fire suppression

- Water-based fire suppression (causes corrosion malfunctions)
- Halon-based fire suppression (banned / dangerous to humans)
- Clean agent fire suppression

- Carbon dioxide displaces oxygen and cools
- Argon lowers oxygen level, preventing combustion
- Inergen (N, Ar, CO_2) lowers oxygen level
- FE-12 (trifluoromethane) inhibits combustion like halon
- FE-200 (heptafluoropropane) inhibits combustion like halon

Fire extinguishers

Class	Fire type	Examples	Suppression method
A	Common combustibles	Wood, paper, cloth, plastic	Water or dry chemical
B	Combustible liquids	Petroleum products, organic solvents	CO_2 or dry chemical
C	Electrical	Electrical wiring and equipment, power tools	CO_2 or dry chemical
D	Flammable metals	Magnesium, titanium	Copper metal or NaCl

Exam tip: The most common type of fire extinguisher is ABC

Fire detectors

- Smoke detectors:
  - *Ionization* - small amount of radioactive material, smoke changes conductivity and triggers alarm
  - *Photoelectric* - light-emitting diode and light-sensitive sensor, smoke scatters light and triggers alarm
- Heat-activated fire detectors: *fixed temperature* / *fixed point* and *rate of rise / rate of increase* (can provide earlier warning but prone to false positives)
- Flame-activated fire detector: detects infrared changes from fire

EMI shielding

- Density of hardware in a data center can cause EM interference: *narrowband* and *broadband*
- EMI prevented by shielded network cabling (e.g. the twist in UTP) and grounded metal cases
- Faraday cages are a continuous covering of conductive material to stop EMI

- NSA's *TEMPEST* technology prevents the observation of computer screens via EM radiation

Exam tip: Understand the principles behind HVAC, fire suppression, environmental controls and EM shielding

Hot and cold aisles

- High-density data center design requires all intake fans to face the cold aisle and all exhaust fans to face the hot aisle
- HVAC then pushes cold air up through tiles in the cold aisle, while hot air is captured by ducts in the hot aisle
- Controls airflow and prevents mixing of hot and cold air

Environmental monitoring

- Electronic tracking of temperature and humidity
- Sensors can be remotely controlled, like other data center elements

Physical security

- Low security locks can be easily compromised (e.g. *bump keys*)
- *Fail-safe* locks unlock when the power goes out or during emergency, for safe exit
- *Mantraps* prevent tailgating by requiring card swipes at 2 doors close together
- Physical access is the most common way to obtain a drive image - it leaves no trace and bypasses all access controls - can be mitigated using drive encryption or a central file server

Video surveillance

- Closed-circuit television (CCTV) can be analog (requires multiplexer to present multiple views on a monitor) or digital / IP-based
- IP-based cameras are susceptible to the same attacks as other IoT devices
- Can be combined with facial recognition

Proximity readers

- Contactless access cards can control electronic doors and be combined with keypad PIN entry
- Readers are integrated into an access control list and can log usage

Other physical controls

- Signage and colour codes (e.g. for badge access levels)
- Lighting (e.g. to improve visibility for CCTV)
- Fencing (e.g. palisade - can't be climbed), walls and barricades
- Guards
- Biometrics (analog to digital conversion requires some margin for error, can result in false positives or negatives)
- Securing network cabling
- Alarms (calibrated to avoid alert fatigue)
- Motion detection
- Escape routes, plans and drills

Control types

- **Deterrent** - reduces likelihood of success or incentive to attack
- **Preventative** - prevents specific actions, such as tailgating
- **Detective -** alarms to alert responders
- **Corrective -** used post-event to minimize damage (e.g. backups)
- **Technical -** involves technology (e.g. biometrics)
- **Administrative -** policies and procedures (e.g. for guards)

Security controls

- Confidentiality is achieved through encryption (strongest), ACLs (scales well) or data hiding (bad practice)
- Integrity can be verified through *hashing*
- Availability is achieved through *redundancy* and *fault tolerance* (e.g. active-active architecture)

Threats and vulnerabilities

Malware and attack methods

Adware

- Legitimate use - showing ads in return for free use of software
- Nefarious use - software presents unwanted ads (e.g. cascading popups)

Virus

- Malicious code that attaches itself to an executable
- When executable is run, virus also executes (requires user action)
- First two types created were *boot sector viruses* and *program viruses*
- A *retrovirus* actively seeks out and disables antivirus software

Worm

- Self-replicating code that attempts to penetrate networks and computer systems
- Difficult to distinguish between viruses and worms, but worms typically do not need to attach itself to an executable (no user action required)

Spyware

- Monitors user behaviour without their knowledge (e.g. keylogging)
- Often banned by law, but this is circumvented by complex EULAs

Trojan

- Software that purports to do one thing but actually does something else
- Back Orifice (c. 1999) is an example of a Windows-based Trojan
- See also: *grayware* (software that isn't malware but not entirely good)

Rootkit

- Rootkits modify the operating system to enable non-standard functionality (e.g. elevated privileges)
- Example: Sony used rootkits to provide copy protection, but did not seek user approval and the rootkit created a vulnerability
- Rootkit types:
    - Firmware
    - Virtual
    - Kernel
    - Library
    - Application

Backdoors

- Methods used by software developers to access an application (e.g. hardcoded passwords)

- More commonly refers to programs that attackers install after gaining unauthorized access, to maintain access

Logic bomb

- Instructions that sit dormant until a condition is met (e.g. date)
- Often installed by authorized users (e.g. insider threat / disgruntled staff)

Botnet

- A bot is software that performs a task under the control of another program (a network of bots is a *botnet*)
- One famous botnet is *Zeus*, a keylogger that steals banking information and/or delivers ransomware

Ransomware

- Malware that extracts a ransome from the user, such as *Cryptolocker* and *WannaCry*

Polymorphism

- Trivial variations in code to evade signature-based malware detection

Armored virus

- Malware can typically be reverse-engineered to determine its origin, functionality, distribution method, etc.
- Malware can be armored (e.g. via encryption) to prevent criminals from stealing IP

Man in the middle (MITM)

- When an attacker places themselves between two hosts and intercepts traffic
- Commonly achieved by *session hijacking* + *cross-site scripting* (cookie theft)
- Interception can also capture key exchange, allowing attacker to decrypt traffic (see also TLS interception)

Denial of service (DoS)

- When an attacker crashes a system or overwhelms a machine with requests
    - Example: a *SYN flood* forces a system to initiate many 3-way handshakes (SYN, SYN/ACK, ACK) but the third packet never comes, so it hangs
    - Example: a ping of death (POD) sends an “unnatural” packet exceeding 64KB and crashes a vulnerable system

- A *distributed denial of service (DDoS)* involves multiple attackers vs. one host
- DoS and DDoS attacks can be mitigated by patching, changing SYN timeouts, distributed workload (e.g. CDN), firewalls (e.g. blocking ICMP packets)

Replay

- When an attacker captures communication between 2 parties and replays it (e.g. replaying a transaction or an authentication method)
- Replays can be prevented by using encryption, cryptographic authentication and timestamps (e.g. quick expiration)

Spoofing

- When an attacker makes data look like it's from another source (e.g. domain, email, IP addresses)
    - Example: Telnet to port 25 on a mail server and fill out any address in From or To (assuming there is no validation)
- Spoofing can also involve acquiring similar or slightly misspelled domains
- When impersonating a system, it makes sense to DoS the legitimate one so that it can't interfere with an attack
    - Administrators are encouraged to limit trusted relationships between hosts (what does this mean, specifically?)
    - Firewalls should also be configured to reject external packets with internal IP addresses (obvious clue that spoofing is being attempted)
- Spoofing is easier to do inside the network because traffic can be observed and proper *sequence numbers* for packets can be constructed (e.g. 3-way handshake)
- Sequence numbers for each session are started from different numbers, and incremented by large numbers (and maybe incremented X time interval)

Smurf attack

- An attacker sends an ICMP packet with a forged IP address to the broadcast address for a network, so that all systems receive it
- The normal response is an *echo reply*, which immediately floods the forged host with up to 254 packets

ARP poisoning

- When a machine sends an ARP request to the network, the reply is received and entered into all devices that hear the reply, to facilitate efficient lookups
- ARP poisoning corrupts the *ARP table* which is like short-term memory for lookups:
    - Example: "Who has this IP address?" "I do, my MAC address is…"

    - Example: “Who has this MAC address?” “I do, my IP address is…”
- Typically used by an attacker to inject themselves into a conversation (MITM)

Client-side attacks

- **Injection attacks**
- **Header manipulation -** occurs when HTTP headers are dynamically generated based on user input, elements can be manipulated by attackers (e.g. XSS)
- **Password attacks -** bad policy, bad choices, dictionary, brute force
- **Birthday attack -** exploits the mathematical probability of certain values coming up more often than intuitively expected
- **Rainbow tables -** precomputed hash values for passwords (no longer requiring computation), best defense is salted hashes

Other attacks

A list of attacks you already know:

- **Spam** (and spim for instant messaging)
- **Phishing**
- **Spear phishing**
- **Vishing** - phone-based phishing
- **Xmas attack** - port scanning and sending a packet with all flags (FIN, URG, PSH) to bypass a stateless firewall which only checks for SYN floods
- **Pharming** - misdirecting users to fake websites, may include DNS poisoning (e.g. to enter credentials)
- **Privilege escalation** - taking actions to obtain “better” credentials
- **Malicious insider threat** - abusing trusted access, can be mitigated by HR screening and separation of duties (e.g. sysadmins can’t manipulate their own logs)
- **DNS / cache poisoning** - see also Kaminsky attack
- **TCP/IP hijacking** - taking control of an existing client-server session (usually web or telnet)
    - *Transitive* access is exploiting a trusted relationship where one party is less protected, to gain access to the other party (e.g. client-server)
- **Typo squatting** / URL hijacking
- **Watering hole attack** - planting malware where specific people visit (e.g. website)

Exam tip: Understand how hijacking attacks are performed through poisoning addressing mechanisms

Social engineering

Techniques

- **Shoulder surfing**
- **Dumpster diving**
- **Tailgating**
- **Third-party authorization** - the illusion of authority / belonging (e.g. tech support, contractors)
- **Phishing / Vishing**
- **Hoaxes**
- **Whaling**

Principles

- **Authority**
- **Intimidation**
- **Consensus / social proof** - manipulating group decisions
- **Scarcity** - bringing what is needed to secure acceptance
- **Urgency** - forcing a decision by making it time-sensitive
- **Familiarity / liking**
- **Trust**

Application and wireless attacks

Wireless attacks

- **Rogue access points**
- **Jamming** / interference - interfering with radio signals
- **Evil twin** - usually enhanced and designed for MiTM
- **War dialing / war driving** - discovering unprotected modems or wifi access points
- **War chalking -** marks in urban areas showing unprotected wifi access points
- **Bluetooth attacks -** exploiting devices which advertise their capabilities
- **Bluejacking -** sending unsolicited messages to Bluetooth devices
- **Bluesnarfing -** using Bluetooth to copy user information off a device (e.g. Bloover)
- **Bluebugging -** using bluetooth to establish a serial connection (full control)
- **Packet sniffing -** mitigated by encryption (e.g. WPA2)
- **Near field communication -** susceptible to interception, corruption and device theft

- **Replay attack** (see above)
- **Initialization vector (IV) attack**

Exam tip: Bluetooth should always have discoverable mode turned off unless you're deliberately pairing a device

Application attacks

Cross-site scripting (XSS)

Code injection attack:

- **Non-persistent XSS attack** - immediately executed by the web server
- **Persistent XSS attack** - script is permanently stored on the web server
- **DOM-based XSS attack -** executed in the browser via DOM (not web server)

Purpose:

- Theft of authentication information from web application
- Session hijacking
- Deploying hostile content
- Changing user settings, including future users
- Impersonating a user
- Phishing or stealing sensitive information

Cross-site request forgery (XSRF)

- Unauthorized commands are sent from a user that the website trusts
- Unlike cross-site scripting (XSS), which exploits the trust a user has for a particular site, CSRF exploits the trust that a site has in a user's browser

Other attacks

- **Injections** - SQL, LDAP (directory systems), XML
- **Directory traversal/command injection** ("../..")
- **Buffer overflow** - why running programs as root is dangerous, more than 50% of sec incidents are due to buffer overflow (e.g. Heartbleed)
- **Integer overflow** - large value could roll over into negative, create logic errors
- **Zero-day**
- **Cookies** - name/value pairs: Expires, Domain, Path, Secure
- **Attachments** - altering files (e.g. JPEGs) to contain executable code that runs when the file is loaded
- **Locally shared objects (LSOs)** - Flash cookies

- **Malicious add-ons** - browser helper objects (BHOs), ActiveX content
- **Session hijacking** - importance of closing a session (e.g. logging off)
- **Sidejacking** - sniffing a session cookie if there is no/poor TLS

Mitigation techniques

Monitoring system logs

- Deciding what should be logged is based on what is required for a forensic record
- Events from different OSI layers can be logged in a common scheme:
    - **Security applications** - anything that helps assess/protect the network
    - **DNS server** - logging resolution requests, updates and forwards
    - **System logs** - program failures, crashes, processes
    - **Performance** - memory, CPU, disk usage (tracking busy vs. idle times)
    - **Event logs** - tracking what happens in an application
    - **Access** - tracking user access, failed logins, time and activity
    - **Firewall** - attempted connections, ports, source and destination
    - **Antivirus** - infections, scanning activity
    - **IDS/IPS logs** - suspicious activity

System hardening

Preparing and securing a system by removing all unnecessary software and services:

- **Protocols** - Telnet, NetBIOS, IPX, FTP
- **Shares**
- **Services and ports** - TCP, ECHO, CHARGEN
- **Rename the administrator account** and secure with a strong password
- **User accounts**
- **Maintain patches**
- **Control physical access**
- **Maintain current patch levels**, even for network devices (e.g. firewalls)
- **Change SNMP defaults** (simple network management protocol)

Recording **hash values for critical files** will allow administrators to detect changes (see also Tripwire)

Password policy

Components of a good password policy:

- At least 8 characters
- At least 3 of these 4 elements:
    - Uppercase letters
    - Lowercase letters
    - Numbers
    - Special characters
- Should not consist of dictionary words, user's login, or personal information (e.g family name)
- Lockout after 3 attempts, with a reset time of 5 minutes (optimal)
- Expired on a routine basis (e.g. 60-90 days)
- Prevent use of previous passwords (e.g. last 5-10)
- Protecting against misuse (e.g. sharing)
- Consequences for violating policy

Baseline configuration

- The process of setting up a system's security state by disabling unneeded things, then applying patches and configuration (good for enterprise consistency)
- Software updates follow this hierarchy:
    - Hotfix - small update to solve a specific problem
    - Patch - larger update that addresses many issues
    - Service pack - large collection of patches and hotfixes
- Large enterprises use continuous monitoring to identify updates
- Remediation beyond patching may involve architecture or configuration changes

Threat and vulnerability discovery

Tools

- **Protocol analyzer** - capture and examine signals and data across a communications channel (e.g. network sniffer, using SPAN ports)
- **Vulnerability scanner -** learn where control are missing / ineffective:
    - Nessus - **network vulnerability scanner**
    - Microsoft Baseline Security Analyzer (MBSA) - **host scanner**
    - Burp Suite - **web vulnerability scanner**
    - **Intrusive vs. non-intrusive** - does it interact with the vulnerability?
    - **Credentialed vs. non-credentialed** - whether or not it has a login
- **Honeypot and honeynet**
- **Port scanner** - search for hosts, open ports, services

    - Ports may show as open, closed or filtered (by firewall)
- **Banner grabbing** - some protocols have welcome messages which can be used to inventory services

Tools can be *active* (detectable, like nmap) or *passive* (not detectable, like Tripwire).

Assessments

- A *threat assessment* enumerates threats facing an enterprise, then estimates the likelihood / impact
- A *vulnerability assessment* evaluates the security posture of the network and its critical systems
- *Baseline reporting* describes which software/components need updating
- *Code review* is proofreading code, much cheaper to catch a bug before software enters formal testing or release
    - The *attack surface* of software is the code that can be accessed by unauthorized parties (how many features are available for attack)
- **Review architecture -** how data flows, ACLs, user connections
- **Review designs -** ACLs, data security, system configuration (how is this different than architecture?)
- **Open vulnerability assessment language (OVAL)** - communicate vulnerabilities in a structured way

Penetration testing

Simulates an attack from a malicious outsider:

    - Verify a threat exists (focus on real-world attack vectors)
    - Bypass security controls
    - Actively test security controls
    - Exploit vulnerabilities
- **Black box testing -** determine functionality by trying inputs and observing outputs
- **White box testing -** testing with comprehensive system knowledge
- **Gray box testing -** a mix of the above (some knowledge, some discovery)

Application, data and host security

Application security controls

- **Error and exception handling** - capture in a log, never present to user; all errors and exceptions should be trapped and handled in the *generating routine*

- **Input validation** - consider all inputs to be hostile
- **Output validation** - prevent harmful content or unauthorized data disclosure
- **Fuzzing** - trying a series of malformed inputs to find input validation errors
- **XSS** - prevented by anti-XSS library functions, testing a variety of inputs
- **XSRF** - prevented by additional authentication tokens/data in web requests
- **Validate input on server-side** (not just client-side, even if it's faster)

Application hardening

Similar to system hardening - starts with removing unnecessary components, properly configuring, patching, and monitoring regularly. Requires:

- An *application configuration baseline*
- *Patch management* process

NoSQL databases vs. SQL databases

- SQL databases are very common
- NoSQL databases are not relational and are used frequently for big-data and real-time functions
- Many NoSQL stores compromise consistency (in the sense of the CAP theorem) in favor of availability, partition tolerance, and speed

Mobile device security

Asset control

Mobile device management (MDM) describes the set of commonly used protection elements for mobile devices:

- **Full device encryption**
- **Screen lock -** automatically after a certain period of inactivity, with a strong password
- **Remote wiping**
- **Automatic wiping** - after a certain number of failed login attempts
- **Remote lockout**

Other protections:

- **GPS locating** - for lost devices
- **Storage segmentation** - separating personal and work profiles

- **Application control and whitelisting** - only certain apps permitted
- **Removable storage** - scanning with antivirus before they connect to network
- **Key and credential management** - risk of strong corporate credentials being protected by weak screen passcodes
- **Geo-tagging** - disabling to prevent location disclosure via multimedia
- **Encryption** - App data should be encrypted and segregated to prevent rogue app access
- **Transitive trust / authentication** - higher risk of trust-based attacks with mobile devices because they connect to many domains and tend to have business and personal profiles

Bring your own device (BYOD)

Advantages and disadvantages of BYOD:

- **Data ownership** - keeping work data in a managed container reduces risk
- **Support costs** - help desk must be familiar with a broader set of devices
- **Patch management -** ensuring that devices are patched, no jailbreaking
- **Antivirus management**
- **Forensics -** policies required for handling both business and personal profiles
- **Employee expectation of privacy -** higher in a BYOD setting, requires policy
- **Onboarding and offboarding** - account creation and termination
- **Adherence to corporate policies -** AUP

Host-based security

OS hardening

Disabling unneeded features and meeting these additional requirements:

- Software comes from a trusted source
- Machines are connected to a trusted network during installation
- Installation includes all current patches / service packs
- Backup is taken after installation for future recovery

A *trusted OS* is certified for use by governments, using the *Common Criteria for Information Technology Security Evaluation (CC)* and commonly employ *mandatory access control (MAC)*

Antivirus

- **Signature-based scanning**
- **Heuristic scanning** - either *weight-based* or *rule-based* behavioural analysis

    - **Automated updates**
    - **Automated scanning**
    - **Media scanning** - removable media like USBs
    - **Manual scanning** - on demand
    - **Email scanning -** antivirus software often installed on mail servers
    - **Resolution -** quarantining or deleting an infected file
    - **Popup blockers**

Application updates

- Applications can be *blacklisted* but this does not protect against dynamic threats
- Application *whitelisting* helps identify permitted applications, but there may be many
- Microsoft has 2 mechanisms for application control:
    - **Software restrictive policies** - usually set by machine, not user
    - **User account level control** - which users can execute which programs

Host-based firewalls

Most OSes come with a software-based "personal" firewall:

- Microsoft has included host-based firewalls since Windows XP
- Linux has built-in firewalls: TCP Wrappers (simple allow/deny), ipchains (rules for input, output, forward) and iptables (more granular)

Host-based intrusion detection

- Examines log files, audit trails and incoming/outgoing network traffic
- Looks for anomalous behaviour:
    - Logins at odd hours
    - Login authentication failures
    - Addition of new user accounts
    - Modification or access of critical system files
    - Modification or removal of binary files (executables)
    - Starting or stopping processes
    - Privilege escalation
    - Use of certain programs
- As with any IDS, the *traffic collector* grabs the information needed for the *analysis engine*, which compares data against a *signature database*.
- A *passive* HIDS observes, while an *active* HIDS can be scripted to perform actions
- On Linux, a HIDS mostly inspects text-based logs, Windows has 3 main logs (Application, System and Security) which require a utility to view (Event Viewer)

Advantages of HIDS

- **Very OS and application-specific** with very detailed signatures
- **Reduce false positive rates**
- **Examine data after it has been decrypted** (unlike network-based IDS)
- **Determine system impact** more effectively (verify patch levels, presence of certain files, system state)

Disadvantages of HIDS

- **Must be installed on every system**
- **High TCO**
- **Uses local system resources**
- **Can't see surrounding activity** such as the network or other hosts
- **If logged locally, can be compromised**

Modern HIDS

The most recent HDS offerings are more like IPSes:

- Integrated system firewall
- Behavioural and signature based IDS
- Application control
- Enterprise management (agent reports back to central server)
- Malware detection and prevention

Hardware-based encryption

- **Trusted platform module (TPM)** is a hardware solution on the motherboard that assists with key generation, storage and RNG; not accessible via normal software channels
- **Hardware security module (HSM)** is a device used to manage and store encryption keys; usually peripherals like USBs that are more efficient and processing than general-purpose computers

Exam tip: Storing privacy keys on a networked device is dangerous, HSM mitigates that risk

Securing alternative environments

Supervisory control and data acquisition (SCADA)

- Also known as *distributed control systems (DCS)* and *industrial control systems (ICS)*
- Historically isolated from other systems (air gapped) but in recent times are often connected to the corporate network, increasing the attack surface

Embedded systems

- See also *mainframes*, *game consoles*, *in-vehicle computers*, *Internet of Things (IOT)*
- These systems also used to be compartmentalized, new security issues arising as they become interconnected

Exam tip: Static environments were configured for a specific purpose and are designed to remain unaltered through their lifecycle. Security risks can be mitigated through *network segmentation*, *application firewalls*, *firmware version control, security layers, TCP wrappers* (obscure or encapsulate true functionality, like a trojan or VPN)

Access control and identity management

Access control and authentication

- *Access control* refers to all security features design to prevent unauthorized access to a system or network
- *Authentication* refers to the process of verifying someone's identity
- The principle of *least privilege* strongly applies here

Types of access control

- **Discretionary (DAC)** - owner of an object can decide permissions for others
- **Mandatory (MAC)** - restricting objects based on sensitivity (e.g. High, Medium, Low, Confidential, Public, Private), known as *multilevel security*, this is the **most strict**
- **Role-based (RBAC)**
- **Rule-based** - uses business rules in ACLs (e.g. no employee can access X after hours)

Types of authentication

- **Username** (not secure)

- **Biometrics**
- **Smart card** - chip card
- **Common access card (CAC)** - for military personnel, DoD employees, contractors
- **Personal identity verification card (PIV)** - for US federal employees, contractors
- **Multifactor** - combining multiple forms of authentication
- **HMAC-based one-time password (HOTP)**
- **Time-based one-time password (TOTP)**
- **Challenge handshake authentication protocol (CHAP)** - supports PPP
- **Single sign-on** - convenient, but also riskier
    - Requires *federation* policies
    - See also: *transitive/trust authentication*
- **Mutual authentication** - both sides verify

Types of factors

- **Something you know**
- **Something you have**
- **Something you are** (e.g. static biometrics such as fingerprints)
- **Something you do** (e.g. dynamic biometrics such as gait, typing)
- **Somewhere you are**

Remote access

The process of connecting by remote access has two elements:

- Temporary network connection (e.g. internet)
- Protocols to negotiate privileges and commands

The three steps for establishing proper privileges are:

- **Authentication** - matching user-supplied credentials to previously stored credentials on a host (usually username and password)
- **Authorization** - granting of specific permissions based on the privileges held by the account
- **Accounting** - collection of billing and other usage records
- Also mentioned: identification, authentication, authorization

Remote authentication dial-in user service (RADIUS)

- Supports many authentication methods, including:
    - Point-to-point protocol (PPP)

    - Password authentication protocol (PAP) - deprecated, password in the clear
    - Challenge handshake authentication protocol (CHAP)
    - UNIX logins
- Connectionless client-server protocol (client is typically a Network Access Server / NAS) using UDP ports 1812 for auth and 1813 for accounting functions
- Client-server communication is encrypted but communications with user machine is passed in the clear
- RADIUS performs both authentication and authorization together in response to a single *Access-Request* message

Terminal access controller access control system+ (TACACS+)

- Uses TCP port 49 and separates authentication, authorization and accounting
- Like RADIUS, uses client server model with NAS as client, and communications with end user passed in the clear
- Also supports a variety of authentication methods (PPP, PPP EAP, PAP, CHAP, token cards and Kerberos)
- Default is “unknown user” which can be assigned some permissions
- Extended TACACS (XTACACS) is a newer version of the original TACACS, but both are considered obsolete

Kerberos

- Passes a symmetric key over an insecure network (e.g. Internet) using the Needham-Schroeder protocol - strong cryptography
- Uses a trusted 3rd party, known as a *key distribution center (KDC)* which consists of an *authentication server (AS)* and a *ticket granting server (TGS)*
- Similar to a driver’s license being used to identify someone, provided by a trusted 3rd party (government)

Lightweight directory access protocol (LDAP)

- LDAP uses the X.500 standard for directory services
- Secure LDAP uses TLS for encryption

Security assertion markup language (SAML)

- XML-based protocol for single sign-on for web applications
- Uses security tokens and an authoritative service provider (SP)

Account management

- **User** - unique IDs are important when it comes to investigating access control issues

- **Group** - reduces complexity and overhead vs. managing permissions by user
    - Enterprise Windows uses *Group Policy Objects (GPOs)* has settings for managing user credentials (e.g. password rules)
- **Domain password policy** - policy for a specific domain (usually means Active Directory domain controller)
    - Has an option for storing passwords using *reversible encryption* for authentication methods like CHAP, should be avoided

Cryptography

Cryptographic concepts

Symmetric

- Requires that both the sender and receiver have the same *key*, a *shared secret*
- Faster and less computationally involved than asymmetric keys
- Key management is important - the key must be known or securely transmitted to the receiver
- Common symmetric algorithms:
    - **DES** - NIST certified
    - **3DES** - *multiple encryption*, uses 2-3 keys instead of single key for DES, slow
    - **AES** - NIST Advanced Encryption Standard, uses Rijndael algorithm, replaces DES and considered the gold standard for strength and efficiency
    - **Blowfish** - designed by Bruce Schneier, fast except when keys need changing
    - **Twofish** - improvement over Blowfish for some weak keys
    - **RC4 (series)** - *stream cipher* designed by Ronald L. Rivest (MIT), fast and used widely in TLS and WEP/WPA

Exam tip: Know DES, 3DES, AES, Blowfish, Twofish and RC4 symmetric algorithms

Public key or asymmetric

- Uses a *public* and *private* key to simplify key management - public key can be sent to anyone and does not require secret transmission
- Based on complex math, usually factoring large numbers called *trapdoor functions* (easy to calculate one way, very hard to calculate the other way)
- Asymmetric keys are distributed using certificates to verify the public key (e.g. TLS handshake)

- Bulk encryption can be done by passing a symmetric key with asymmetric encryption, which is faster and allows for *perfect forward secrecy*
- Common asymmetric algorithms:
    - **RSA** - designed by Ronald L. Rivest + others, prime number factorization
    - **Diffie-Hellman** - used in SSL, SSH and IPSec, important because it allows 2 strangers to share a secret key (see below)
    - **Elliptical curve cryptography (ECC)** - small key size, good for low-power devices, high security
    - **ElGamal** - used in GNU privacy guard, PGP

Exam tip: Know RSA, Diffie-Hellman and ECC asymmetric algorithms

Session key

- A *session key* is used for encrypting messages during a communication session, generated by *random seeds*
- Provides more protection during a session, including *perfect forward secrecy*

Key exchange

- Early key exchanges were performed by trusted couriers, *out-of-band*
- Secure key exchange can happen even when packets are being intercepted, *in-band*
- *Diffie-Hellman key exchange* depends on two random numbers, each chosen by one of the parties and kept secret, a key is generated together and used
    - *Ephemeral keys* are used only once after generation, for perfect forward secrecy (*ephemeral Diffie-Hellman*)
    - *Elliptic-curve Diffie-Hellman (ECDH)* allows two parties to generate a shared secret using ECC

Cryptographic methods

- Most cryptographic methods rely on an *algorithm* and a *key*
- Cryptographic operations are used to:
    - Protect confidentiality
    - Protect integrity via hashing
    - Manage non-repudiation via digital signatures (providing proof of integrity and origin of data)

Block vs. stream

- *Block operations* are performed on blocks of data, enabling both *substitution* and *transposition* operations

- *Stream ciphers* are faster but only use substitution and don't provide integrity or authentication protections, used with web-based audio/video (data transfers in small pieces)

Elliptic curve cryptography (ECC)

- A simple function that is drawn as a gently looping curve, defined by: $y^2 = x^3 + ax + b$
- You can add two points on the curve together and get a third point on the curve, like a public key algorithm:
    - 2 users agree on a curve and a fixed curve point (public)
    - User 1 chooses a secret random number and computes a public key based on a point on the curve
    - User 2 does the same thing, now both can generate the same shared secret using the two public points
- Major advantages are speed (efficiency of calculation, especially for low power devices) and strength for a relatively small key length

Quantum cryptography

- *Quantum key distribution* involves encoding information in qubits instead of classical bits (e.g. photons) and transmitting them over a quantum communications channel, with no worry of eavesdropping because of the *observer effect* - easy to tell if data has been disturbed in transit
- Once an entire key is sent securely, symmetric encryption can be used for the actual data

Hashing

- One-way encryption that is difficult to reverse, used primarily for data integrity:
    - **Message authentication code (MAC)** is generated by hashing
    - **Hash-based message authentication code (HMAC)** is a hash algorithm using a previously shared secret, providing integrity and authentication
    - Hashing algorithms can be compromised by a *collision attack*, where an attacker finds two different messages that hash to the same value (difficult)
- Common hashing algorithms:
    - **MD2** - all MDs developed by Ronald L. Rivest (MIT)
    - **MD4**
    - **MD5** - better than previous versions but has weaknesses, 2 different .exe files can have the same MD5 hash value
    - **SHA-1** - developed by NIST, vulnerable to collision attack, deprecated

    - **SHA-2** - collective name for the algorithms below:
        - **SHA-224**
        - **SHA-256**
        - **SHA-384**
        - **SHA-512**
    - **SHA-3** - completely different from previous versions, replaces SHA2
    - **RIPEMD** - RACE Integrity Primitives Evaluation Message Digest, known to have collision problems, strengthened to RIPEMD-160

Cryptographic objectives

- **Perfect forward secrecy** - ensuring that past sessions can't be compromised by future compromises of secret keys or passwords
- **Transport encryption** - protect data that is in motion (e.g. SSL/TLS)
- **Non-repudiation** - verify that a message has been sent and received so that the sender can't refute sending (or receiving) the information
- **Key escrow** - when a private key is held by you and a 3rd party (e.g. government) if the key holder becomes inaccessible or data is required by court order
- **Steganography** - e.g. hiding messages in images via *least significant bit encoding (LSB)*
- **Digital signatures** - uses public key cryptography, allows traceability to the person signing the message through the use of their private key, with hashes used to verify integrity

Cryptographic applications

- **PGP** - uses both symmetric (bulk) and asymmetric (passing symmetric key) encryption
- **GnuPG/GPG** - open source implementation of the OpenPGP standard
- **NT LAN Manager** - Microsoft authentication protocol for use with *Server Message Block (SMB)* protocol, primarily used for file shares and printers. Mostly replaced by *Kerberos* but NTLMv2 still used for these functions:
    - Authenticating to a server using an IP address
    - Authenticating to a server belonging to a different Active Directory forest
    - Authenticating to a server that doesn't belong to a domain
    - No Active Directory domain exists (workgroup or peer-to-peer connections)
    - HMAC-MD5 is used as the challenge-response protocol
- **One-time pad** - key is the same size as the data being encrypted, unbreakable but impractical

- **Cipher suite** - arrangement of algorithms used for different functions (e.g. authentication, encryption, digital signature, hashing)
- **Key stretching** - strengthening weak keys by adding more computational complexity:
    - *Salt* - additional random data used with a one-way function, protecting against brute-force and rainbow table attacks
    - *Password-based key derivation function 2 (PBKDF2)* - uses a password + salt and applies an HMAC to the input thousands of times
    - *Bcrypt* - uses Blowfish cipher and salting to make brute-forcing unfeasible

Public key infrastructure (PKI)

- A person's public key must be bound to a person's identity to establish trust
- *Registration authorities (RAs)* and *certificate authorities (CAs)* act as trusted 3rd parties, verifying identity and signing the certificate with its private key
- CAs are made up of software, hardware, procedures, policies and people who are involved in validating individual identities and generating certificates
- RAs accept a request for a digital certificate and perform the steps to register and authenticate the individual. There are three types, with increasingly strict requirements:
    - **Class 1** - used to verify an individual's identity via email (e.g. to digitally sign and encrypt a message
    - **Class 2** - used for software signing by vendors who want to assure integrity (origin, no tampering)
    - **Class 3** - used by a company to set up its own CA, so that it can verify identity and generate certificates internally
- A *key store* is where keys can be held by an application for use by other applications (e.g. via API)

Exam tip: The RA verifies identity of the requestor on behalf of the CA. The CA generates the certificate using information forwarded by the RA.

Trust and certificate verification

- Trusted CAs' digital certificates and public keys are downloaded and stored on a local computer (e.g. browser certificate list)
- Some companies only allow internal certificates (e.g. digitally signed software can only be installed if signed by the company's CA) or with controlled policies (e.g. Entrust)
- Steps to validate a certificate:

    a. Compare with list of trusted certificates on the local machine
    b. Calculate a message digest
    c. Use CA's public key to decrypt the digital signature and recover the original message digest within the digital signature (validation)
    d. Compare the 2 message digest values to ensure integrity
    e. Review identification information, such as email address
    f. Review validity dates
    g. Check revocation list to see if the cert has been revoked
- Certificates are created and formatted based on the *X.509* standard, which describes the required fields and what can be entered into them:
    - **Version number** - X.509 version
    - **Subject** - owner of the certificate
    - **Public key** - identify the public key and algorithm used to create key pair
    - **Issuer** - CA that generated and digitally signed the cert
    - **Serial number**
    - **Validity** - date range
    - **Certificate usage** - approved use
    - **Signature algorithm** - hashing and digital signature algorithms used to digitally sign the cert
    - **Extensions** - allows additional data to be encoded (e.g. company customizations)

Certificate attributes

- **End-entity certs** - issued by a CA to a specific subject
- **CA certs** - can be self-signed or can be issued by a superior CA within a hierarchical model
- **Cross-certs** - used when independent CAs establish peer-to-peer trust relationships
- **Policy cert** - used to provide centrally controlled policy information to PKI clients

Certificate lifecycles

- **Registration and generation**
- **Certificate signing request**
- **Renewal** - less strict because it assumes prior registration
- **Revocation** - requires authentication before a request can be processed
- **Suspension** - putting a cert on hold temporarily
- **Destruction**

Exam tip: Certificate revocation checks are done either by examining the CRL or using the *Online Certificate Status Protocol (OCSP)*

Private key protection

- Appropriate size
- Appropriate lifetime (not using it past expiry)
- Properly destroyed
- Never exposed in cleartext
- Private key should not be copied or shared
- Secure storage
- Secure transport
- Review use with software
- Password protected

CA private key

- These are the most sensitive of all key pairs, because the trust relationship relies on them
- Often kept in tamper-proof hardware encryption store

Key recovery

- Companies archive keys in case the employee is no longer available
- *Key archiving* and *key recovery*, *recovery agent* (helps recover the key but might be a risk)
- At least two people can be required to authenticate by the key recovery software (known as *dual control*), to enforce *separation of duties*
    - Also known as *m of n authentication*

Public certificate authorities

- Public CAs specialize in verifying individual identities and creating + maintaining their certificates: Entrust, GoDaddy, VeriSign are examples
- Public CAs and their *root certificates* are installed and configured in web browsers by default
- There is no global regulation or standardization for certs or their classes, so most develop a *certificate policy*

Trust models

- **Trust domain** - a construct of systems, personnel, applications, protocols, technologies and policies that work to provide protection
- **Trust anchor** - an agreed-upon 3rd party (e.g. DMV/government)

- **Hierarchical** - the root CA is the ultimate trust anchor, with *subordinate CAs* (*intermediate* and *leaf*) below
    - Client software follows the *certificate path* until it finds a root certificate that is trusted
- **Peer-to-peer** - one CA is not subordinate to the other and have no common anchor, but certify each other's public keys, known as *cross-certification*
    - Scalability is an issue because each CA must certify every other CA that is participating
- **Hybrid** - two companies have their own internal hierarchical models and are connected through a peer-to-peer model using cross-certification
    - A *bridge* CA can also be used to issue cross-certificates for the connected trust domains

Appendix

OSI model

Please **D**o **N**ot **T**each **S**tudents **P**ointless **A**cronyms

OSI layer	Protocols
7. application	DHCP, DNS, FTP, TFTP, SSH, LDAP, IMAP, POP3, Gopher, HTTP, NFS, NNTP, NTP, SIP, SSI, SMPP, SMTP, SNMP, Telnet
6. presentation	MIME, XDR, EBCDIC, RDP
5. session	Named pipes, NetBIOS, PPTP, RTP, SAP, SOCKS, SPDY
4. transport	DCCP, SCTP, SPX, TCP, UDP
3. network	AppleTalk, IPv4, IPv6, EGP, EIGRP, ICMP, IPSec, IGMP, IGRP, MPLS, IPX, X.25, routers
2. data link	ARP, ATM, Frame relay, HDLC, IEEE 802.2, IEEE 802.3, LLC, L2TP, PPP, PPTP, LLDP, STP, SDLC, SLIP, X.25, switches, network cards
1. physical	Bluetooth, DSL/ADSL, ISDN, IEEE 1394, IEEE 802.3, IEEE 802.11, IEEE 802.15, IEEE 802.16, RS-232, RS-485, SONET/SDM, USB, cables

Common ports

TCP port	UDP port	Keyword	Protocol
20		FTP-Data	File transfer (default data)
21		FTP	File transfer control
22		SSH	Secure shell login
22		SCP	SCP uses SSH
22		SFTP	SFTP uses SSH
23		TELNET	Telnet
25		SMTP	Simple mail transfer
49		TACACS+	Remote access auth
53	53	DNS	Domains
	69	TFTP	Trivial FTP
80		HTTP	Web
88	88	Kerberos	Kerberos
110		POP3	Email
143		IMAP	Email
139	137, 138	NetBIOS	NetBIOS
389	389		LDAP
443		HTTPS	HTTPS
636		LDAPS	LDAPS
989, 990		FTPS	FTP over SSL/TLS
	1812, 1813	RADIUS	Remote access auth
3389	3389	RDP	Remote desktop

Helpful links

- Practice tests
- Acronyms