SY0-501 - CONTENT RESUME

CompTIA Security+

Version of February 25, 2018

Contents

1 Threats, Attacks and Vulnerabilities

1.1 Given a scenario, analyze indicators of compromise and determine the type of malware

Malware : type of software that is malicious. It performs bad functions on our computer or other computer on the network.

Properties :

- All malwares work together.

Examples :

- **Drive-by-download** : visit website => download malicious file directly.

Solutions :

1. Stay updated with latest patches to avoid vulnerabilities

1.1.1 Viruses

Virus (named after human viruses) : malware propagating through file system or from device to device through network.

Properties :

- Doesn't need user to click something but need to run or execute a program to be able to replicate.
- Some are simply invisible while other can e.g. make pop ups.

Examples :

- *Viruses associated with applications (run the application => run the virus).*
- *Viruses installing inside the boot sector (doesnt even need the OS to perform).*
- *Virus running script inside OS or browser.*
- *Macro viruses : **pas compris**.*

Solutions :

1. Use anti-virus and keep signature file updated (to stop it before it's installing)!

1.1.2 Crypto-malware/Ransomware

Crypto malware : Ransomware whose encrypting all data and asking for ransom. It encrypt everything except OS so that it can display message asking for untracable money to decrypt data.

Solutions :

1. Use offline[1] backup
2. OS updated
3. Signature antivirus list updated

[1] because modern crypto malware find ur backups online

1.1.3 Worm

Worm : type of virus that can move itself through a computer network between systems without any human intervention.

Properties :

- Replicate quickly ⇒ difficult to contain once propagitating.

Solutions :

1. Filter with firewall
2. Filter with **IDS** = intrusion detection system/**IPS** = intrusion prevention system

1.1.4 Trojan

Trojan horse (from greek trojan war) : Application pretending to be something else than malware.

Properties :

- Runs into computer to embedd itself and perform its functions.
- Once inside : free reign.
- Opens backdoor for other malware to go into the system.

Solutions :

1. Antivirus[2].

Remark : some software include backdoor. (e.g. old linux kernel, bad softwares).

1.1.5 Rootkit

Rootkit (from *root user*) : malware that modifies the kernel of the OS to avoid antivirus or antimalware softwares identifiying.

Properties :

- Invisible by nature : can't be found in task manager ⇒ difficult to remove.
- Usually combined with software to create a malware so that this software cannot be removed (*access denied*).

Solutions :

1. Anti-virus and anti-malware file up to date.
2. Rootkit remover.
3. Secure boot with UEFI BIOS

Remark : Highest level user in Windows = **administrator user**, in Linux = **root user**.

[2] stop it even if u performed the installing

1.1.6 Keylogger

Keylogger : piece of software saving all **keystrokes** putting in file and sending to bad guys.
Properties :

- Can also Store search engine queries or screenshots.
- Usually installed with a well-known malware.

Solutions :

1. Antivirus/malware updated with latest signatures.
2. Firewall rules/monitoring software : block data exfiltration attempt.
3. Run a keylogging scanner (check for keylogging activity).

1.1.7 Adware

Adware : malware that is popping-up advertisements and sometimes slowing down computer performances once installed.
Properties :

- Often installed as a trojan.
- Be careful with fake adware removers that just install more adwares.

Solutions :

1. Antivirus signature list up-to-date
2. Careful with applications you install
3. Backups because often implemented very deeply into systems
4. Run scans

1.1.8 Spyware

Spyware : type of malware watching what you are doing (websites browsing) in order to capture your habits and embed some code into sites u visit or wait for you to input some personnal information and steal it.
Properties :

- Often presented as fake security software or installed with other software like peer-to-peer software.

Solutions :

1. Antivirus signature list up-to-date
2. Careful with applications you install
3. Backups because often implemented very deeply into systems
4. Run scans

Remark : adwares and spywares are the most frequent type of malwares.

1.1.9 Bot/Botnet

Botnet (*robot network*) : virtual robot inside of a device and performing functions that are commanded to the device.
Properties :

- End-user has no idea that bot its running inside of computer.
- Usually coming from trojan horse (e.g. from email) or taking advantage of OS or application vulneratbility.
- Once installed : waits for commands from main central control system.
- Used alot for DDoS. Botnets are for rent.

Solution :

1. OS/application patches, anti-virus/malware updated signatures.
2. Scan system, network monitoring (ingoing or outgoing traffic unknown).
3. Block botnet commanding messages traffic with firewall or IPS.

1.1.10 RAT

RAT (=remote access trojan) : category of trojan horses that sets up a backdoor and begins providing administrative level remote access to your system.
Properties :

- Often downloaded as a trojan.

Solutions :

1. Don't run unknown software
2. Antivirus up-to-date
3. Backup

1.1.11 Logic bomb

Logic bomb : type of malware waiting for a specific event to occur (e.g. a backup process). When it occurs the logic bomb usually delete or remove information from systems.
Solutions :

1. alerts when a change has occured in the system
2. constant auditing of the system to ensure nothing has changed in the OS

1.1.12 Backdoor

Backdoor : a functionality unknown from the user and giving a secrete access to a software. The introduction of a backdoor in a software transforms it in a trojan.

1.2 Compare and contrast types of attacks

1.2.1 Social engineering : Phishing

Phishing : technique used to try to convince you to give personal informations.
Properties :

- Mix between social engineering and spoofing.

1.2.2 Social engineering : Spear phishing

Spear phishing : phishing with inside information to customize the content presented to the target.

1.2.3 Social engineering : Whaling

Whaling : spear phishing the CEO.

1.2.4 Social engineering : Vishing

Vishing (*voice phishing*) : phishing through phone.

1.2.5 Social engineering : Tailgating

Tailgating : when somebody uses someone else to access area.

1.2.6 Social engineering : Impersonation

Impersonation : pretend to be somebody you aren't.

1.2.7 Social engineering : Dumpster diving

Going into trash bin to look for informations. Usually no law prevent you doing this.

1.2.8 Social engineering : Shoulder surfing

... (shoulder = épaule)

1.2.9 Social engineering : Hoax

Warn somebody of a fake threat to take his money.

1.2.10 Social engineering : Watering hole attack

Attack somebody else you visit. For example attack coffee shop near organization's building.

1.2.11 Social engineering : Principles

- Authority : pretend to be somebody you are not (help desk/CEO/...)
- Intimidation : "bad things will happen if you do not help"
- Consensus : using what other people have done to justify what you are doing ("my coworker was able to do this")
- Scarcity : this situation will be this way for a limited amount of time only
- Familiarity : take advantage of the other's trust by being his friend
- Trust : make that the other guy trusts you
- Urgency : giving information is urgent

1.2.12 Application/service attacks : DoS

Causing a service to fail. Can be due to OS vulnerabilities, overwhelming service, turning off the power to a building. Example : take down DNS server to create your own and control people traffic :-)

1.2.13 Application/service attacks : DDoS

When the service is being denied because the attack is coming from so many places at the same time (ex : army of botnets).

1.2.14 Application/service attacks : Man-in-the-middle

...

1.2.15 Application/service attacks : Buffer overflow

When you're writing information to memory and it spills over past the allocated space that was originally set for the amount of data.

1.2.16 Application/service attacks : Injection

Inject code into the input of an application when this application is not performing appropriate checks.

Examples :

- **HTML** = HyperText Markup Language (hypertext = text with hyperlinks, markup = balise), SQL (most common), **XML** = Extensible Markup Language (to encode documents in a format both human and machine readable), **LDAP** = Lightweight Directory Access Protocol (to retrieve informations from LDAP databases).

1.2.17 Application/service attacks : Cross-site scripting

Vulnerability enabling attackers to inject client-side scripts into web pages viewed by other users.

Examples :

- **Non-persistent XSS** (also called reflected XSS) : you modify URL so that when someone clicks on it it executes nonexpected code.
- **Persistent XSS** : when the data provided by the attacker is saved by the server, and then permanently displayed on "normal" pages returned to other users in the course of regular browsing.

1.2.18 Application/service attacks : Cross-site request forgery

When you modify URL to make the one clicking on it perform unintended things on the website (like changing password). The difference with non-persistent XSS is that you trick the user instead of the web server.

1.2.19 Application/service attacks : Privilege escalation

Gain a higher level access to the system than what yout authentication allows.

1.2.20 Application/service attacks : ARP poisoning

ARP = Address Resolution Protocol : protocol converting an IP adress into a MAC adress).

MitM allowing an attacker to redirect communications in a network if it uses ARP (like Ethernet and Wi-Fi). The bad guy sends spoofed ARP messages to the router to say its MAC adress is associated with the IP of the victim. It is then between the user and the router.

1.2.21 Application/service attacks : Amplification

In a DDoS context : when botnets are passing through an intermediate service to amplifie their requests.

Example :

- You spoof a web server's IP, sending domain name queries to DNS which responds to the web server (=DNS reflection) with a bigger packet (=DNS amplification).

1.2.22 Application/service attacks : DNS poisoning

Also called DNS spoofing. When you corrupt Domain Name System by introducing data into the DNS resolver's cache, causing the name server to return an incorrect IP address. This results in traffic being diverted to the attacker's computer (or any other computer).

1.2.23 Application/service attacks : Domain hijacking

Act of changing the registration of a domain name without the permission of its original registrant.

1.2.24 Application/service attacks : Man-in-the-browser

Proxy trojan horse (= Trojan used to hide the source of malicious activity) that infects a web browser to modify web pages, modify transaction content or insert additional transactions, all in a completely covert fashion invisible to both the user and host web application.

Examples :

- hidden in web browser extensions

1.2.25 Application/service attacks : Zero day

...

1.2.26 Application/service attacks : Replay

Type of MitM attack where you reuse data u spied on user in an ulterior attack.

1.2.27 Application/service attacks : Pass the hash

Technique allowing an attacker to authenticate to a remote server or service by using the underlying hash of a user's password, instead of requiring the associated plaintext password as is normally the case.

It exploits an implementation weakness in the authentication protocol, where password hash remain static from session to session until the password is next changed.

1.2.28 Application/service attacks : Hijacking and related attacks

- **Clickjacking** (détournemen de click) : tricking a web user into clicking on something different from what the user perceives they are clicking.
- **Session hijacking** (or "cookie hijacking") : exploitation of a valid computer session to gain unauthorized access to information or services in a computer system.

  User's cookie is stolen by an attacker using an intermediary computer or with access to the saved cookies on the victim's computer.
- **URL hijacking** : Trick the user by making him visit a different URL than what he intented to.
- **Typo squatting** : Type of URL hijacking that exploit typos made by users into URL bar to redirect on fake website.

1.2.29 Application/service attacks : Driver manipulation

Driver = program that operates or controls a particular type of device that is attached to a computer.

- **Shimming** (shim = cale : comme entre une porte et une charnière) (shimming = filling in the space between to objects) : for example pretend that the application is using an older and vulnerable version of Windows (Windows has its own shim for retro compatibility with other versions).
- **Refactoring** (= remaniement) : (meta/polymorphic malware) : each time malware is downloaded, its downloaded as a different executable to avoid antivirus signatures check.

1.2.30 Application/service attacks : MAC spoofing

...

Spoofing = when a device pretends to be something it's not (fake web server, fake DNS server, email adress looking like somebody, fake phone number).

1.2.31 Application/service attacks : IP spoofing

Used for DNS amplification or ARP poisoning.

Easier to prevent than MAC spoofing.

1.2.32 Wireless attacks : Replay

...

Wireless ⇒ easier to capture data than wired network.

WEP = Wired Equivalent Privacy (not used anymore).

1.2.33 Wireless attacks : IV

Cracking WEP requires thousands of **IV** (= Initial Vector) packets.

1.2.34 Wireless attacks : Evil twin

Wireless Evil Twin : buy a wireless access point and configure it the same way as the existing network ⇒ users connect to the evil twin instead of the original wireless network.

Solution :

1. Make sure u use HTTPS and VPN to encrypt your communications.

1.2.35 Wireless attacks : Rogue AP

= Rogue access point = point d'accès non autorisé.

1.2.36 Wireless attacks : Jamming

Goal : transmit interfering wireless signals to make a DoS.

Reactive jamming : jam only when someone else tries to communicate.

1.2.37 Wireless attacks : WPS

Wi-Fi Protected Setup.

WPS has a design flow : the authentication PIN is an eight-digit number = 7 digits + a check-sum. It is validated in two parts : first 4 digits then 3 others ⇒ $10,000$ possibilities and $1,000$ possibilities = 4 hours to go through all of them.

Pixie Dust (= other flaw) : receive an encrypted message and then crack it offline

1.2.38 Wireless attacks : Bluejacking

Sending of unsolicited messages to a mobile device over a bluetooth connection.

Bluetooth = ∼ 10 meters.

1.2.39 Wireless attacks : Bluesnarfing

snarfing = copie.

Access a bluetooth device and transfer data.

1.2.40 Wireless attacks : RFID

= Radio-Frequency identification.
We power the RFID chip by sending radio-frequency messages to active the tag.
Examples :

- Data capture and replay attack.
- Spoof the reader
- Signal jamming
- Decrypt communication (many default keys are on google)

1.2.41 Wireless attacks : NFC

= Near field communication (like MasterCard paying with phone close to bancontact machine).

- Same security problems than for RFID.

1.2.42 Wireless attacks : Disassociation

It is a DoS in which you disconnect somebody from its wireless connexion.
Based on management frames used in the 802.11 protocol that are sent in clear.

1.2.43 Cryptographic attacks : Birthday

... Birthday paradox ⇒

1.2.44 Cryptographic attacks : Known plain text/cipher text

When the attacker has the encrypted information and a piece of the corresponding plaintext (called the **crib**).

1.2.45 Cryptographic attacks : Rainbow tables

Table in which you have stored all possible hashes of a particular hashing algorithm (very useful for long passwords).
Solutions :

1. salt

1.2.46 Cryptographic attacks : Dictionary

Start by testing common passwords.

1.2.47 Cryptographic attacks : Brute force

Difficult online.

1.2.48 Cryptographic attacks : Online vs. offline

...

1.2.49 Cryptographic attacks : Collision

Attacks integrity.
Solutions :

1. use larger hashes

You can create a fake certification authority with a hash collision.

1.2.50 Cryptographic attacks : Downgrade

= force the systems to downgrade their security.

1.2.51 Cryptographic attacks : Replay

Example :

- Alice sends hashed password to Bob but Oscar sniff it and sends it too

1.2.52 Cryptographic attacks : Weak implementations

Example :

- WEP
- DES 56 bits

1.3 Explain threat actor types and attributes

1.3.1 Types of actors : Script kiddies

...

1.3.2 Types of actors : Hacktivist

...

1.3.3 Types of actors : Organized crime

...

1.3.4 Types of actors : Nation states/APT

APT = Advanced Persistent Threat = constant attacks.

1.3.5 Types of actors : Insiders

...

1.3.6 Types of actors : Competitors

DoS, espionage, harm reputation

1.3.7 Attributes of actors : Internal/external

...

1.3.8 Attributes of actors : Level of sophistication

...

1.3.9 Attributes of actors : Resources/funding

...

1.3.10 Attributes of actors : Intent/motivation

...

1.3.11 Use of open-source intelligence

...

1.4 Explain penetration testing concepts

1.4.1 Active reconnaissance

Usually, this is done with a vulnerability scan (ping scans, ports scan, service scan,...)

1.4.2 Passive reconnaissance

Occurs before active reconnaissance. This is where you're not touching any of the equipment. You're instead trying to gather as much information as you can from sources that are already available. information gathering phase where you're not touching any of the equipment. You're instead trying to gather as much information as you can from sources that are already available.

1.4.3 Pivot

A penetration tester can be able to use a compromised host as a bridge to pivot to another network or system that is not directly accessible from his attacking system.

1.4.4 Initial exploitation

Usually there is an initial exploitation where a vulnerability is taken advantage of. Someone gets into the network, or gets into a system, and that is usually the hardest part. Once you get past that first initial exploitation, things tend to be a little bit easier.

1.4.5 Persistence

Maintain access with backdoor.

1.4.6 Escalation of privilege

...

1.4.7 Black box

...

1.4.8 White box

...

1.4.9 Gray box

...

1.4.10 Penetration testing vs. vulnerability scanning

Vulnerability scanning : less invasive, you do not penetrate the system. Port scan/identifying system. (penetration testing = actually exploit vulnerabilities you found)

1.5 Explain vulnerability scanning concepts

1.5.1 Passively test security controls

...

1.5.2 Identify vulnerability

...

1.5.3 Identify lack of security controls

...

1.5.4 Identify common misconfigurations

... (réseau mal configuré)

1.5.5 Intrusive vs. non-intrusive

Non-intrusive : we're simply gathering information about what we're seeing on the network.

Example

- Simple packet capture process, and then looking through those packets to determine what conversation might be going on.

Intrusive : checking a vulnerability to see if it exists without actually taking advantage of that vulnerability.

1.5.6 Credentialed vs. non-credentialed

Credentialed : you provide the scanner with a legitimate username and password, it uses that to get into the system as a normal user, and then tries to find ways to get around the existing security **Non-credentialed** : you don't have any access to the server you are scanning

Example :

- you don't have a username and password that you could use to authenticate.

1.5.7 False positive

When the vulnerability scanner tells you that there is indeed a vulnerability on your system, but the reality is that the vulnerability is not one that could be exploited.

1.6 Explain the impact associated with types of vulnerabilities

1.6.1 Race conditions

...

1.6.2 Vulnerabilities due to end-of-life systems

Occurs when a device or a component or a piece of software is no longer under support from the vendor

1.6.3 Vulnerabilities due to Embedded systems

Often connected to the internet ⇒ convenient to gain access to the system.

Usually running an outdated version of software, because not upgraded ⇒ could be vulnerable to new exploits.

1.6.4 Vulnerabilities due to Lack of vendor support

Vendor support is very important to be able to provide timely updates and security patches.

1.6.5 Improper input handling

...

1.6.6 Improper error handling

...

1.6.7 Misconfiguration/weak configuration

...

1.6.8 Default configuration

...

1.6.9 Resource exhaustion

Denial of service condition that happens when the resources required to execute an action are entirely expended (=dépensée), preventing that action from occurring.

1.6.10 Untrained users

...

1.6.11 Improperly configured accounts

When there might be an account that really isn't used for anything. Maybe it was set up originally for a particular use but is now no longer used.

You should make sure that those accounts are either disabled or that they're deleted from your systems.

1.6.12 Vulnerable business processes

Occurs when an organization doesn't have a set of checks and balances to be able to handle its most significant business processes.

Attackers infiltrate the enterprise and look for vulnerable practices, susceptible systems, or operational loopholes (= échappatoires). Once a weakness has been identified, a part of the process is altered to benefit the attacker, without the enterprise or its client detecting the change.

1.6.13 Weak cipher suites and implementations

Cipher suite = encryption protocol(s) + length of encryption key. Usually also includes a hash algorithm.

Examples :

- SSL or TLS (that has $\sim 300 \neq$ suites).

Use a strong ciphersuite (say bye to DES-56 or MD5) !

1.6.14 Memory/buffer vulnerability : Memory leak

Occurs when memory is allocated during the execution of a program and it is never unallocated when it's finished being used : as you use more and more of the application, you continue to use more and more of the memory.

It slowly begins to grow and use up all available memory and ultimately either crashes the application or it crashes the computer it's running on.

1.6.15 Memory/buffer vulnerability : Integer overflow

When you're trying to put a very large number into a place that has a very small allocated area $\Rightarrow$ manipulation of memory.

1.6.16 Memory/buffer vulnerability : Buffer overflow

Buffer : temporarily stores data while the data is the process of moving from one place to another or between processes within a computer.

Occurs when a certain amount of space has been allocated to store variables in application and this application allows you to store a variable that's larger than that allocated space, spills over into other memory areas, and potentially allows more access to the system than normally would be available.

The developers, need to be very careful about how they check what you're putting into the buffer used by the application.

1.6.17 Memory/buffer vulnerability : Pointer dereference

When the programmer is dereferencing a portion of memory that's being used by an application, except in this case there is nothing at that memory address to dereference and the application crashes.

1.6.18 Memory/buffer vulnerability : DLL injection

When the bad guys will their own libraries in place so that when the application references the library, they are effectively referencing the bad guys' code.

1.6.19 System sprawl/undocumented assets

(sprawl = s'étendre) **Virtualization sprawl** (/virtual machine sprawl/VM sprawl/virtual server sprawl) = phenomenon that occurs when the number of virtual machines (VMs) on a network reaches a point where the administrator can no longer manage them effectively. Virtualization sprawl may also be referred to as virtual machine sprawl, VM sprawl or virtual server sprawl.

1.6.20 Architecture/design weaknesses

...

1.6.21 New threats/zero day

...

1.6.22 Improper certificate and key management

There needs to be a formal process set up for managing certificates.

2 Technologies and tools

2.1 Install and configure network components, both hardware and software-based, to support organizational security

A **firewall** is simply looking at the traffic going by, comparing it to a list of access controls, and then either allowing or disallowing that traffic (= **Firewall rules**).

2.1.1 Firewall : ACL

= Access control list.

ACL usually does stateless inspection, where it doesn't know from which application a packet comes from nor the session while a firewall can know it.

2.1.2 Firewall : Application-based vs. network-based

Traditional **network-based** firewalls filter traffic by TCP or UDP port number. That's OSI layer 4, the transport layer of the OSI stack.

These days, next-generation firewalls and modern technologies allow us to filter based on the application. That allows us to filter all the way up to OSI layer 7.

2.1.3 Firewall : Stateful vs. stateless

cf. ACL

2.1.4 Firewall : Implicit deny

When a firewall denies something not because it is explicitely mentionned in its rules but because the traffic coming through the firewall doesn't match any rule.

2.1.5 VPN concentrator

VPN = Virtual Private Network in the sense that it acts like the source and the target of the communications are in the same private network by allows to set up an encrypted tunnel.

It is used to protect users' identities or to spoof a location (like a proxy does).

VPN concentrator = type of networking device that provides secure creation of VPN connections and delivery of messages between VPN nodes ; it is a type of router device, built specifically for creating and managing VPN communication infrastructures.

2.1.6 VPN concentrator : Remote access vs. site-to-site

A **site-to-site VPN** allows offices in multiple fixed locations to establish secure connections with each other over a public network such as the Internet.

Remote access VPN = VPN allowing an individual user to connect to a private business network from a remote location using a laptop or desktop computer connected to the Internet.

2.1.7 VPN concentrator : IPSec

= Internet Protocol Security : allos Layer 3 encryption of all IP traffic from one site to the other ⇒ confidentiality. But IPSec also provides integrity check to ensure nobody is replaying traffic through the VPN connexion.

IPSec is a very standardized protocol ⇒ you can have one manufacturer's firewall at one side and a completely different manufacturer's firewall at the other side, but they'll still be able to communicate using IPSec.

Two core protocols are associated with IPSec : **AH** (= authentication header) and **ESP** (= Encapsulation security payload).

- **Tunnel mode** : both the IP header and the data are encrypted. They're wrapped around an IPSec header and an IPSec trailer, and then a completely different IP header is put on the front of the packet. This means that, if somebody sees that packet going through, they're not going to have any idea what the actual IP destination is because all of that information is encrypted when you're using tunnel mode.
- **Transport mode** : = mode of IPSec in which th

  In contrast in the **IPSec Tunnel mode**, both the IP header and the data are encrypted ⇒ if somebody sees the packet he has no idea what the actual IP destination is because this information is encrypted.
- **AH** : = Authentication header, provides integrity of the data that is being sent through the network.

  Commonly, IPSec will take the IP header and the data, combine that with a shared key, and provide a hash. And usually, the has is one based on MD5, SHA-1, or SHA-2, and it's adding that authentication header to the beginning of the packet.

  AH is one of the two core protocols associated with IPSec.
- **ESP** : = Encapsulation Security Payload

  ESP is one of the two core protocols associated with IPSec.

  using 3DES or usually AES for encryption, and it adds a header, a trailer, and an Integrity Check Value. That means that you can encrypt the IP header, the data, and you have an ESP trailer inside of this encrypted information. And on the outside, you have not only your new IP header, but the ESP header and Integrity Check Value. This means that you can authenticate almost all of the data when you're running this IPSec datagram and using ESP to encrypt the data.

  In most IPSec implementations, you're not only using the ESP for the encryption, but you're using the authentication header at the same time. This means that you can have this encrypted data inside of your packet, but you can authenticate the entire IP packet. That means that you can do this either in a transport mode and a tunnel mode to ensure that not only is your traffic protected and encrypted, but now you can also be assured that's exactly what was sent by the original station.

2.1.8 VPN concentrator : Split tunnel vs. full tunnel

Full tunnel : all traffic, regardless of its destination, will all traverse the secure tunnel.

Split tunnel : when all of the traffic from your site to the corporate network traverses this encrypted tunnel, but if you need to communicate to a third-party website that is not part of your corporate network, it will use the normal communication outside the scope of that VPN communication.

2.1.9 VPN concentrator : TLS

The most common type of VPNS are SSL VPNs that are using SSL or TLS protocol **running over TCP port 443**.

Most SSL VPN clients are built into existing browsers or operating systems, and you're usually logging in with your normal authentication.

2.1.10 VPN concentrator : Always-on VPN

Some software can be configured as Always On, which means anytime you're using your laptop, it's always using an encrypted tunnel back to your corporate network.

Examples :

- HTTPS Everywhere chrome extension

2.1.11 NIPS/NIDS

(N)IPS = (Network) Intrusion Prevention System : block the malicious traffic.

(N)IDS = (Network) Intrusion Detection System : inform you that an exploit against an operating system on a network occured (but doesn't stop any traffic).

2.1.12 NIPS/NIDS : Signature-based

When a signature is predefined inside of the IPS and it's watching for traffic to traverse the network that matches this signature exactly.

If it identifies traffic that matches exactly what we're seeing, it will block that traffic at the IPS.

2.1.13 NIPS/NIDS : Heuristic/behavioral

Used by some of the more advanced IPS : can identify attacks based on heuristics.

The IPS could be configured with a set of characteristics that might define an attack. As traffic is coming through, the heuristics can then examine that traffic and make a determination on if an attack is taking place or not.

2.1.14 NIPS/NIDS : Anomaly

Your IPS will sit on the network and begin to understand what a normal traffic flow is for your network. If any traffic comes through that doesn't match the normal flow of traffic, the anomaly based identification will block it at the IPS.

2.1.15 NIPS/NIDS : Inline vs. passive

Passive : the IPS receives a copy of the traffic and is able to make a decision on what to do once that information is received. Since it is a passive monitor it is not in the middle of the communication and cannot block the traffic. The only possibility to block the traffic in passive mode is to send an **out-of-band** (because the IPS is not part of the traffic flow) response : a **TCP reset frame** to both the source and the destination of the communication which will close the communication untill another traffic flow between the two devices is set up. Doing that you are hoping that u ended the communication before much of the malicious state is able to traverse the network.

Note that UDP does not allow to perform a reset : there is no way to stop a commuinication in an out-of-band mode if the UDP protocol is used.

Inline (monitoring) : when all traffic pass through the IPS which makes a decision on allowing it through the network or not. Since it sits inline, the response to any malicious traffic will be to immediately drop it.

2.1.16 NIPS/NIDS : In-band vs. out-of-band

It looks that it is the same as in-line vs out-line.

2.1.17 NIPS/NIDS : Rules

An IPS makes the decision on what vulnerabilities to look for and what to do if a vulnerability is found based on a series of rules. Usually thousands of different rules that are grouped together by different characteristics and you can make some broad settings to say anything that's a database injection, you may want to block. An IPS is difficult to configure.

2.1.18 NIPS/NIDS : Analytics

- **False positive** : when the system has told us that there has been an intrusion onto the network but in reality it's a case of mistaken identity and there was not an intrusion at all.
- **False negative** : when malicious traffic came through the IPS but the IPS did not identify it as malicious.

2.1.19 Router

= device that forwards traffic between different IP subnets (= subnetworks) that may have different network types (copper(=cuivre)/fiber/ethernet/...).

Routers are considered to be layer 3 devices. That means they make their routing decision at the network layer of the OSI model. If there's a router inside of a switch, you'll sometimes hear these referred to as layer 3 switches.

2.1.20 Router : ACLs

= access control list : one of the capabilities built into a router, it is used to allow or deny traffic, to evaluate traffic very similarly to what a firewall might do.

An ACL could evaluate a source IP address, a destination IP address, a port number that might be in use, and then decide whether to allow or deny that traffic through the router. And like a firewall, there's usually a list of rules for an access control list. And the router will follow that list until that traffic matches one of the rules in the access control list.

2.1.21 Router : Antispoofing

Can be done by filtering out any IP address ranges that should not be flowing through the firewall. Also, You can also configure your router with **RPF** (= reverse path forwarding) to prevent IP spoofing : the router will try to find the reverse route of the packet in its routing table. If a reverse route is not found on the interface where the packet arrived, it means that the packed is spoofed and it is immediately dropped.

Examples :

- **RFC1918** (= standard defining how private IP adresses should look like) IP adresses should not be routed to the internet.

2.1.22 Switch

= device that is effectively bridging traffic in hardware. They're using an application specific integrated circuit (= **ASIC**) to do this very quickly in the hardware of these devices.

2.1.23 Switch : Port security

NAC = Network Access Control, sometimes referred as **IEEE 802.1X** : requires that someone provide a username and a password and authenticate before they are able to gain access to any of the switch interfaces.

We're really talking about port-based network access control because it is the physical interface, or the port on the switch where we're providing the security.

2.1.24 Switch : Layer 2 vs. Layer 3

We often refer to a switch as an OSI layer 2 device, because it's making its forwarding decision based on the MAC address or the layer 2 address of the traffic going through the switch.

we can find switches that have routing capability enabled in them as well. And we commonly refer to these as layer 3 switches.

The switching is still operating at layer 2, making its forwarding decisions based on MAC address, but you can also configure interfaces to act as routed interfaces that would forward traffic based on the layer 3 IP configuration. This isn't changing the way that switching works, and it's not changing the way that routing operates. It's simply combining both a switch, and a router within the same physical device.

2.1.25 Switch : Loop prevention

If you connect two switches to each other, the packets will rotate through those switches until you break that connection.

This can bring down a network very, very quickly. As more people put traffic onto the network, more and more traffic will begin to loop around, and you could bring down a network in a matter of seconds.

Solution :

1. **spanning-tree protocol** automatically identifies a loop and prevents a loop from occurring on a switch network. You may see this referred to as **IEEE 802.1D**. If a connection between Network A and bridge six suddenly becomes unavailable, spanning-tree will recognize the change, and it will change bridge configuration in bridge five and bridge 11 to now allow traffic to traverse the other direction around the problem. So spanning-tree is not only making sure that the network is available, its preventing any loops and downtime from occurring on the network as well.

2.1.26 Switch : Flood guard

Switches maintain a large list of MAC addresses that are associated with the interfaces that it sees communicating on the switch. If you are able to flood the network with MAC addresses, you would very quickly overflow that index of addresses, causing a denial of service.

Solution :

1. **Flood guard** : it configures a maximum number of MAC addresses that could possibly be seen on any particular interface. You get to define how many MAC addresses is appropriate for a particular interface to prevent anyone from overloading the number of MAC addresses on this network.

2.1.27 Proxy

= device that sits between your users and usually the internet to help filter and protect them from the internet communication.

A proxy is able to provide security capabilities. Not only is it able to cache information to make your network communication more efficient, it's able to provide access control. A proxy could perform URL filtering (the user is requesting a URL to a site that they are not allowed to visit, the proxy will immediately send back a response saying that you don't have permission to visit that URL), look for viruses inside of the network communication, and much more.

2.1.28 Proxy : Forward and reverse proxy

A **forward proxy** is used internally to protect users from the internet. The proxy will analyze the response from internet and make sure that everything in that response is legitimate and secure, and then send that response off to the user.

A **reverse proxy** is used to protect internal services like web servers. This kind of proxy retrieves resources on behalf of a client from one or more servers that are then returned to the client as if they originated from the Web server itself.

Reverse proxies can hide the existence and characteristics of an origin server or servers.

2.1.29 Proxy : Transparent

The end users have no idea there's a proxy in the middle, and no additional configuration needs to occur on the operating system to be able to take advantage of the proxy. A **Transparent/open proxy** = proxy that has been set up and configured by a third party that you have no knowledge of. Open proxies are commonly used to circumvent existing security controls. So if a user inside of your network can't visit a particular URL because there is URL filtering, they will instead visit the proxy and tell the proxy to visit that URL on their behalf, thereby going around the URL filtering that you have on your network.

2.1.30 Proxy : Application/multipurpose

Application proxy : the proxy itself understands the way applications operate so that it's able to take a request for an application and proxy that request on the user's behalf. Some proxies may only know one type of application. They may be able to take HTTP or browser requests and proxy them on behalf of the user. Other proxies are more advanced and are able to use many different kinds of applications.

Multipurpose proxy : can be used with various protocols/ OS and uses the NAT method.

NAT = Network Adress Translation = method of remapping one IP address space into another by modifying network address information in IP header of packets while they are in transit across a traffic routing device). It has become a popular and essential tool in conserving global address space in the face of IPv4 address exhaustion. One Internet-routable IP address of a NAT gateway can be used for an entire private network. **IP masquerading** is a technique that hides an entire IP address space, usually consisting of private IP addresses, behind a single IP address in another, usually public address space. The address that has to be hidden is changed into a single (public) IP address as "new" source address of the outgoing IP packet so it appears as originating not from the hidden host but from the routing device itself. Because of the popularity of this technique to conserve IPv4 address space, the term NAT has become virtually synonymous with IP masquerading.

2.1.31 Load balancer

A **load balancer** is designed to take a load of traffic and distribute it across multiple resources without the end-user even realizing that it's occurring.

Used for :

- Load balancing : you can have many web servers configured and have everyone visit one single URL. But in reality, their load may be distributed across multiple internal web servers.
- Providing fault tolerance : if a particular web server goes down it will redirect your request to another available web server.
- Offload the encryption process : decrypt the SSL encryption before forwarding the request to the web server.
- **Caching** : requested information can be cached locally so that next time it is done you don't have to make the request down to the web server.
- Configuring quality of service : some applications may get priority over others, even if they may all be going to the same web servers.

2.1.32 Load balancer : Scheduling

- **Affinity** : The load balancer will always use the same server for a particular user or a particular application instance or session. Used when everything that occurs in that application should be occurring on the same web server.
- **Round-robin** : Each server is selected in turn : A then B then C then A then B then C then ...

2.1.33 Load balancer : Active-passive

Some servers will be currently active and able to take requests, and other servers are on standby. If an active web server fails, the load balancer will identify the failure and begin using one of the standby servers in its place.

2.1.34 Load balancer : Active-active

In active-active mode, the load balancer is going to monitor the load that's occurring on these different servers. And if one server is more loaded than the others, it will use some of the other servers first.

That means that all of these servers are active. And requests coming through the load balancer could use any of these active servers at any time.

2.1.35 Load balancer : Virtual IPs

= **VIP**s. This is an IP address that doesn't correspond to an actual physical network interface. Used by the load balancer that will then redirect the request at this IP adress to the web servers that have different IP addresses.

2.1.36 Access point : SSID

2.1.37 Access point : MAC filtering

2.1.38 Access point : Signal strength

2.1.39 Access point : Band selection/width

2.1.40 Access point : Antenna types and placement

2.1.41 Access point : Fat vs. thin

2.1.42 Access point : Controller-based vs. standalone

2.1.43 SIEM: Aggregation

2.1.44 SIEM: Correlation

2.1.45 SIEM: Automated alerting and triggers

2.1.46 SIEM: Time synchronization

2.1.47 SIEM: Event deduplication

2.1.48 SIEM: Logs/WORM

2.1.49 DLP : USB blocking

2.1.50 DLP : Cloud-based

2.1.51 DLP : Email

2.1.52 NAC : Dissolvable vs. permanent

2.1.53 NAC : Host health checks

2.1.54 NAC : Agent vs. agentless

2.1.55 Mail gateway : Spam filter

2.1.56 Mail gateway : DLP

2.1.57 Mail gateway : Encryption

2.1.58 Bridge

2.1.59 SSL/TLS accelerators

2.1.60 SSL decryptors

2.1.61 Media gateway

2.1.62 Hardware security module

2.2 Given a scenario, use appropriate software tools to assess the security posture of an organization

2.2.1 Protocol analyzer

2.2.2 Network scanners : Rogue system detection

2.2.3 Network scanners : Network mapping

2.2.4 Wireless scanners/cracker

2.2.5 Password cracker

2.2.6 Vulnerability scanner

2.2.7 Configuration compliance scanner

2.2.8 Exploitation frameworks

3 Architecture and Design

3.1 Explain use cases and purpose for frameworks, best practices and secure configuration guides.

3.1.1 Industry-standard frameworks and reference architectures : Regulatory

3.1.2 Industry-standard frameworks and reference architectures : Non-regulatory

3.1.3 Industry-standard frameworks and reference architectures : National vs. international

3.1.4 Industry-standard frameworks and reference architectures : Industry-specific frameworks

3.1.5 Benchmarks/secure configuration guides : Platform/vendor-specific guides

- **Web server**
- **Operating system**
- **Application server**
- **Network infrastructure devices**

3.1.6 Benchmarks/secure configuration guides : General purpose guides

3.1.7 Defense-in-depth/layered security : Vendor diversity

3.1.8 Defense-in-depth/layered security : Control diversity

- **Administrative** :
- **Technical** :

3.1.9 Defense-in-depth/layered security : User training

3.2 Given a scenario, implement secure network architecture concepts.

3.2.1 Zones/topologies : DMZ

3.2.2 Zones/topologies : Extranet

3.2.3 Zones/topologies : Intranet

3.2.4 Zones/topologies : Wireless

3.2.5 Zones/topologies : Guest

3.2.6 Zones/topologies : Honeynets

3.2.7 Zones/topologies : NAT

3.2.8 Zones/topologies : Ad hoc

3.2.9 Segregation/segmentation/isolation : Physical

3.2.10 Segregation/segmentation/isolation : Logical (VLAN)

3.2.11 Segregation/segmentation/isolation : Virtualization

3.2.12 Segregation/segmentation/isolation : Air gaps

3.2.13 Tunneling/VPN : Site-to-site

3.2.14 Tunneling/VPN : Remote access

3.2.15 Security device/technology placement : Sensors

3.2.16 Security device/technology placement : Collectors

3.2.17 Security device/technology placement : Correlation engines

3.2.18 Security device/technology placement : Filters

3.2.19 Security device/technology placement : Proxies

3.2.20 Security device/technology placement : Firewalls

3.2.21 Security device/technology placement : VPN concentrators

3.2.22 Security device/technology placement : SSL accelerators

3.2.23 Security device/technology placement : Load balancers

3.2.24 Security device/technology placement : DDoS mitigator

3.2.25 Security device/technology placement : Aggregation switches

3.2.26 Security device/technology placement : Taps and port mirror

3.2.27 SDN

3.3 Given a scenario, implement secure systems design.

3.3.1 Hardware/firmware security : FDE/SED

3.3.2 Hardware/firmware security : TPM

3.3.3 Hardware/firmware security : HSM

3.3.4 Hardware/firmware security : UEFI/BIOS

3.3.5 Hardware/firmware security : Secure boot and attestation

3.3.6 Hardware/firmware security : Supply chain

- **Server**
- **Workstation**
- **Appliance**
- **Kiosk**
- **Mobile OS**

3.3.10 Operating systems : Patch management

3.3.11 Operating systems : Disabling unnecessary ports and services

3.3.12 Operating systems : Least functionality

3.3.13 Operating systems : Secure configurations

3.3.14 Operating systems : Trusted operating system

3.3.15 Operating systems : Application whitelisting/blacklisting

3.3.16 Operating systems : Disable default accounts/passwords

3.3.17 Peripherals : Wireless keyboards

3.3.18 Peripherals : Wireless mice

3.3.19 Peripherals : Displays

3.3.20 Peripherals : WiFi-enabled MicroSD cards

3.3.21 Peripherals : Printers/MFDs

3.3.22 Peripherals : External storage devices

3.3.23 Peripherals : Digital cameras

3.4 Explain the importance of secure staging deployment concepts.

3.4.1 Sandboxing

3.4.2 Environment : Development

3.4.3 Environment : Test

3.4.4 Environment : Staging

3.4.5 Environment : Production

3.4.6 Secure baseline

3.4.7 Integrity measurement :

3.5 Explain the security implications of embedded systems.

3.5.1 SCADA/ICS

3.5.2 Smart devices/IoT : Wearable technology

3.5.3 Smart devices/IoT : Home automation

3.5.4 HVAC

3.5.5 SoC

3.5.6 RTOS

3.5.7 Printers/MFDs

3.5.8 Camera systems

3.5.9 Special purpose : Medical devices

3.5.10 Special purpose : Vehicles

3.5.11 Special purpose : Aircraft/UAV

3.6 Summarize secure application development and deployment concepts.

3.6.1 Development life-cycle models : Waterfall vs. Agile

4 Identity and Access Management

4.1 Compare and contrast identity and access management concepts

4.1.1 Identification, authentication, authorization and accounting (AAA)

4.1.2 Multifactor authentication : Something you are

4.1.3 Multifactor authentication : Something you have

4.1.4 Multifactor authentication : Something you know

4.1.5 Multifactor authentication : Somewhere you are

4.1.6 Multifactor authentication : Something you do

4.1.7 Federation

4.1.8 Single sign-on

4.1.9 Transitive trust

4.2 Given a scenario, install and configure identity and access services.

4.2.1 LDAP

4.2.2 Kerberos

4.2.3 TACACS+

4.2.4 CHAP

4.2.5 PAP

4.2.6 MSCHAP

4.2.7 RADIUS

4.2.8 SAML

4.2.9 OpenID Connect

4.2.10 OAUTH

4.2.11 Shibboleth

4.2.12 Secure token

4.2.13 NTLM

4.3 Given a scenario, implement identity and access management controls.

4.3.1 Access control models : MAC

4.3.2 Access control models : DAC

4.3.3 Access control models : ABAC

4.3.4 Access control models : Role-based access control

4.3.5 Access control models : Rule-based access control

4.3.6 Physical access control : Proximity cards

4.3.7 Physical access control : Smart cards

4.3.8 Biometric factors : Fingerprint scanner

4.3.9 Biometric factors : Retinal scanner

4.3.10 Biometric factors : Iris scanner

5 Risk Management

5.1 Explain the importance of policies, plans and procedures related to organizational security.

5.1.1 Standard operating procedure

5.1.2 Agreement types : BPA

5.1.3 Agreement types : SLA

5.1.4 Agreement types : ISA

5.1.5 Agreement types : MOU/MOA

5.1.6 Personnel management : Mandatory vacations

5.1.7 Personnel management : Job rotation

5.1.8 Personnel management : Separation of duties

5.1.9 Personnel management : Clean desk

5.1.10 Personnel management : Background checks

5.1.11 Personnel management : Exit interviews

5.1.12 Personnel management : Role-based awareness training

- Data owner :
- System administrator :
- System owner :
- User :
- Privileged user :
- Executive user :

5.1.13 Personnel management : NDA

5.1.14 Personnel management : Onboarding

5.1.15 Personnel management : Continuing education

5.1.16 Personnel management : Acceptable use policy/rules of behavior

5.1.17 Personnel management : Adverse actions

5.1.18 General security policies : Social media networks/applications

5.1.19 General security policies : Personal email

5.2 Summarize business impact analysis concepts.

5.2.1 RTO/RPO

5.2.2 MTBF

5.2.3 MTTR

5.2.4 Mission-essential functions

5.2.5 Identification of critical systems

5.2.6 Single point of failure

5.2.7 Impact : Life

5.2.8 Impact : Property

5.2.9 Impact : Safety

5.2.10 Impact : Finance

5.2.11 Impact : Reputation

5.2.12 Privacy impact assessment

5.2.13 Privacy threshold assessment

5.3 Explain risk management processes and concepts.

5.3.1 Threat assessment : Environmental

5.3.2 Threat assessment : Manmade

5.3.3 Threat assessment : Internal vs. external

5.3.4 Risk assessment : SLE

5.3.5 Risk assessment : ALE

5.3.6 Risk assessment : ARO

5.3.7 Risk assessment : Asset value

5.3.8 Risk assessment : Risk register

5.3.9 Risk assessment : Likelihood of occurrence

5.3.10 Risk assessment : Supply chain assessment

5.3.11 Risk assessment : Impact

5.3.12 Risk assessment : Quantitative

5.3.13 Risk assessment : Qualitative

5.3.14 Risk assessment : Testing

- Vulnerability testing authorization :

5.3.15 Risk assessment : Risk response techniques

- Accept :
- Transfer :
- Avoid :
- Mitigate :

5.3.16 Change management

5.4 Given a scenario, follow incident response procedures.

5.4.1 Incident response plan : Documented incident types/category definitions

5.4.2 Incident response plan : Roles and responsibilities

5.4.3 Incident response plan : Reporting requirements/escalation

5.4.4 Incident response plan : Cyber-incident response teams

5.4.5 Incident response plan : Exercise

5.4.6 Incident response process : Preparation

5.4.7 Incident response process : Identification

5.4.8 Incident response process : Containment

5.4.9 Incident response process : Eradication

5.4.10 Incident response process : Recovery

5.4.11 Incident response process : Lessons learned

5.5 Summarize basic concepts of forensics.

5.5.1 Order of volatility

5.5.2 Chain of custody

5.5.3 Legal hold

5.5.4 Data acquisition : Capture system image

5.5.5 Data acquisition : Network traffic and logs

5.5.6 Data acquisition : Capture video

5.5.7 Data acquisition : Record time offset

5.5.8 Data acquisition : Take hashes

5.5.9 Data acquisition : Screenshots

5.5.10 Data acquisition : Witness interviews

5.5.11 Preservation

5.5.12 Recovery

5.5.13 Strategic intelligence/counterintelligence gathering : Active logging

5.5.14 Track man-hours

5.6 Explain disaster recovery and continuity of operation concepts.

5.6.1 Recovery sites : Hot site

5.6.2 Recovery sites : Warm site

5.6.3 Recovery sites : Cold site

5.6.4 Order of restoration

5.6.5 Backup concepts : Differential

5.6.6 Backup concepts : Incremental

5.6.7 Backup concepts : Snapshots

6 Cryptography and PKI

6.1 Compare and contrast basic concepts of cryptography.

6.1.1 Symmetric algorithms

6.1.2 Modes of operation

6.1.3 Asymmetric algorithms

6.1.4 Hashing

6.1.5 Salt, IV, nonce

6.1.6 Elliptic curve

6.1.7 Weak/deprecated algorithms

6.1.8 Key exchange

6.1.9 Digital signatures

6.1.10 Diffusion

6.1.11 Confusion

6.1.12 Collision

6.1.13 Steganography

6.1.14 Obfuscation

6.1.15 Stream vs. block

6.1.16 Key strength

6.1.17 Session keys

6.1.18 Ephemeral key

6.1.19 Secret algorithm

6.1.20 Data-in-transit

6.1.21 Data-at-rest

6.1.22 Data-in-use

6.1.23 Random/pseudo-random number generation

6.1.24 Key stretching

6.1.25 Implementation vs. algorithm selection : Crypto service provider

6.1.26 Implementation vs. algorithm selection : Crypto modules

6.1.27 Perfect forward secrecy

6.1.28 Security through obscurity

6.1.29 Common use cases : Low power devices

6.1.30 Common use cases : Low latency

6.1.31 Common use cases : High resiliency

6.1.32 Common use cases : Supporting confidentiality

6.1.33 Common use cases : Supporting integrity

6.1.34 Common use cases : Supporting obfuscation

- DHE :
- ECDHE :

6.2.14 Asymmetric algorithms : Elliptic curve

6.2.15 Asymmetric algorithms : PGP/GPG

6.2.16 Hashing algorithms : MD5

6.2.17 Hashing algorithms : SHA

6.2.18 Hashing algorithms : HMAC

6.2.19 Hashing algorithms : RIPEMD

6.2.20 Key stretching algorithms : BCRYPT

6.2.21 Key stretching algorithms : PBKDF2

6.2.22 Obfuscation : XOR

6.2.23 Obfuscation : ROT13

6.2.24 Obfuscation : Substitution ciphers

6.3 Given a scenario, install and configure wireless security settings.

6.3.1 Cryptographic protocols : WPA

6.3.2 Cryptographic protocols : WPA2

6.3.3 Cryptographic protocols : CCMP

6.3.4 Cryptographic protocols : TKIP

6.3.5 Authentication protocols : EAP

6.3.6 Authentication protocols : PEAP

6.3.7 Authentication protocols : EAP-FAST

6.3.8 Authentication protocols : EAP-TLS

6.3.9 Authentication protocols : EAP-TTLS

6.3.10 Authentication protocols : IEEE 802.1x

6.3.11 Authentication protocols : RADIUS Federation

6.3.12 Methods : PSK vs. Enterprise vs. Open

6.3.13 Methods : WPS

6.3.14 Methods : Captive portals

6.4 Given a scenario, implement public key infrastructure.

6.4.1 Components : CA

6.4.2 Components : Intermediate CA

6.4.3 Components : CRL

6.4.4 Components : OCSP

6.4.5 Components : CSR

6.4.6 Components : Certificate

6.4.7 Components : Public key

6.4.8 Components : Private key

6.4.9 Components : Object identifiers (OID)

3DES : Triple Digital Encryption Standard

AAA : Authentication, Authorization, and Accounting

ABAC : Attribute-based Access Control

ACL : Access Control List

AES : Advanced Encryption Standard

AES256 : Advanced Encryption Standards 256bit

AH : Authentication Header

ALE : Annualized Loss Expectancy

AP : Access Point

API : Application Programming Interface

DNAT : Destination Network Address Transaction

DNS : Domain Name Service (Server)

DoS : Denial of Service

DRP : Disaster Recovery Plan

DSA : Digital Signature Algorithm

DSL : Digital Subscriber Line

DSU : Data Service Unit

EAP : Extensible Authentication Protocol

ECB : Electronic Code Book

ECC : Elliptic Curve Cryptography

ECDHE : Elliptic Curve Diffie-Hellman Ephemeral

ECDSA : Elliptic Curve Digital Signature Algorithm

EFS : Encrypted File System

EMI : Electromagnetic Interference

EMP : Electro Magnetic Pulse

ERP : Enterprise Resource Planning

ESN : Electronic Serial Number

ESP : Encapsulated Security Payload

EF : Exposure Factor

FACL : File System Access Control List

FAR : False Acceptance Rate

FDE : Full Disk Encryption

FRR : False Rejection Rate

FTP : File Transfer Protocol

FTPS : Secured File Transfer Protocol

GCM : Galois Counter Mode

GPG : Gnu Privacy Guard

GPO : Group Policy Object

GPS : Global Positioning System

GPU : Graphic Processing Unit

GRE : Generic Routing Encapsulation

HA : High Availability

HDD : Hard Disk Drive

HIDS : Host-based Intrusion Detection System

HIPS : Host-based Intrusion Prevention System

HMAC : Hashed Message Authentication Code

HOTP : HMAC-based One-Time Password

HSM : Hardware Security Module

HTML : Hypertext Markup Language

HTTP : Hypertext Transfer Protocol

HTTPS : Hypertext Transfer Protocol over SSL/TLS

HVAC : Heating, Ventilation and Air Conditioning

IaaS : Infrastructure as a Service

ICMP : Internet Control Message Protocol

ICS : Industrial Control Systems

ID : Identification

IDEA : International Data Encryption Algorithm

IDF : Intermediate Distribution Frame

IdP : Identity Provider

IDS : Intrusion Detection System

IEEE : Institute of Electrical and Electronic Engineers

IIS : Internet Information System

IKE : Internet Key Exchange

IM : Instant Messaging

IMAP4 : Internet Message Access Protocol v4

IoT : Internet of Things

IP : Internet Protocol

IPSec : Internet Protocol Security

IR : Incident Response

IR : Infrared

IRC : Internet Relay Chat

IRP : Incident Response Plan

ISA : Interconnection Security Agreement

ISP : Internet Service Provider

ISSO : Information Systems Security Officer

ITCP : IT Contingency Plan

IV : Initialization Vector

KDC : Key Distribution Center

KEK : Key Encryption Key

L2TP : Layer 2 Tunneling Protocol

LAN : Local Area Network

LDAP : Lightweight Directory Access Protocol

LEAP : Lightweight Extensible Authentication Protocol

MaaS : Monitoring as a Service

MAC : Mandatory Access Control

MAC : Media Access Control

MAC : Message Authentication Code

MAN : Metropolitan Area Network

MBR : Master Boot Record

MD5 : Message Digest 5

MDF : Main Distribution Frame

MDM : Mobile Device Management

MFA : Multi-Factor Authentication

MFD : Multi-function Device

MITM : Man-in-the-Middle

MMS : Multimedia Message Service

MOA : Memorandum of Agreement

MOU : Memorandum of Understanding

MPLS : Multi-protocol Label Switching

MSCHAP : Microsoft Challenge Handshake

Authentication : Protocol

MSP : Managed Service Provider

MTBF : Mean Time Between Failures

MTTF : Mean Time to Failure

MTTR : Mean Time to Recover or Mean Time to Repair

MTU : Maximum Transmission Unit

NAC : Network Access Control

NAT : Network Address Translation

NDA : Non-disclosure Agreement

NFC : Near Field Communication

NGAC : Next Generation Access Control

NIDS : Network-based Intrusion Detection System

NIPS : Network-based Intrusion Prevention System

NIST : National Institute of Standards & Technology

NTFS : New Technology File System

NTLM : New Technology LAN Manager

NTP : Network Time Protocol

OAUTH : Open Authorization

OCSP : Online Certificate Status Protocol

OID : Object Identifier

OS : Operating System

OTA : Over The Air

OVAL : Open Vulnerability Assessment Language

P12 : PKCS #12

P2P : Peer to Peer

PaaS : Platform as a Service

PAC : Proxy Auto Configuration

PAM : Pluggable Authentication Modules

PAP : Password Authentication Protocol

PAT : Port Address Translation

PBKDF2 : Password-based Key Derivation Function 2

PBX : Private Branch Exchange

PCAP : Packet Capture

PEAP : Protected Extensible Authentication Protocol

PED : Personal Electronic Device

PEM : Privacy-enhanced Electronic Mail

PFS : Perfect Forward Secrecy

PFX : Personal Exchange Format

PGP : Pretty Good Privacy

PHI : Personal Health Information

PII : Personally Identifiable Information

PIV : Personal Identity Verification

PKI : Public Key Infrastructure

POODLE : Padding Oracle on Downgrade Legacy Encryption

POP : Post Office Protocol

POTS : Plain Old Telephone Service

PPP : Point-to-Point Protocol

PPTP : Point-to-Point Tunneling Protocol

PSK : Pre-shared Key

PTZ : Pan-Tilt-Zoom

RA : Recovery Agent

RA : Registration Authority

RAD : Rapid Application Development

RADIUS : Remote Authentication Dial-in User Server

RAID : Redundant Array of Inexpensive Disks

RAS : Remote Access Server

RAT : Remote Access Trojan

RBAC : Role-based Access Control

RBAC : Rule-based Access Control

RC4 : Rivest Cipher version 4

RDP : Remote Desktop Protocol

RFID : Radio Frequency Identifier

RIPEMD : RACE Integrity Primitives

Evaluation : Message Digest

ROI : Return on Investment

RMF : Risk Management Framework

RPO : Recovery Point Objective

RSA : Rivest, Shamir, & Adleman

RTBH : Remotely Triggered Black Hole

RTO : Recovery Time Objective

RTOS : Real-time Operating System

RTP : Real-time Transport Protocol

S/MIME : Secure/Multipurpose Internet Mail Extensions

SaaS : Software as a Service

SAML : Security Assertions Markup Language

SAN : Storage Area Network

SAN : Subject Alternative Name

SCADA : System Control and Data Acquisition

SCAP : Security Content Automation Protocol

SCEP : Simple Certificate Enrollment Protocol

SCP : Secure Copy

SCSI : Small Computer System Interface

SDK : Software Development Kit

SDLC : Software Development Life Cycle

SDLM : Software Development Life Cycle Methodology

SDN : Software Defined Network

SED : Self-encrypting Drive

SEH : Structured Exception Handler

SFTP : Secured File Transfer Protocol

SHA : Secure Hashing Algorithm

SHTTP : Secure Hypertext Transfer Protocol

SIEM : Security Information and Event Management

SIM : Subscriber Identity Module

SLA : Service Level Agreement

SLE : Single Loss Expectancy

SMB : Server Message Block

SMS : Short Message Service

SMTP : Simple Mail Transfer Protocol

SMTPS : Simple Mail Transfer Protocol Secure

SNMP : Simple Network Management Protocol

SOAP : Simple Object Access Protocol

SoC : System on Chip

SPF : Sender Policy Framework

SPIM : Spam over Internet Messaging

SPoF : Single Point of Failure

SQL : Structured Query Language

SRTP : Secure Real-Time Protocol

SSD : Solid State Drive

SSH : Secure Shell

SSID : Service Set Identifier

SSL : Secure Sockets Layer

SSO : Single Sign-on

STP : Shielded Twisted Pair

TACACS+ : Terminal Access Controller Access

Control : System Plus

TCP/IP : Transmission Control Protocol/Internet Protocol

TGT : Ticket Granting Ticket

TKIP : Temporal Key Integrity Protocol

TLS : Transport Layer Security

TOTP : Time-based One-time Password

TPM : Trusted Platform Module

TSIG : Transaction Signature

UAT : User Acceptance Testing

UAV : Unmanned Aerial Vehicle

UDP : User Datagram Protocol

UEFI : Unified Extensible Firmware Interface

UPS : Uninterruptable Power Supply

URI : Uniform Resource Identifier

URL : Universal Resource Locator

USB : Universal Serial Bus

USB : OTG USB On The Go

UTM : Unified Threat Management

UTP : Unshielded Twisted Pair

VDE : Virtual Desktop Environment

VDI : Virtual Desktop Infrastructure
VLAN : Virtual Local Area Network
VLSM : Variable Length Subnet Masking
VM : Virtual Machine
VoIP : Voice over IP
VPN : Virtual Private Network
VTC : Video Teleconferencing
WAF : Web Application Firewall
WAP : Wireless Access Point
WEP : Wired Equivalent Privacy
WIDS : Wireless Intrusion Detection System
WIPS : Wireless Intrusion Prevention System
WORM : Write Once Read Many
WPA : WiFi Protected Access
WPA2 : WiFi Protected Access 2
WPS : WiFi Protected Setup
WTLS : Wireless TLS
XML : Extensible Markup Language
XOR : Exclusive Or
XSRF : Cross-site Request Forgery
XSS : Cross-site Scripting