CISM®

DOMAIN 3

Welcome to the third CISM Domain.

- 27% of the exam questions on the certification are from this domain.
- Access control. - MAC, DAC, RBAC, ABAC
- Objects and subjects.
- IAAA - Identification and Authentication, Authorization and Accountability
- Type 1, 2, and 3 authentication - Something you know, something you have, and something you are
- The history of cryptography, symmetric/asymmetric encryption, hashing, and digital signatures
- Patch management, configuration management, and change management
- Security assessment and security audits.
- Social engineering.
- Software testing.
- Buying software from other companies.
- Data remanence and destruction.

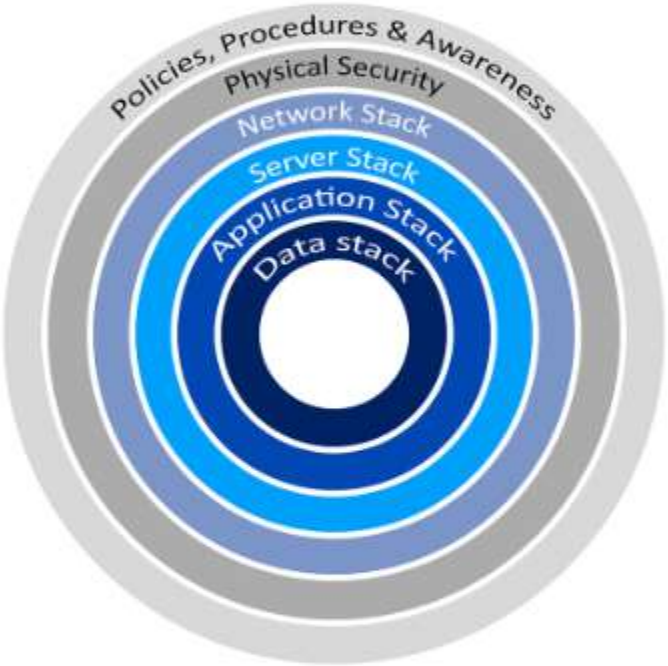
Access Control Defensive Categories and Types:

- **Access Control Categories:**
  - **Administrative (Directive) Controls:**
    - Organizational policies and procedures.
    - Regulation.
    - Training and awareness.
  - **Technical Controls:**
    - Hardware/software/firmware – Firewalls, routers, encryption.
  - **Physical Controls:**
    - Locks, fences, guards, dogs, gates, bollards.

Access Control Defensive Categories and Types:

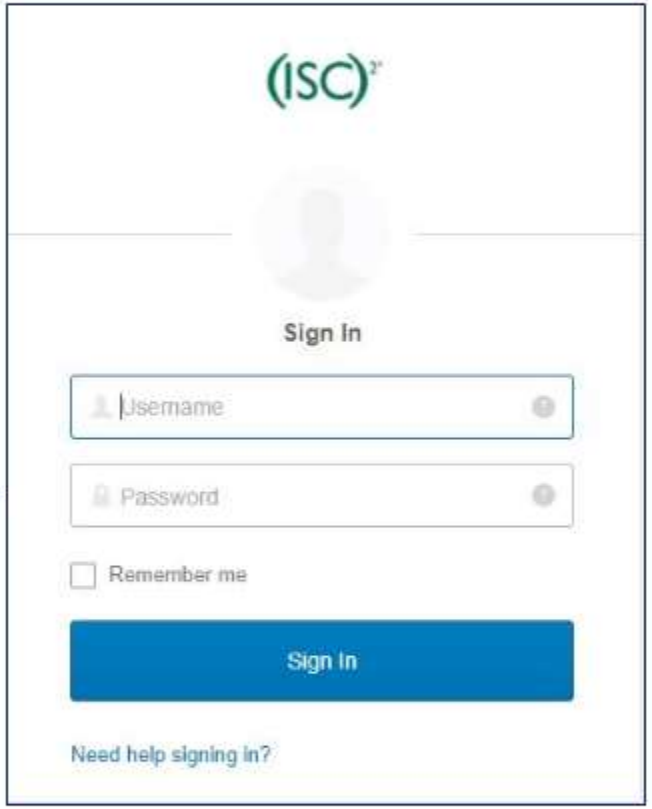
- **Access Control Types** (Many can be multiple types – On the exam look at question content to see which type it is).
  - **Preventative:**
    - Prevents action from happening – Least privilege, drug tests, IPS, firewalls, encryption.
  - **Detective:**
    - Controls that Detect during or after an attack – IDS, CCTV, alarms, anti-virus.
  - **Corrective:**
    - Controls that Correct an attack – Anti-virus, patches, IPS.
  - **Recovery:**
    - Controls that help us Recover after an attack – DR Environment, backups, HA Environments.
  - **Deterrent:**
    - Controls that Deter an attack – Fences, security guards, dogs, lights, Beware of the dog signs.
  - **Compensating:**
    - Controls that Compensate – other controls that are impossible or too costly to implement.

- **Access Control:**
  - Our **Access Control** is determined by our policies, procedures, and standards.
  - This outlines how we grant access whom to what:
    - We use least privilege, need to know, and we give our staff and systems exactly the access they need and no more.
  - Access control spans all the layers of our defense in depth model, different permissions are granted to different subjects depending on their need to access the systems or data and that adheres to the procedures for that area.
  - We covered some of the physical parts of access control in Domain 3's Physical Security, how we use fences, locks, turnstiles, bollards, ...
  - On the logical side we do this by implementing the access security models we talked about in Domain 1, how we Identify, Authenticate, Authorize our subjects and how we keep them Accountable (IAAA).
  - We never use group logins or accounts, they have no accountability.



IAAA (Identification and Authentication, Authorization and Accountability):

- **Identification**
  - Your name, username, ID number, employee number, SSN etc.
  - "I am Thor".
- **Authentication**
  - "Prove you are Thor". – Should **always** be done with multi-factor authentication!
    - **Something you know - Type 1** Authentication (passwords, pass phrase, PIN etc.).
    - **Something you have - Type 2** Authentication (ID, passport, smart card, token, cookie on PC etc.).
    - **Something you are - Type 3** Authentication (and Biometrics) (Fingerprint, iris scan, facial geometry etc.).
    - **Somewhere you are - Type 4** Authentication (IP/MAC Address).
    - **Something you do - Type 5** Authentication (Signature, pattern unlock).



IAAA:

- **Authorization**
  - What are you allowed to access – We use Access Control models, what and how we implement depends on the organization and what our security goals are. More on this in Domain 5 - Identity and Access Management (DAC, MAC, RBAC, RUBAC)
- **Accountability** (also often referred to as Auditing)
  - Trace an Action to a Subject's Identity:
    - Prove who/what a given action was performed by (non-repudiation).

IT Security is there to Support the organization.

- We are there to enable the organization to fulfil its mission statement and the business' goals.
- We are **not** the most important part of the organization, but we span the entire organization.
- We are Security leaders **and** Business leaders – Answer exam questions wearing BOTH hats.

Security governance principles.

- **Least Privilege and Need to Know.**
  - **Least Privilege** – (Minimum necessary access) Give users/systems exactly the access they need, no more, no less.
  - **Need to Know** – Even if you have access, if you do not need to know, then you should not access the data.

- **Non-repudiation.**
  - A user can not deny having performed a certain action. This uses both Authentication and Integrity.

- **Subject and Object.**
  - **Subject** – (Active) Most often users, but can also be programs – Subject manipulates Object.
  - **Object** – (Passive) Any passive data (both physical paper and data) – Object is manipulated by Subject.
  - Some can be both at different times, an active program is a subject; when closed, the data in program can be object.

- **IAAA Access Management:**
  - **Something you know - Type 1 Authentication:**
    - Passwords, pass phrase, PIN etc., also called Knowledge factors.
    - The subject uses these to authenticate their identity, if they know the secret, they must be who they say they are
    - This is the most commonly used form of authentication, and a password is the most common knowledge factor.
    - The user is required to prove knowledge of a secret in order to authenticate.
    - Variations include both longer ones formed from multiple words (a passphrase) and the shorter purely numeric PINs (personal identification number) commonly used for cash machines (ATM's).
    - It is the weakest form of authentication, and can easily be compromised.
    - Secret questions like "Where were you born?" are poor examples of a knowledge factor, it is known by a lot of people and can often be researched easily.
      - Sarah Palin had her email account hacked during the 2008 US Presidential campaign using her secret questions. Since she used basic ones (high school and birthday, …) the hackers could easily find that information online, he reset her password with the information and gained full control of her email account.

- **IAAA Access Management:**
  - **Something you know - Type 1 Authentication:**
    - **Passwords:**
      - It is always easier to guess or steal passwords than it is to break the encryption.
      - We have password policies to ensure they are as secure as possible.
        - They should contain minimum length, upper/lower case letters, numbers and symbols, they should not contain full words or other easy to guess phrases.
        - They have an expiration date, password reuse policy and minimum use before users can change it again.
        - Common and less secure passwords often contain:
          - The name of a pet, child, family member, significant other, anniversary dates, birthdays, birthplace, favorite holiday, something related to a favorite sports team, or the word "password".
          - Winter2017 is not a good password, even if it does fulfil the password requirements.
      - **Key stretching** – Adding 1-2 seconds to password verification.
        - If an attacker is brute forcing a password and needs millions of tries it will become an unfeasible attack.

- **IAAA Access Management:**
  - **Something you know - Type 1 Authentication:**
    - **Passwords:**
      - **Brute Force attacks** (Limit number of wrong logins):
        - Uses the entire key space (every possible key), with enough time any plaintext can be decrypted.
        - Effective against all key based ciphers except the one-time pad, it would eventually decrypt it, but it would also generate so many false positives the data would be useless.
      - **Dictionary attacks** (Limit number of wrong logins, do not allow dictionary words in passwords):
        - Based on a pre-arranged listing, often dictionary words
        - Often succeed because people choose short passwords that are ordinary words and numbers at the end.
      - **Rainbow tables attacks** (Limit number of wrong logins, Salts):
        - Pre-made list of plaintext and matching ciphertext.
        - Often Passwords and matching Hashes a table can have 1,000,000's of pairs.

- **IAAA Access Management:**
  - **Something you know - Type 1 Authentication:**
    - **Passwords:**
      - **Keylogging** (Keystroke logging):
        - A keylogger is added to the user's computer and it records every keystroke the user enters.
        - **Hardware**, attached to the USB port where the keyboard is plugged in.
          - Can either call home or needs to be removed to retrieve the information
        - **Software**, a program installed on the computer.
          - The computer is often compromised by a trojan, where the payload is the keylogger or a backdoor.
          - The keylogger calls home or uploads the keystrokes to a server at regular intervals.

- **IAAA Access Management:**
  - **Something you know - Type 1 Authentication:**
    - **Passwords:**
      - **Salt (salting):**
        - Random data that is used as an additional input to a one-way function that hashes a password or passphrase.
        - Salts are very similar to nonce.
        - The primary function of salts is to defend against dictionary attacks or a pre-compiled rainbow table attack.
      - **Nonce:** (arbitrary number that may only be used once).
        - It is often a random or pseudo-random number issued in an authentication protocol to ensure that old communications cannot be reused in replay attacks.
        - They can also be useful as initialization vectors and in cryptographic hash function.

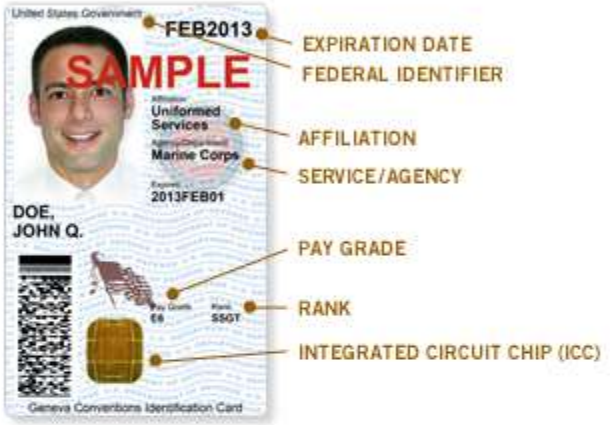
- **IAAA Access Management:**
  - **Something you know - Type 1 Authentication:**
    - **Passwords:**
      - **Clipping levels:** Clipping levels are in place to **prevent administrative overhead.** 
        - It allows authorized users who forget or mistype their password to still have a couple of extra tries.
        - It prevents password guessing by locking the user account for a certain timeframe (an hour), or until unlocked by an administrator.
      - Many systems store a cryptographic hash of passwords.
        - If an attacker can get access to the file of hashed passwords guessing can be done off-line, rapidly testing candidate passwords against the true password's hash value.
        - This will circumvent the clipping levels, stealing is always easier than decrypting it.
      - Some access systems store user passwords in plaintext, they are used to compare user log on attempts.
        - We need to secure every link in the chain, attackers will go for the weakest one, it is often people, but can just as well be our systems.

- **IAAA Access Management:**
  - **Something you know - Type 1 Authentication:**
    - **Password Management:**
      - We covered some password requirements, here are the official recommendations by the U.S. Department of Defense and Microsoft.
        - Password history = set to remember 24 passwords.
        - Maximum password age = 90 days.
        - Minimum password age = 2 days (to prevent users from cycling through 24 passwords to return to their favorite password again).
        - Minimum password length = 8 characters.
        - Passwords must meet complexity requirements = true.
        - Store password using reversible encryption = false.

- **IAAA Access Management:**
  - **Something you have - Type 2 Authentication:**
    - ID, passport, smart card, token, cookie on PC, these are called Possession factors.
      - The subject uses these to authenticate their identity, if they have the item, they must be who they say they are.
      - Simple forms can be credit cards, you have the card and you know the pin, that is multifactor authentication.
      - Most also assume a shared trust, you have your passport, it looks like you on the picture, we trust the issuer so we assume the passport is real.

- **IAAA Access Management:**
  - **Something you have - Type 2 Authentication:**
    - **Single-use passwords:**
      - Having passwords which are only valid once makes many potential attacks ineffective, just like one-time pads.
      - While they are passwords, it is something you have in your possession, not something you know.
      - Some are one-time-pads with a challenge-response or just a pin or phase sent to your phone or email you need to enter to confirm the transaction or the login.
      - Most users find single use passwords extremely inconvenient.
      - They are widely implemented in online banking, where they are known as TANs (Transaction Authentication Numbers).
      - Most private users only do a few transactions each week, the single-use passwords has not led to customers refusing to use it.
        - It is their money, they actually care about keeping those safe.

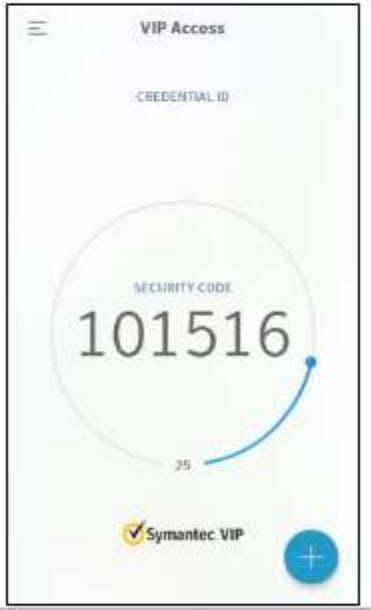
- **IAAA Access Management:**
  - **Something you have - Type 2 Authentication:**
    - **Smart Cards and tokens** (contact or contactless):
      - They contain a computer circuit using an ICC (Integrated Circuit Chip).
      - **Contact Cards** - Inserted into a machine to be read.
        - This can be credit cards you insert into the chip reader or the DOD CAC (Common Access Card).
      - **Contactless Cards** - can be read by proximity.
        - Key fobs or credit cards where you just hold it close to a reader.
        - They use a RFID (Radio Frequency Identification) tag (transponder) which is then read by a RFID Transceiver.
    - **Magnetic Stripe Cards:**
      - Swiped through a reader, no circuit.
      - Very easy to duplicate.



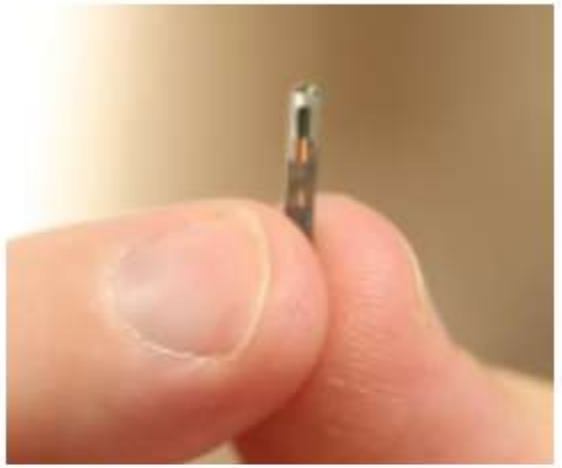
ThorTeaches.com

- **IAAA Access Management:**
  - **Something you have - Type 2 Authentication:**
    - **Tokens:**
      - HOTP and TOTP can be either hardware or software based.
      - Cellphone software applications are more common now.
        - **HOTP** (HMAC-based one-time password):
          - Shared secret and incremental counter, generate code when asked, valid till used.
        - **TOTP** (Time-based One-Time Password):
          - Time based with shared secret, often generated every 30 or 60 seconds, synchronized clocks are critical.

TOTP tokens Hardware ↑
Software ↓



- **IAAA Access Management:**
  - **Something you have - Type 2 Authentication:**
    - A Wisconsin company Three Square Market (32M) is offering to implant tiny radio-frequency chips in its employees.
    - They says the employees are lining up for the technology.
    - Employees who have the rice-grain-sized RFID chip implanted between their thumb and forefinger can then use it "to make purchases in their break room micro market, open doors, login to computers, use the copy machine,"
    - "The chip is not trackable and only contains information you choose to associate with it," the company said "This chip does not have GPS capabilities."
    - I would never do this, I understand they save 5 seconds at the copier.
      - I just have an innate skepticism when companies says, “We can’t or we won’t use this for anything else than intended”.
      - History proves they rarely do just that.

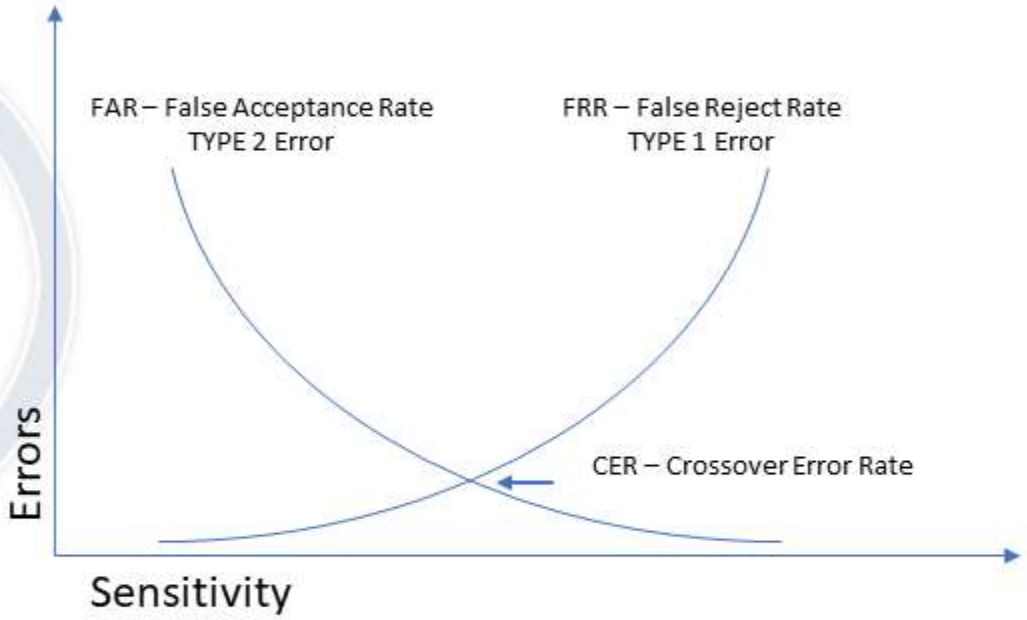


RDFI chip to make access easier.

- **IAAA Access Management:**
  - **Something you are - Type 3 Authentication (Biometrics):**
    - Fingerprint, iris scan, facial geometry etc., these are also called realistic authentication.
      - The subject uses these to authenticate their identity, if they are that, they must be who they say they are.
      - Something that is unique to you, this one comes with more issues than the two other common authentication factors.
      - We can allow unauthorized people into our facilities or systems if we accept someone by mistake. (False Accept)
      - We can prevent our authorized people from entering our facilities if we refuse them by mistake. (False Reject).

Fingerprint reader, with keypad.
This is multifactor authentication.

- **IAAA Access Management:**
  - **Something you are - Type 3 Authentication (Biometrics):**
    - Errors for Biometric Authentication:
      - **FRR** (False rejection rate) Type 1 error:
        - Authorized users are rejected.
        - This can be too high settings - 99% accuracy on biometrics.
      - **FAR** (False accept rate) Type 2 error:
        - Unauthorized user is granted access.
        - This is a very serious error.
      - We want a good mix of FRR and FAR where they meet on the graph is the **CER** (Crossover Error Rate), this is where we want to be.



- **IAAA Access Management:**
  - **Something you are - Type 3 Authentication (Biometrics):**
    - Biometric identifiers are often categorized as physiological and behavioral characteristics.
      - **Physiological characteristics** uses the shape of the body, these do not change unless a drastic event occurs.
        - Fingerprint, palm veins, facial recognition, DNA, palm print, hand geometry, iris recognition, retina and odor.
      - **Behavioral characteristics** uses the pattern of behavior of a person, these can change, but most often revert back to the baseline.
        - Typing rhythm, how you walk, signature and voice.
    - We also need to respect and protect our employees privacy:
      - Some fingerprint patterns are related to chromosomal diseases.
      - Iris patterns could reveal genetic sex, retina scans can show if a person is pregnant or diabetic.
      - Hand vein patterns could reveal vascular diseases.
      - Most behavioral biometrics could reveal neurological diseases, etc.

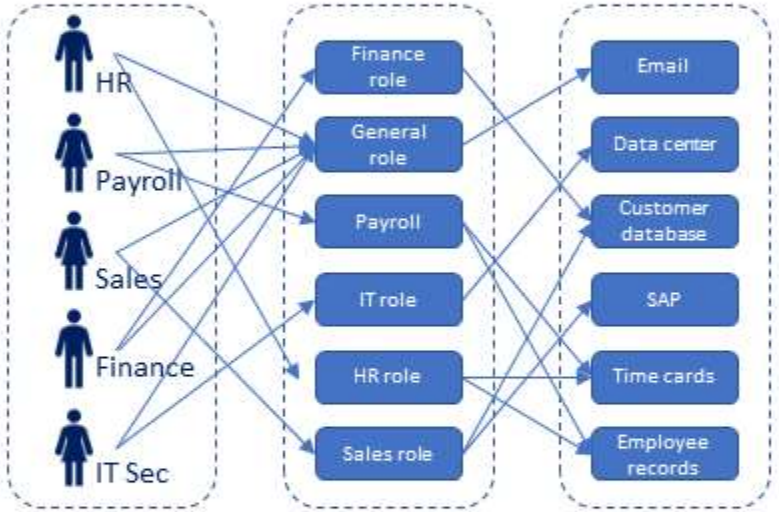
- **IAAA Access Management:**
  - **Something you are - Type 3 Authentication (Biometrics):**
    - Issues with Biometric Authentication:
      - While passwords and smart cards should be safe, because you keep them a secret and secure, biometrics is inherently not and something others can easily find out.
      - Attackers can take pictures of your face, your finger prints, your hands, your ears and print good enough copies to get past a biometric scan.
      - It is possible to copy fingerprints from your high resolution social media posts if you do a peace sign like the one on the right here.
      - How you type, sign your name and your voice pattern can be recorded, also not too difficult to cheat biometrics if it is worth the effort.
      - Some types are still inherently more secure, but they are often also more invasive.

- **IAAA Access Management:**
  - **Something you are - Type 3 Authentication (Biometrics):**
    - Issues with Biometric Authentication:
      - Lost passwords and ID cards can be replaced with new different ones, biometrics can't.
      - Which should make us question even more the mass collection of biometric data.
        - When Home Depot loses 10 million credit card numbers it is bad, but they can be reissued.
        - The US Office of Personnel Management got hacked and lost 5.6 million federal employees' fingerprints.
        - The FBI has a database with 52 million facial images and Homeland Security and U.S. Customs and Border Patrol is working on adding the iris scans and 170 million foreigner fingerprints to the FBI's database.
        - The compromises of the future will have much more wide reaching ramifications than the ones we have seen until now.

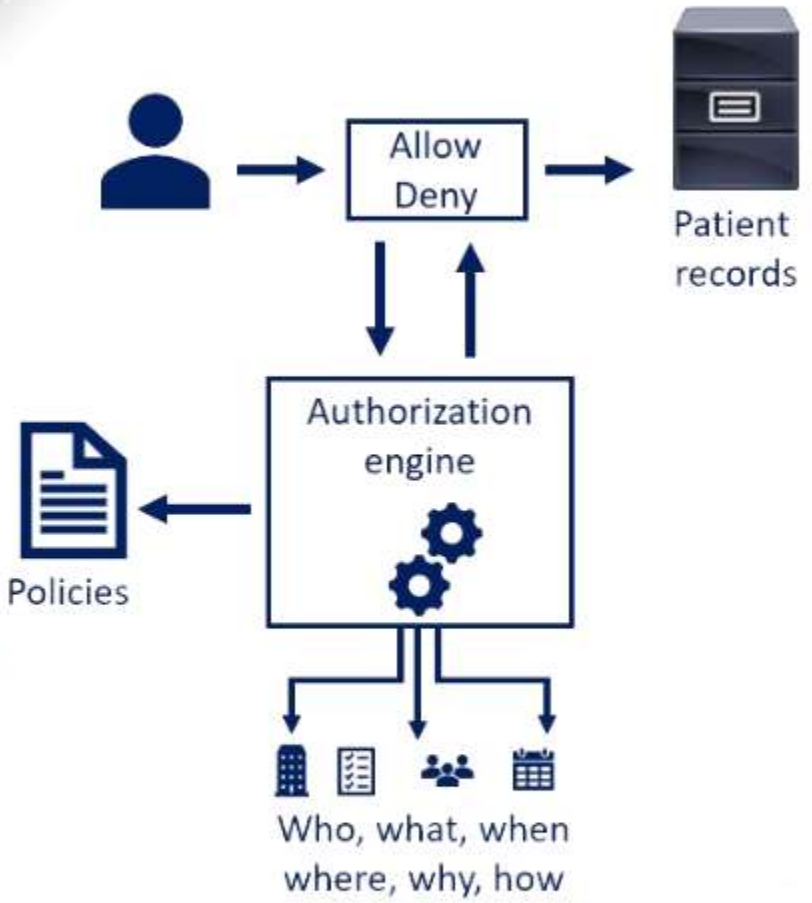
- **IAAA Access Management:**
  - **Authorization**
    - We use Access Control models to determine what a subject is allowed to access
    - What and how we implement depends on the organization and what our security goals are, type can often be chosen dependent on which leg of the CIA Triad is the most important one to us.
    - If it is **Confidentiality** we would most likely go with Mandatory Access Control.
    - If it is **Availability** we would most likely go with Discretionary Access Control.
    - If it is **Integrity** we would most likely go with Role Based Access Control or Attribute Based Access Control.
    - There technically is also RUBAC (Rule Based Access Control), it is mostly used on firewalls with IF/THEN statements, but can be used in conjunction with the other models to provide defense in depth.

- **IAAA Access Management:** 
  - **DAC (Discretionary Access Control):** Often used when Availability is most important.
    - Access to an object is assigned at the discretion of the object owner.
    - The owner can add, remove rights, commonly used by most OS's'.
    - Uses DACL's (Discretionary ACL), based on user identity.
  - **MAC (Mandatory Access Control):** Often used when Confidentiality is most important.
    - Access to an object is determined by labels and clearance, this is often used in the military or in organizations where confidentiality is very important.
    - **Labels**: Objects have Labels assigned to them, the subject's clearance must dominate the object's label.
      - The label is used to allow Subjects with the right clearance access them.
      - Labels are often more granular than just "Top Secret", they can be "Top Secret – Nuclear".
    - **Clearance**: Subjects have Clearance assigned to them.
      - Based on a formal decision on a subject's current and future trustworthiness.
      - The higher the clearance the more in depth the background checks should be.

- **IAAA Access Management:**
  - **RBAC (Role Based Access Control):** Often used when Integrity is most important.
    - Policy neutral access control mechanism defined around roles and privileges.
    - A role is assigned permissions, and subjects in that role are added to the group, if they move to another position they are moved to the permissions group for that position.
    - It makes administration of 1,000's of users and 10,000's of permissions much easier to manage.
    - The most commonly used form of access control.
    - If implemented right it can also enforce separation of duties and prevent authorization/privilege creep .
      - We move employees transferring within the organization from one role to another and we do not just add the new role to the old one.

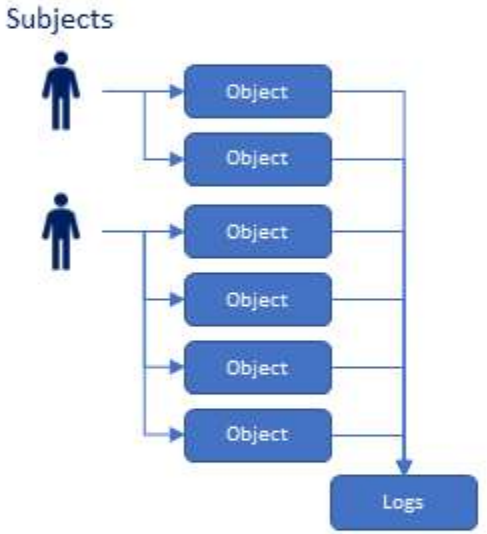


- **IAAA Access Management:**
  - **ABAC** - (Attribute Based Access Control)
  - Access to objects is granted based on subjects, objects AND environmental conditions.
    - Attributes could be:
      - *Subject* (user) – Name, role, ID, clearance, etc.
      - *Object* (resource) – Name, owner, and date of creation.
      - *Environment* – Location and/or time of access, and threat levels.
    - Expected to be used by 70% of large enterprises within the next 5 years, versus around 10% today.
    - Can also be referred to as policy-based access control (PBAC) or claims-based access control (CBAC).

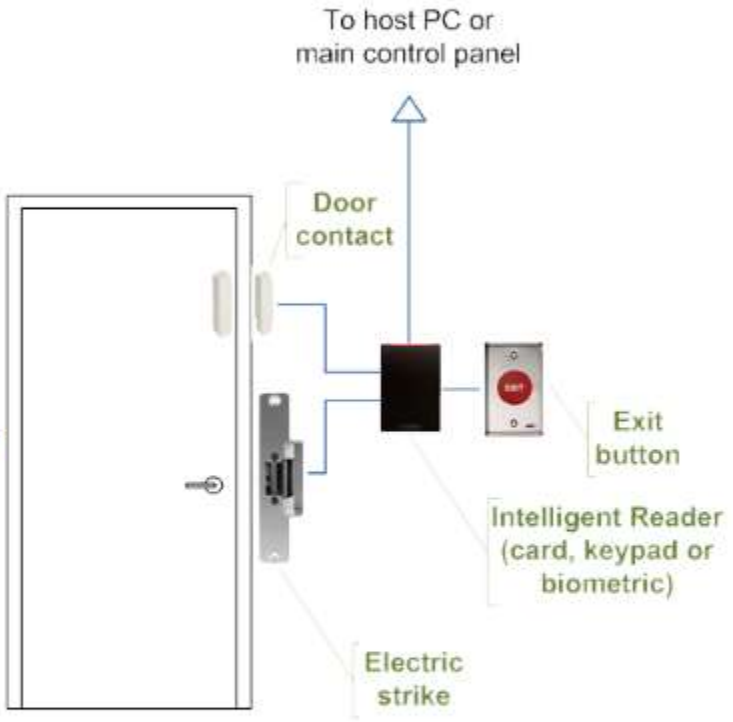


- **IAAA Access Management:**
  - **Context-based access control:**
    - Access to an object is controlled based on certain contextual parameters, such as location, time, sequence of responses, access history.
    - Providing the username and password combination followed by a challenge and response mechanism such as CAPTCHA, filtering the access based on MAC addresses on wireless, or a firewall filtering the data based on packet analysis are all examples of context-dependent access control mechanisms.
  - **Content-based access control:**
    - Access is provided based on the attributes or content of an object, then it is known as a content-dependent access control.
    - In this type of control, the value and attributes of the content that is being accessed determine the control requirements.
    - Hiding or showing menus in an application, views in databases, and access to confidential information are all content-dependent.

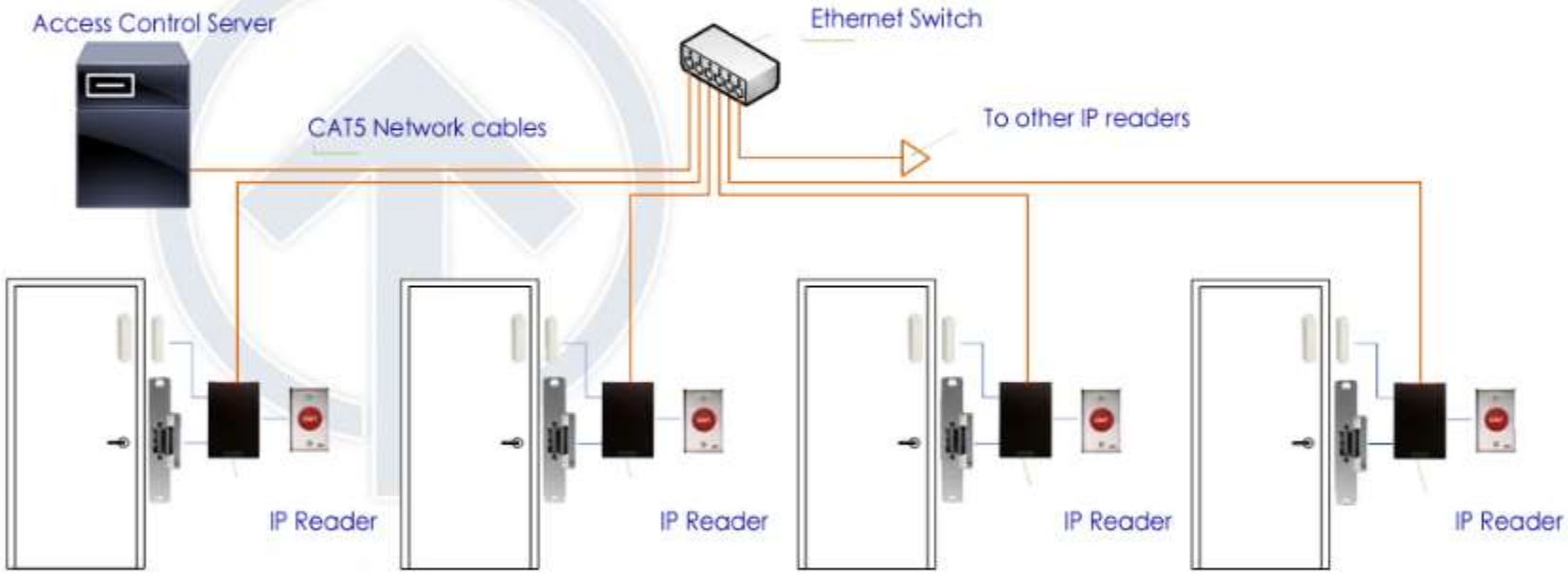
- **IAAA Access Management:**
  - **Accountability** (often referred to as Auditing):
    - Traces an Action to a Subject's Identity:
      - Proves who performed given action, it provides non-repudiation.
      - Group or shared accounts are never OK, they have zero accountability.
      - Uses audit trails and logs, to associate a subject with its actions.



- **Access control:**
  - **Access control systems:**
    - We can use centralized and/or decentralized (distributed) access control systems, depending on which type makes the most sense. Both options provide different benefits.
    - Access control decisions are made by comparing the credential to an access control list.
    - This look-up can be done by a host or server, by an access control panel, or by a reader.
    - Most common is hub and spoke with a control panel as the hub, and the readers as the spokes.
    - Today most private organizations use Role Based Access Control (RBAC).
      - You are in Payroll you get the payroll staff access and permissions, if you move to HR, you lose your payroll access and get HR access assigned.



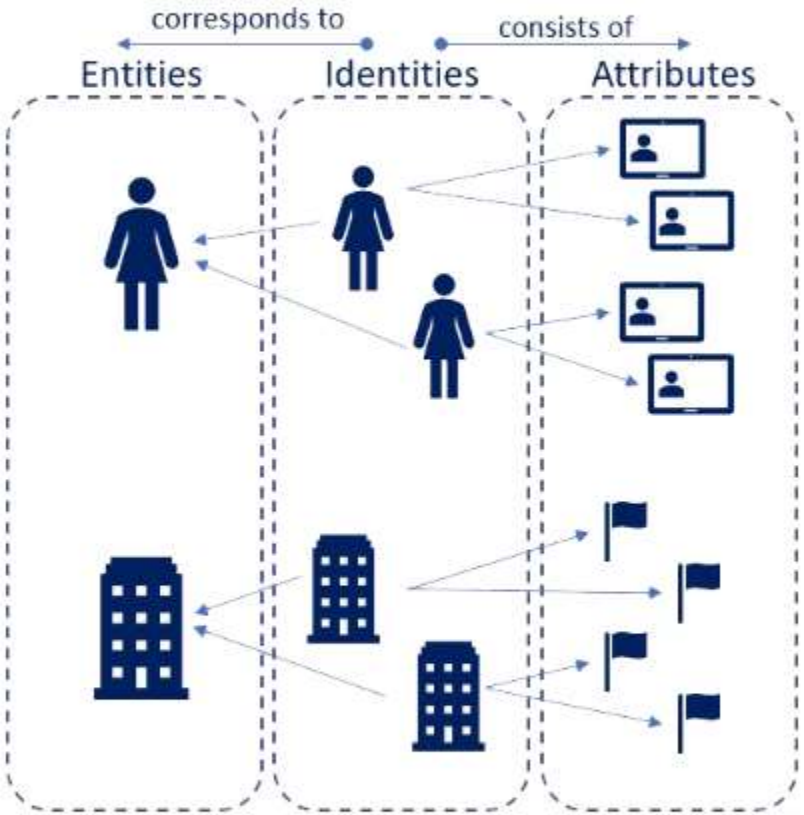
- **Access control:**
  - **Access control systems:**
    - Normal systems are much larger, but you get the idea from this drawing how they would connect.
    - In a perfect world, access control systems should be physically and logically segmented from the rest of our IP Network, in reality it is most often segmented logically with VLANs, but in many cases not even that.



- **Access control:**
  - **Access control systems:**
    - **Centralized Pro's:** (Decentralized Con's):
      - All systems and locations have the same security posture.
      - Easier to manage: All records, configurations and policies are centralized and only configured once per policy.
        - Attackers look for the weakest link in our chain, if a small satellite office is not following our security posture, they can be an easy way onto our network.
      - It is more secure, only a few people have access and can make changes to the system.
      - It can also provide separation of duties, the local admin can't edit/delete logs from their facility.
      - SSO can be used for user access to multiple systems with one login.
    - **Centralized Con's:** (Decentralized Pro's):
      - Traffic overhead and response time, how long does it take for a door lock to authenticate the user against the database at the head office?
      - Is connectivity to the head office stable, is important equipment on redundant power and internet?

- **Access control:**
  - **Access control systems:**
    - **Hybrid:**
      - Controlled centralized, but the access lists for that location are pushed daily/hourly to a local server, local admins have no access.
      - We still need to ensure the local site uses the organization security posture on everything else.

- **Access control:**
  - **Identity and access provisioning:**
    - We can have multiple identities per entity and each identity can have multiple attributes.
      - I can be staff, alumni and enrolled student at a college.
      - As staff I could have access to different areas and data than I would as alumni and student.
      - Companies can have the same, they can be the parent company, then smaller companies under the parent umbrella, all with different attributes.



- **Access control:**
  - **Identity and access provisioning lifecycle:**
    - This is a suggested lifecycle example from "Identity Management Design Guide with IBM Tivoli Identity Manager".
    - You obviously don't have to implement it verbatim, but find a clear policy that works for your organization.
      - Life cycle rules provide administrators with the ability to define life cycle operations to be executed as the result of an event. Life cycle rules are especially useful in automating recurring administrative tasks.
        - Password policy compliance checking.
        - Notifying users to change their passwords before they expire.
        - Identifying life cycle changes such as accounts that are inactive for more than 30 consecutive days.
        - Identifying new accounts that have not been used for more than 10 days following their creation.
        - Identifying accounts that are candidates for deletion because they have been suspended for more than 30 days.
        - When a contract expires, identifying all accounts belonging to a business partner or contractor's employees and revoking their access rights.

- **Access control:**
  - **Federated identity:**
    - How we link a person's electronic identity and attributes across multiple distinct identity management systems.
    - **FIDM** (Federated Identity Management):
      - Having a common set of policies, practices and protocols in place to manage the identity and trust into IT users and devices across organizations.
      - **SSO** is a subset of federated identity management, it only uses authentication and technical interoperability.
    - Technologies used for federated identity include SAML, OAuth, OpenID, Security Tokens, Microsoft Azure Cloud Services, Windows Identity Foundation...
      - **SAML** (Security Assertion Markup Language):
        - An XML-based, open-standard data format for exchanging authentication and authorization data between parties.
        - The single most important requirement that SAML addresses is web browser SSO.

- **Access control:**
  - **Federated identity:**
    - **SSO** (Single sign-on):
      - Users use a single sign-on for multiple systems.
      - Often deployed in organizations where users have to access 10+ systems, and they think it is too burdensome to remember all those passwords.
      - SSO have the same strong password requirements as normal single system passwords.
      - If an attacker compromises a single password they have access to everything that user can access.
    - **Super sign-on.**
      - One login can allow you to access many systems and sites.
      - Social media logins are common super sign-ons, if an account is compromised an attacker can often access multiple other sites or systems, the social media account is linked all the other systems.

- **Access control:**
  - **IDaaS** (Identity as a Service):
    - Identity and access management that is built, hosted and managed by a third-party service provider.
    - Native cloud-based IDaaS solutions can provide SSO functionality through the cloud, Federated Identity Management for Access Governance, Password Management, ...
    - Hybrid IAM solutions from vendors like Microsoft and Amazon provide cloud-based directories that link with on-premises IAM systems.

- **Access control - Authentication protocols:**
  - Communications or cryptographic protocols designed to transfer authentication data between two entities.
  - They authenticate to the connecting entity (often a server) as well as authenticate itself (often a server or desktop) by declaring the type of information needed for authentication as well as syntax.
  - It is the most important layer of protection needed for secure communication between networks.
  - **Kerberos:**
    - Authentication protocol that works on the basis of tickets to allow nodes communicating over a non-secure network to prove their identity to each other in a secure manner.
    - The protocol was named after the character Kerberos (or Cerberus) from Greek mythology, the three-headed guard dog of Hades.
    - It is based on a client–server model and it provides mutual authentication both the user and the server verify each other's identity.
    - Messages are protected against eavesdropping and replay attacks.

Cryptography – the science of secure communication.

- For the exam, what you need to know is that cryptography helps us:
  - Keep our secrets secret (**Confidentiality**) ← This is what most people think all cryptography does.
  - Keep our data unaltered (**Integrity**).
  - Provide a way to verify (**Authentication**) our Subjects; it can also provide **non-repudiation**.
- Cryptography has been used for 1000's of years to keep secrets secret.
- Encryption should be strong enough to be unbreakable or at least take a very long time to break; there obviously needs to be a balance between Confidentiality and Availability.
- **Modular Math:**
  - Cryptography uses a lot of modular math.
  - For the exam you need to know what it is but you don't need to know how to do it.
  - Numbers "wrap around" after they reach a certain value (modulus), which is also why it is called clock math.
  - Adding "X" (24) to "E" (5) = "C" (3) - The English alphabet wraps around after the 26th letter (modulus).

Cryptography – the science of secure communication.

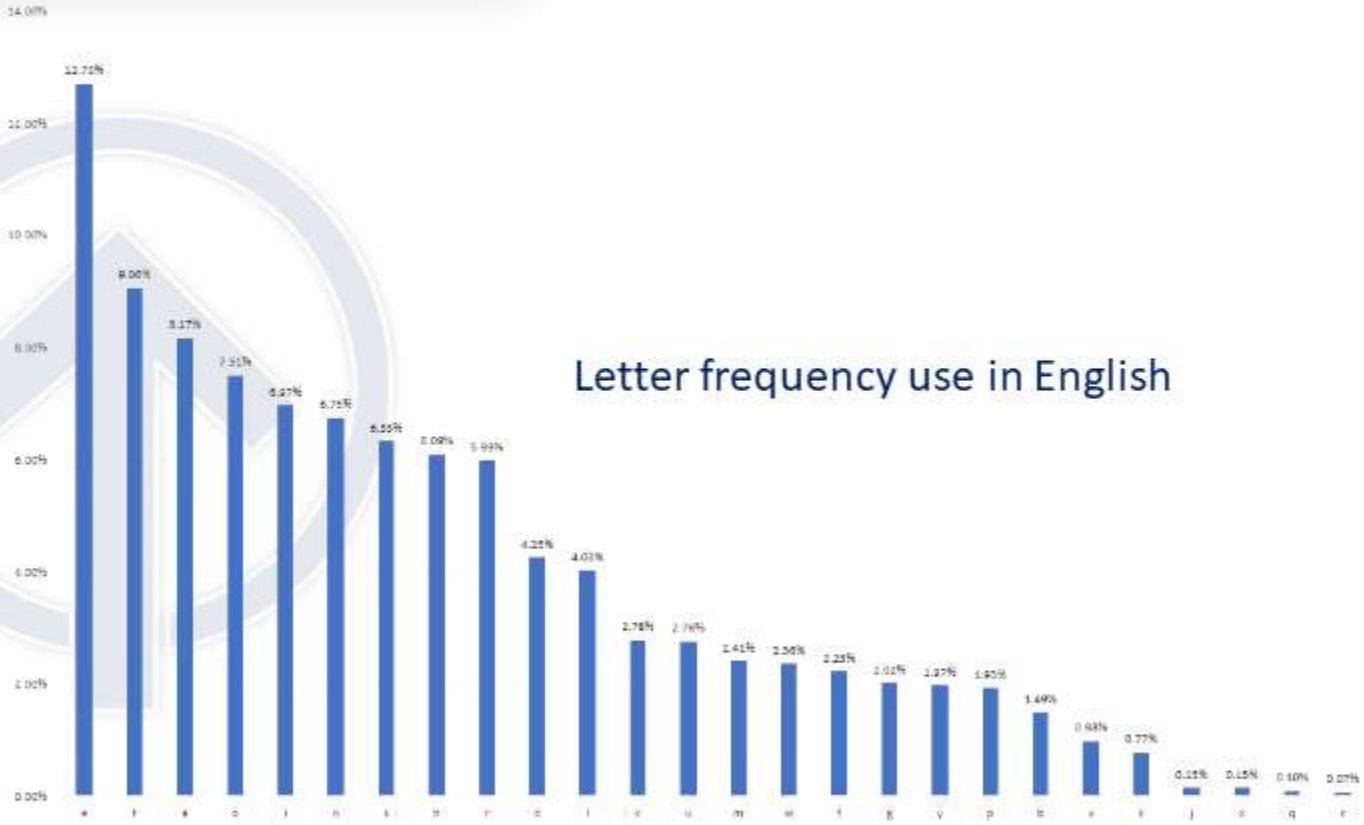
- **Definitions:**
  - **Cryptology** is the science of securing communications.
  - **Cryptography** creates messages where the meaning is hidden.
  - **Cryptanalysis** is the science of breaking encrypted communication.
    - Cryptanalysis is used to breach cryptographic security systems and gain access to the contents of encrypted messages, even if the cryptographic key is unknown.
    - It uses mathematical analysis of the cryptographic algorithm, as well as side-channel attacks that do not target weaknesses in the cryptographic algorithms themselves, but instead exploit weaknesses in their implementation and the devices that run them.
  - **Cipher** is a cryptographic algorithm.

Cryptography – the science of secure communication.

- **Definitions:**
  - **Plaintext** (Cleartext) is an unencrypted message.
  - **Ciphertext** is an encrypted message.
  - **Encryption** converts the plaintext to a ciphertext.
  - **Decryption** turns a ciphertext back into a plaintext.
- **Book Cipher** - Use of a well-known text (Often a book) as the key.
  - Messages would then look like 244.2.13, 12.3.7, 41.42.1. ...
  - The person reviewing the message would look at page 244, sentence 2, word 13, then page 12, sentence 3, word 7, page 41, sentence 42 word 1, ...
- **Running-Key Cipher** – uses a well-known test as a key as well, but uses a previously agreed upon phrase.
  - If we use the CISSP Code of Ethics preamble "The safety and welfare of society and the common good..."
  - The sender would add the plaintext message to the letters from the key, and the receiver would subtract the letters from the key.

Cryptography

- **Mono and Polyalphabetic Ciphers:**
  - **Monoalphabetic Ciphers** - Substitutes one letter for another - "T" would be "W" for instance - very easy to break with frequency analysis (or even without).
  - **Polyalphabetic Ciphers** - Similar but uses different starting point each round, "T" may be "W" on first round, but "D" on second round, more secure, but still not very secure.
- **Frequency Analysis** (analyzing the frequency of a certain character) – In English “E” is used 12.7% of the time. Given enough encrypted substitution text, you can break it just with that.



Letter frequency use in English

Cryptography

- **Exclusive Or (XOR)** ⊕
  - XOR is very useful in basic cryptography; we add a key to the plaintext to make the ciphertext.
  - If we have the Key we can decipher the Cipher text.
  - Used in most symmetric encryption (or at least used in the algorithm behind it).
- **Confusion** is the relationship between the plaintext and ciphertext; it should be as random (confusing) as possible.
- **Diffusion** is how the order of the plaintext should be "diffused" (dispersed) in the ciphertext.
- **Substitution** replaces one character for another, this provides diffusion.
- **Permutation** (transposition) provides confusion by rearranging the characters of the plaintext.

XOR truth table

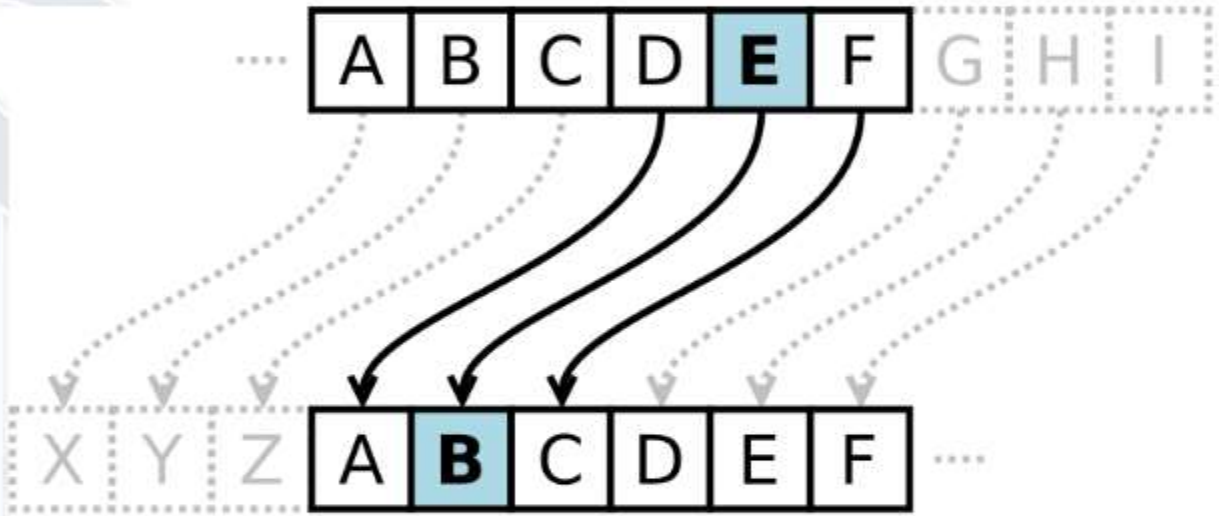
Input		Output
A	B	
0	0	0
0	1	1
1	0	1
1	1	0

0 = false, 1 = true

Plain text 0010101010
Key 1010011001
Ciphertext 1000110011

Cryptography

- **The history of Cryptography** (yes, this is testable).
  - **Spartan Scytale** - Message written lengthwise on a long thin piece of parchment wrapped around a certain size round stick. By itself it would make no sense, but if rewrapped around a stick of the same diameter it would be decipherable.
  - **Caesar Cipher** (Substitution) - Done by switching letters a certain numbers of spots in the alphabet. “Pass the exam" moved 3 back would be “Mxpp qeb buxj.”



Cryptography

- **The history of Cryptography**
  - The **Vigenère cipher** is a polyalphabetic cipher named after Blaise de Vigenère, a French cryptographer who lived in the 16th century.
  - The alphabet is repeated 26 times to form a matrix (Vigenère Square).
  - It uses the plaintext (x axis) and a key (y axis).
    - If the plaintext is CISSP and the key is THOR, the ciphertext would be VPGJI.
    - The key wraps if the plaintext is longer than the key (it normally is).

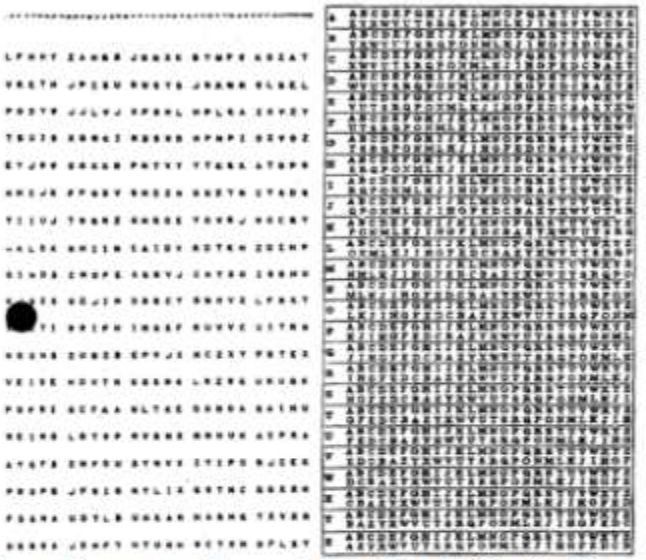
	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
A	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
B	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A
C	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B
D	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C
E	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D
F	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E
G	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F
H	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G
I	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H
J	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I
K	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J
L	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K
M	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L
N	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M
O	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N
P	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
Q	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P
R	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q
S	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R
T	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S
U	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T
V	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U
W	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V
X	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W
Y	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X
Z	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y

Cryptography

- **The history of Cryptography**
  - **Cipher Disk** - 2 concentric disks with alphabets on them, either just as agreed upon "T" is "D" (monoalphabetic) or "T" is "D" again, but the inner disk is turned in an pre-agreed upon direction and turns every X number of letters (decoder rings).
  - **Enigma** - Rotary based. Was 3 rotors early on, which was broken, so the Germans added 1 rotor, making it much harder. Breaking the Enigma was responsible for ending the war early and saving millions of lives.
  - **Purple** (US name) - Japanese rotary based, very similar to the Enigma.
    - Broken by the US, England and Russia (3 rotors).
    - When the Russians learned Japan was not attacking them, they moved the majority of their eastern troops to Moscow to fight the Germans. They had decoded that Japan was going for Southeast Asia.

Cryptography

- **The history of Cryptography:**
  - **One-Time Pad:**
    - Cryptographic algorithm where plaintext is combined with a random key.
    - It is the only existing mathematically unbreakable encryption.
      - While it is unbreakable it is also very impractical.
      - It has ONE use per pad; they should never be reused.
      - Characters on the pad have to be truly random.
      - The pads are kept secure.
  - **Vernam Cipher** (The first known use of a one-time pad).
    - It used bits, and the bits were XORed to the plaintext bits.
  - Project **VENONA** was a project by the US and the UK to break the KGB's encryption from 1943 to 1980.
    - The KGB used one-time pads (unbreakable if not reused) for sensitive transmissions.
    - The KGB reused pads, many messages were decoded, leading to the arrest of many high-profile US residents.



One-time pad used by the U.S. National Security Agency, code named DIANA

Cryptography

- **The history of Cryptography**
  - **The Jefferson Disk** (Bazeries Cylinder) - is a cipher system using a set of wheels or disks, each with the 26 letters of the alphabet arranged around the edge. Jefferson (US president) invented it, and Bazeries improved it.
    - The order of the letters is different for each disk and is usually scrambled in some random way.
    - Each disk is marked with a unique number.
    - A hole in the center of the disks allows them to be stacked on an axle.
    - The disks are removable and can be mounted on the axle in any order desired.
    - The order of the disks is the cipher key, and both sender and receiver must arrange the disks in the same predefined order.
    - Jefferson's device had 36 disks.

Cryptography

- **The history of Cryptography**
  - **SIGABA:**
    - A rotor machine used by the United States throughout World War II and into the 1950s, similar to the Enigma.
    - It was more complex, and was built after examining the weaknesses of the Enigma.
    - No successful cryptanalysis of the machine during its service lifetime is publicly known.
    - It used 3x 5 sets of rotors.
    - The SIGABA was very large, heavy, expensive, difficult to operate, mechanically complex, and fragile.

Cryptography

With the common use of Cryptography, many governments realized how important it was that cryptographic algorithms were added to export restrictions in the same category as munitions.

- **COCOM** (Coordinating Committee of Multilateral Export Controls) 1947 – 1994.
  - Was used to prevent the export of "Critical Technologies" from “Western” countries to the "Iron Curtain" countries during the cold war.
  - Encryption is considered "Critical Technologies"
- **Wassenaar Arrangement** - 1996 – present.
  - Similar to COCOM, but with former "Iron Curtain" countries being members
  - Limits exports on military and "dual-use” technologies. Cryptography is part of that.
  - Some nations also use it to prevent their citizens from having strong encryption (easier to spy on your own people if they can't use strong cryptography).

Cryptography

- **Asymmetric vs Symmetric Encryption and Hybrid:**
  - **Asymmetric**
    - Pros: It does not need a pre-shared key, only 2x users = total keys.
    - Cons: It is much slower, it is weaker per bit.
  - **Symmetric**:
    - Pros: Much faster, stronger per bit.
    - Cons: Needs a pre-shared key, **n(n-1)/2** users, becomes unmanageable with many users.

Users	Symmetric keys	Asymmetric Keys
2	1	4
5	10	10
10	45	20
30	435	60
100	4,950	200
500	124,750	1000
5000	12,497,500	10000
10000	49,995,000	20000

  - **Hybrid Encryption:**
    - Uses Asymmetric encryption to share a Symmetric Key (session key).
    - We use the security over an unsecure media from Asymmetric for the initial exchange and we use the speed and higher security of the Symmetric for the actual data transfer.
    - The Asymmetric Encryption may send a new session key every so often to ensure security.

Cryptography

Symmetric Encryption:

- **DES** - Data Encryption Standard (Single DES).
  - For the exam it may be called DEA (algorithm) or DES (standard)
  - No longer secure and it has multiple attack vectors published.
  - Symmetric - 64 bit block cipher - 56 bit key, 16 rounds of encryption, uses Fistel.
  - DES has 5 different modes it can encrypt data with, they include: Block, Stream, Initialization Vector and if encryption errors propagate to the next block.
    - **ECB** (Electronic Code Book), **CBC** (Cipher Block Chaining), **CFB** (Cipher Feedback), **OFB** (Output Feedback) and **CTR** (Counter).

Cryptography

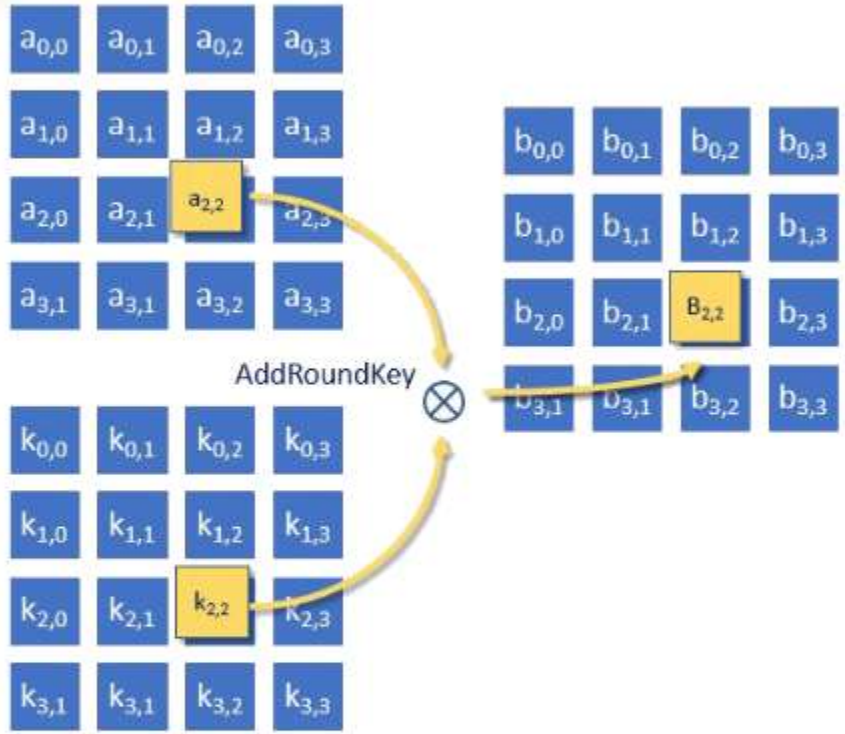
Symmetric Encryption:

- **3 DES** (Triple DES):
  - Was developed to extend life of DES systems while getting ready for AES.
  - Symmetric - 64 bit block cipher - 56 bit key, 16 rounds of encryption, uses Fistel.
  - 3 rounds of DES vs 1.
    - K1 (keymode1) - 3 different keys with 112 bit key strength.
    - K2 (keymode2) - 2 different keys with 80 bits and 1/3 same key.
    - K3 (keymode3) – Same key 3 times, just as insecure as DES (encrypt/decrypt/encrypt).
  - Considered secure until 2030 and still commonly used (K1).
- **IDEA** (International Data Encryption Algorithm):
  - Designed to replace DES.
  - Symmetric, 128bit key, 64bit block size, considered safe.
  - Not widely used now, since it is patented and slower than AES.

Cryptography

Symmetric Encryption:

- **AES** - Advanced Encryption Standard (Rijndael).
  - Symmetric.
  - Considered secure.
  - Open source.
  - Uses both transposition and substitution.
  - Widely used today.
  - AES operates on a 4 × 4 column-major order matrix of bytes.
  - **Initial Round**:
    - **AddRoundKey** — each byte is combined with a block of the round key using bitwise XOR.

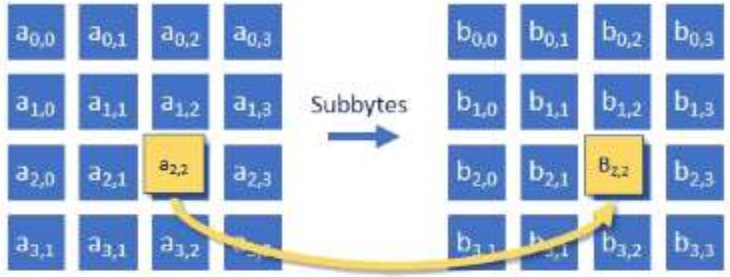


In the AddRoundKey step, each byte is combined with a byte of the subkey using XOR (⊕).

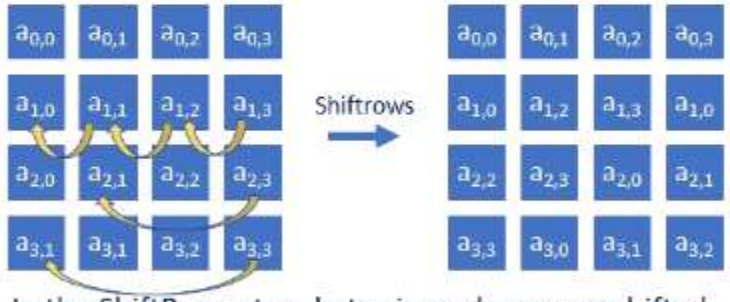
Cryptography

Symmetric Encryption:

- AES
  - Rounds:
    - **SubBytes** — a non-linear substitution step where each byte is replaced with another according to a lookup table.
    - **ShiftRows** — a transposition step where the last three rows of the state are shifted a certain number of steps.
    - **MixColumns** — a mixing operation which operates on the columns, combining the four bytes in each column.
  - **Final Round** (no MixColumns):
    - SubBytes
    - ShiftRows
    - AddRoundKey



In the SubBytes step, each is replaced with its entry in a fixed 8-bit lookup table

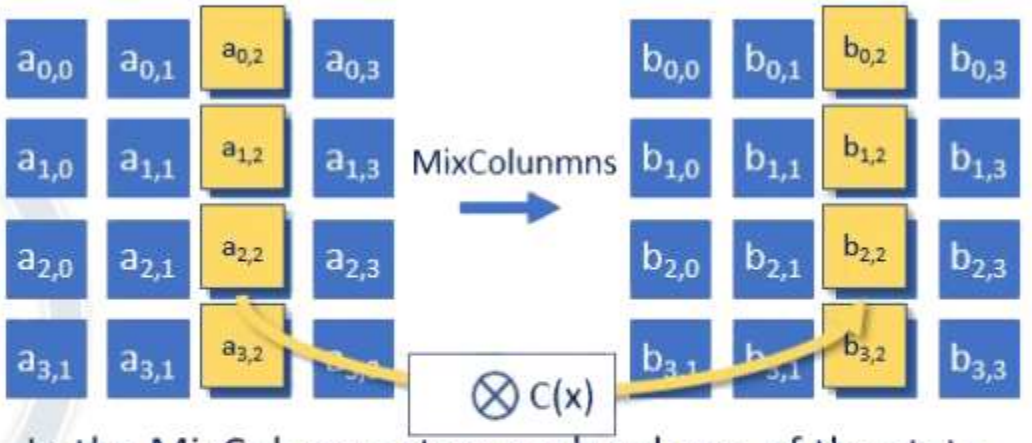


In the ShiftRows step, bytes in each row are shifted to the left. The number of places each byte is shifted differs for each row.

Cryptography

Symmetric Encryption:

- **AES**
  - The key size used for an AES cipher specifies the number of repetitions of transformation rounds that convert the plaintext, into the ciphertext.
  - The number of cycles depends on the key length:
    - 10 cycles for 128-bit keys.
    - 12 cycles for 192-bit keys.
    - 14 cycles for 256-bit keys.



In the MixColumns step, each column of the state is multiplied with a fixed polynomial *c(x)*

Cryptography

Symmetric Encryption:

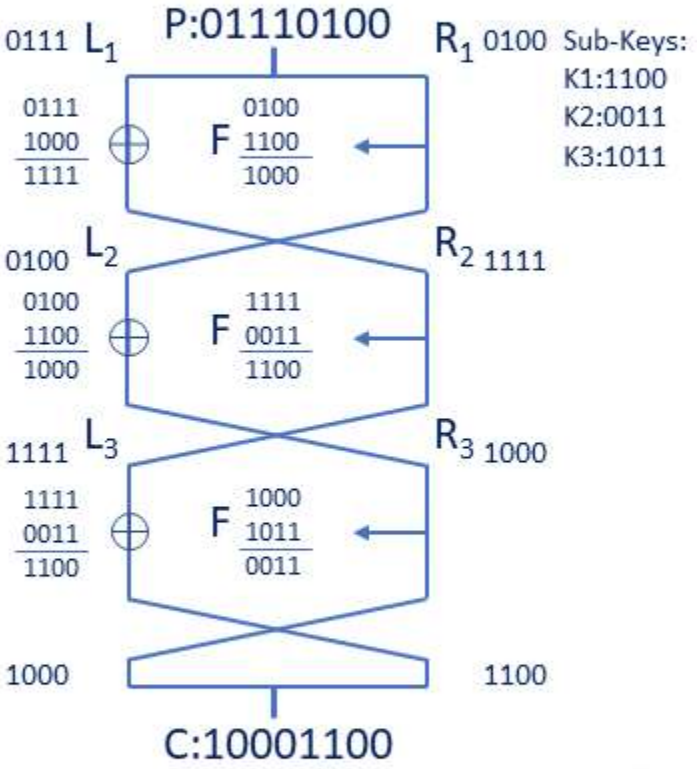
- **Blowfish** - publish domain.
  - Uses Fistel.
  - Symmetric, block cipher, 64 bit blocks, 32-448 bit key lengths.
  - No longer considered secure.
  - Developer recommends using Twofish.

- **Twofish.**
  - Uses Fistel.
  - Symmetric, block cipher 128bit blocks, key length 128, 192, 256 bits.
  - Considered secure.

Cryptography

Symmetric Encryption:

- Fistel cipher (Fistel network):
  - The Cipher splits a plaintext block into two halves (L and R).
  - The process goes through several rounds, the right half of the block does not change.
  - The right half (R_n) is XOR'ed with a subkey (K_n)for each round (F).
  - The XOR'ed value (F) is XOR'ed with the left block (L_n).
  - The recipient reverses the subkey order and XOR's to get the plaintext.
  - **Feistel or modified Feistel Algorithms:**
    - Blowfish, Camellia, CAST-128, DES, FEAL, ICE, KASUMI, LOKI97, Lucifer, MARS, MAGENTA, MISTY1, RC5, TEA, Triple DES, Twofish, XTEA, …
  - **Generalized Feistel Algorithms:**
    - CAST-256, MacGuffin, RC2, RC6, Skipjack.



Cryptography

Symmetric Encryption

- **RC4:**
  - Used by WEP/WPA/SSL/TLS.
  - Pseudorandom keystream.
  - No longer considered secure.
  - Symmetric, Stream cipher, 40-2048 bit key length.
- **RC5:**
  - Symmetric, Block Cipher, 32, 64, 128bit blocks, Key length 0-2040bits, uses Fistel.
  - Considered Secure (if high enough blocks/key).
- **RC6** – AES3 Finalist:
  - Based on RC5, but changed to meet AES requirements, uses Fistel.
  - Symmetric, Block Cipher, 128bit blocks, 128, 192, 256bit key length.
  - Considered Secure.

Cryptography

Asymmetric Encryption (Public Key Encryption)

- We have used symmetric encryption for 1000's of years. Asymmetric is, however, a new player.
  - In the 1970s, multiple Asymmetric keys were developed, including Diffie-Hellman (DH - 1976) and RSA (Rivest, Shamir and Adleman - 1977).
  - **Asymmetric Encryption uses 2 keys: a Public Key and a Private Key (Key Pair).**
    - Your **Public Key** is publicly available.
      - Used by others to encrypt messages sent to you. Since the key is asymmetric, the cipher text can't be decrypted with your public Key.
    - Your **Private Key** - You keep this safe.
      - You use it to decrypt messages sent with your public key.
    - Also used for digital signatures, slightly reversed.
    - You encrypt with your private key and the recipient decrypts with your public key.

Cryptography

Asymmetric Encryption

- **Prime Number Factorization:**
  - Factoring large Prime numbers using a one-way factorization - It is easy to multiply 2 numbers, but hard to discern the 2 numbers multiplied from the result.
  - 1373 x 8081 = 11095213 - It will be hard to tell which numbers were multiplied to get 11095213.
  - Between 1 and 10,000 there are 1229 prime numbers, and strong encryption uses much higher prime numbers.

- **Discrete Logarithms:**
  - Another one-way function - this one uses Logarithms, which is the opposite of exponentiation.
  - 5 to the 12th power = 244140625, but asking 244140625 is 5 to the what power is much harder.
  - Discrete Logarithms apply the concept to groups, making them much harder so solve.

Cryptography

Asymmetric Encryption

- **RSA cryptography**
  - New keypair from **very** large prime numbers - creates public/private key pair.
  - Used to exchange symmetric keys, it is slow, and the algorithm was patent protected (1977-1997 - 20 years).
  - Asymmetric, 1094-4096bit key, Considered secure.
  - **RSA-704** uses these 2 prime numbers, remember I said LARGE prime numbers were factorized:
    - 8143859259110045265727809126284429335877899002167627883200914172429324360133004116702003240828777970252499
    - 9091213529597818878440658302600437485892608310328358720428512168960411528640933367824950788367956756806141
  - They then produce this result, and while this number is known, figuring out the 2 prime numbers is very difficult:
    - 7403756347956171282804679609742957314259318888923128908493623263897276503402826627689199641962511784399589433050212758537011896809828673317327310893090055250511687706329907239638078671008609696253793465056379 6359

Cryptography

Asymmetric Encryption:

- **Diffie–Hellman (DH)** key exchange is a method of securely exchanging cryptographic keys over a public channel and was one of the first public-key protocols.
  - It is one of the earliest practical examples of public key exchange implemented within the field of cryptography.
  - The Diffie–Hellman key exchange method allows two parties that have no prior knowledge of each other to jointly establish a shared secret key over an insecure channel.
    - This key can then be used to encrypt subsequent communications using a symmetric key cipher.
- **Elliptic Curve Cryptography (ECC)** is a one-way function that uses discrete Logarithms applied to elliptical curves. Much stronger per bit than normal discrete Logarithms.
  - Often found on low-power devices since they can use shorter key lengths and be as secure.
  - Patented, so less used since it is patented and costs money to use, 256-bit ECC key is just as strong as a 3,072-bit RSA key.

Cryptography

Asymmetric Encryption:

- **ElGamal** is an asymmetric key encryption algorithm for public-key cryptography which is based on the Diffie–Hellman key exchange. ElGamal encryption is used in the free GNU Privacy Guard software, recent versions of PGP, and other cryptosystems.
- **DSA** (Digital Signature Algorithm) uses a different algorithm for signing and encryption than RSA, yet provides the same level of security. Key generation has two phases.
  - The first phase is a choice of *algorithm parameters* which may be shared between different users of the system, while the second phase computes public and private keys for a single user.
  - DSA is a variant of the ElGamal signature scheme, which should not be confused with ElGamal encryption.
- **Knapsack** (Merkle–Hellman knapsack cryptosystem) is one-way.
  - The public key is used only for encryption, and the private key is used only for decryption, making it unusable for authentication by cryptographic signing.
  - No longer secure.

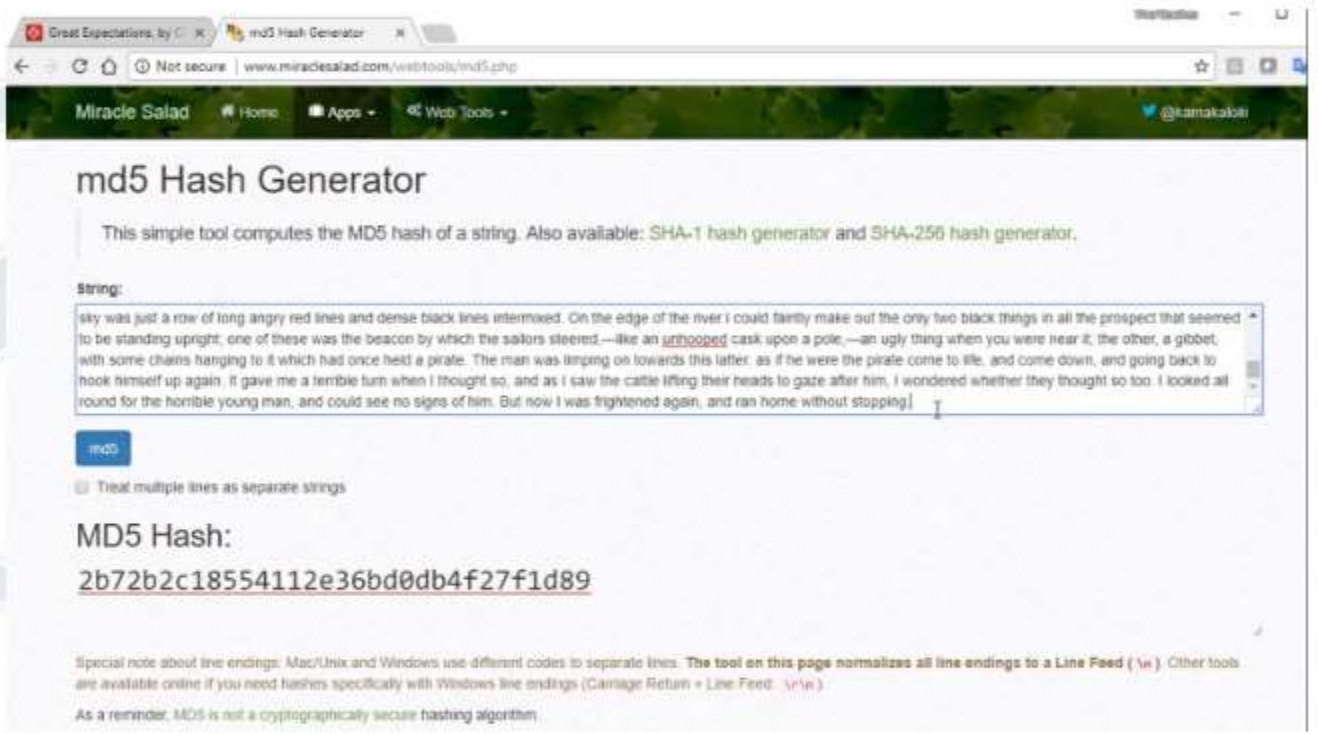
Cryptography

- **Hash Functions** (One-Way Hash Functions) are used for **Integrity**:
  - A **variable-length plaintext** is hashed into a **fixed-length value** hash or MD (Message Digest).
  - It is used to prove the Integrity of the data has not changed. Even changing a comma in a 1000 page document will produce an entirely new hash.
  - **Collisions:** When 2 hashes of different data provide the same hash. It is possible, but very unlikely.
  - **MD5** (Message Digest 5):
    - 128bit Fixed-Length hash, used very widely until a flaw was found making it possible to produce collisions in a reasonable amount of time.
    - While not a chosen-text collision, it is still a collision.
    - Still widely used.
  - **MD6** (Message Digest 6):
    - Was not used for very long; was supposed to replace MD5, but SHA2/3 were better.
    - It was in the running for the SHA3 race, but withdrawn due to flaws.

Cryptography

- **Hash Functions:**
  - Just 1 bit change completely changes the hash.
  - Using *Great Expectations* (Charles Dickens - 1867 Edition again, 4 pages at font size 11, 1827 words, 7731 characters).
  - **Hash#1** is the original **2b72b2c18554112e36bd0db4f27f1d89**
  - **Hash#2** is with 1 comma removed **21b78d32ed57a684e7702b4a30363161**
  - Just a single "**.**" added will change the hash value to **5058f1af8388633f609cadb75a75dc9d**

Remember: variable-length input, fixed-length output.



Cryptography

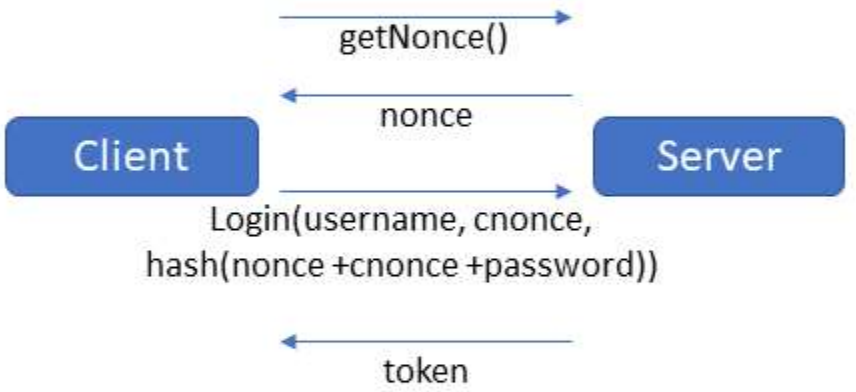
- **Hash Functions:**
  - **SHA1** (Secure Hash Algorithm 1):
    - 160bit Hash Value.
    - Found to have weak collision avoidance, but still commonly used.
  - **SHA2** (Secure Hash Algorithm 2):
    - Considered collision resistant.
    - Somewhat used now, relatively new.
  - **SHA3** (Secure Hash Algorithm 3):
    - Finalized in August 2015.
  - **HAVAL** (Hash of Variable Length):
    - The Message Digest (MD) length is variable (128, 169, 192, 224, 256bits).
    - Uses the MD design principles, but is faster.
    - Not widely used.

Cryptography

- **Hash Functions:**
  - **RIPEMD:**
    - Developed outside of defense to ensure no government backdoors.
    - 128, 256, 320bit hashes.
    - Not widely used.
    - No longer secure.
  - **RIPEMD160:**
    - Redesigned, fixing flaws of RIPEMD.
    - 160bit hashes.
    - Not widely used.
    - Considered secure.

Cryptography

- **Hash Functions:**
  - **Salt (Salting):**
    - Random data that is used as an additional input to a one-way function that "hashes" a password or passphrase.
    - Salts are very similar to nonces.
    - The primary function of salts is to defend against dictionary attacks or a pre-compiled rainbow table attack.
  - **Nonce:** (arbitrary number that may only be used once).
    - It is often a random or pseudo-random number issued in an authentication protocol to ensure that old communications cannot be reused in replay attacks.
    - They can also be useful as initialization vectors and in cryptographic hash function.



Cryptography

- **Cryptographic Attacks:**
  - **Steal the Key** - Modern encryption being so difficult to break, it is easier to recover the private key.
    - Law enforcement does this when they get search warrants, to recover the private key from the PC or phone of someone charged with a crime.
    - Attackers do this by gaining access to your system or key repository; they can then decrypt your data.
  - **Brute Force:**
    - Uses the entire key space (every possible key); with enough time, any plaintext can be decrypted.
    - Effective against all key-based ciphers except the one-time pad; it would eventually decrypt it, but it would also generate so many false positives that the data would be useless.
  - **Key stretching:** Adding 1-2 seconds to password verification.
    - If an attacker is brute forcing password and needs millions of attempts, it will become an unfeasible attack vector.
  - **Digraph attack:** Similar to frequency analysis/attacks, but looks at common pairs of letters (TH, HE, IN, ER).

Cryptography

- **Cryptographic Attacks:**
  - **Man-in-the-Middle Attack (MITM):**
    - The attacker secretly relays and may alter communication between two parties, who believe they are directly communicating with each other.
    - The attacker must be able to intercept all relevant messages passing between the two victims.
    - They can alter the information, just steal it or inject new messages.

→ Hi John, this is Bob. Send me your key.
→ Mike forwards the request.
← John sends his key to Mike
← Mike replaces John's key with his own, and sends to Bob.
→ Bob uses the key to encrypt the message and sends it.
→ Mike decrypts, alters or just steals the Information, encrypts it with John's key and sends it on to John.

  - **Session Hijacking** (TCP Session Hijacking):
    - An attacker takes over a web user's session ID and masquerades as the authorized user.
    - Once the session ID has been accessed, through session prediction the attacker pretends to be the user, and as that user, can do anything the user is authorized to do on the network.

Cryptography

- **Cryptographic Attacks:**
  - Social Engineering
    - Much easier than breaking the key is convincing the key holder to hand it over to the "help desk".

FREE ICECREAM!

A very successful social engineering attack was a Pen-Test company driving up in front of a company office with "Free Ice Cream" and company logo signs on an ice cream van. The employees had to enter their username and password to 'prove' they were real employees. They were rewarded with an "approved" message and got their free ice cream. The Pen-Testers got 90%+ of the employees' usernames and passwords from those who were there that day.

Cryptography

- **Cryptographic Attacks:**
  - **Rainbow Tables:**
    - Pre-made list of plaintext and matching ciphertext.
    - Often Passwords and matching Hashes, a table can contain have 1,000,000's of pairs.
  - **Known Plaintext:**
    - You know the plaintext and the ciphertext, and using those you try to figure out the key.
  - **Chosen Plaintext:**
    - Similar to Known Plaintext, but the attacker choses the plaintext, then tries to figure out the key.
  - **Adaptive Chosen Plaintext:**
    - Same as Chosen Plaintext, the attacker “adapts" the following rounds dependent on the previous rounds.
  - **Meet-in-the-Middle:**
    - A known plaintext attack, the intruder has to know some parts of plaintext and their ciphertexts, used to break ciphers, which have two or more secret keys for multiple encryption using the same algorithm.

Cryptography

- **Cryptographic Attacks**
  - **Known Key** - (Not really known, because if it was, the attacker would have the key).
    - The attacker knows 'something' about the key, making easier to break it.
    - The password could be exactly 8 characters, first character has to be upper case and last has to be a number.
  - **Differential Cryptanalysis:**
    - Tries to find the “difference" between the related plaintexts; if the plaintexts are only a few bits different, can we discern anything? Can we see non-randomness?
    - The same bit should have a 50/50 chance of flipping; areas where this is not so can be a clue to the key.
  - **Linear Cryptanalysis:**
    - A type of known plaintext attack where the attacker has a lot of plaintext/ciphertext pairs created with the same key.
    - The attacker studies the pairs to learn information about the key used to create the ciphertext.
  - **Differential Linear Cryptanalysis** is Differential and Linear Cryptanalysis combined.

Cryptography

- **Cryptographic Attacks:**
  - **Side Channel Attacks:**
    - Attackers use physical data to break a crypto system. This can be CPU cycles, power consumption while encrypting/decrypting, ...
  - **Implementation Attacks:**
    - Some vulnerability is left from the implementation of the application, system or service.
    - It is almost always easier to find a flaw in the system than to break the cryptography.
    - Is the key stored somewhere in plaintext? Is the key stored somewhere not very secure? Is anything stored in memory?
  - **Key Clustering:**
    - When 2 different Symmetric Keys used on the same plaintext produce the same ciphertext, both can decrypt ciphertext from the other key.

Cryptography

- **Implementing Cryptography:**
  - **PKI** (Public Key Infrastructure):
    - Uses Asymmetric and Symmetric Encryption as well as Hashing to provide and manage digital certificates.
    - To ensure PKI works well, we keep the private key secret.
    - We also store a copy of the key pair somewhere central and secure (key repository).
    - We have policies in place that require 2 Security Administrators to retrieve the key pair (if only 1 person did it, chances of key compromise would be higher).
    - If users lose their private key and if no key repository is kept, anything encrypted with the public key is inaccessible).

  - **Key Escrow:**
    - Keys are kept by a 3rd party organization (often law enforcement).

Cryptography

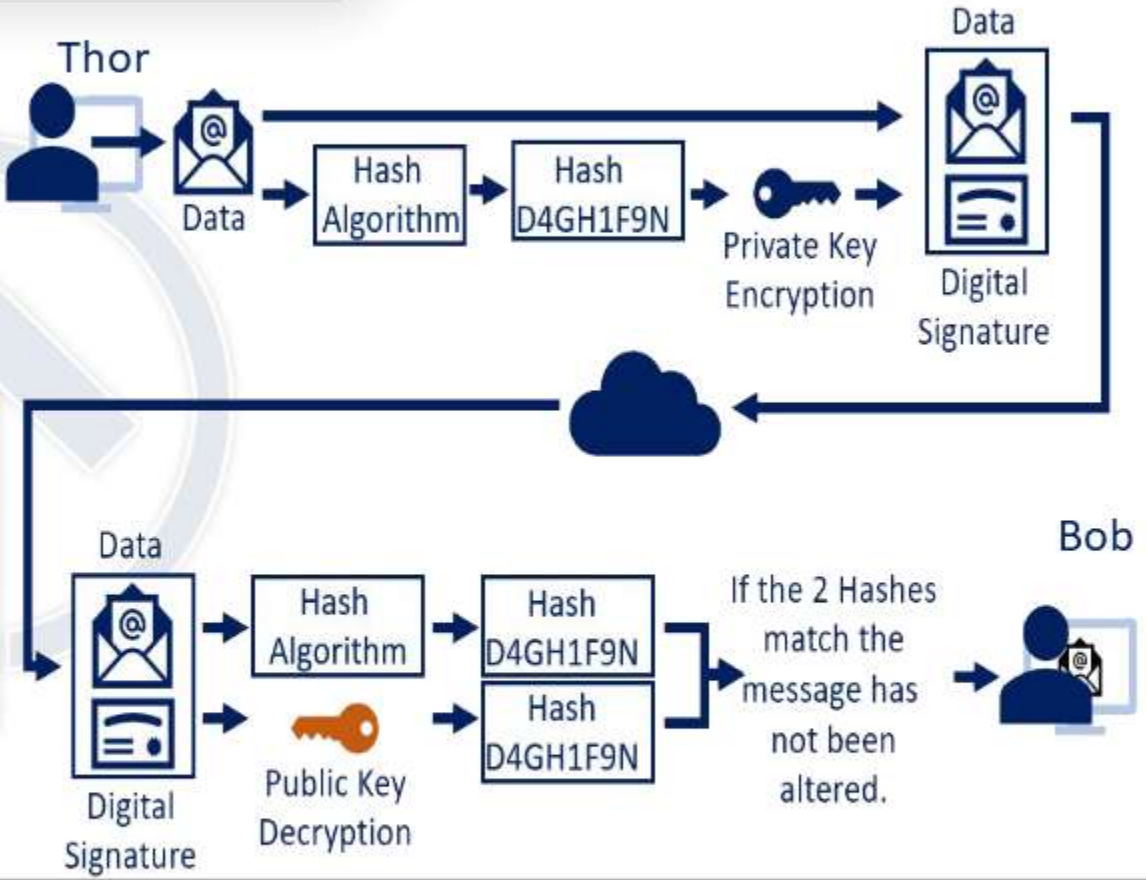
- **Implementing Cryptography:**
  - **Digital Signatures:**
    - **Digital certificates** are public keys signed with a digital signature.
      - Server based - SSL for instance – is assigned to the server (stored on the server).
      - Client based - Digital Signature – is assigned to a person (stored on your PC).
      - **CA** (Certification Authority):
        - Issues and revokes certificates.
        - Can be run internally in your organization or in public (Verisign or GoDaddy, for instance).
      - **ORA** (Organizational Registration Authorities):
        - Done within an organization.
        - Authenticates the certificate holder prior to certificate issuance.

Cryptography

- **Implementing Cryptography:**
  - **Digital Signatures:**
    - **Digital certificates** (continued):
      - **CRL** (Certification Revocation List):
        - Maintained by the CA.
        - Certificates are revoked if a private key is compromised, if an employee leaves the organization, etc.
        - Server side, starting to be replaced by OCSP (client/server side hybrid).
      - **OCSP** (Online Certification Status Protocol):
        - Client/server hybrid, better balance, faster, keeps lists of revoked certificates.
  - The **Clipper chip** was a chipset that was developed and promoted by the United States National Security Agency (NSA) as an encryption device that "secured" voice and data messages. with a built-in backdoor.
    - It was intended to be adopted by telecommunications companies for voice/data transmission but was abandoned after public outcry, and was later found to have many security flaws (it used Skipjack).

Cryptography

- **Implementing Cryptography:**
  - **Digital Signatures:**
    - Provides **Integrity** and **Non-Repudiation**.
    - I want to send an email to Bob.
      - My email is Hashed, the hash is encrypted with my private key (the encrypted Hash is my Digital Signature), I attach the signature to the email and send it.
      - Bob receives it, he generates a hash, and decrypts my signature with my public key. If the hash he generated and the hash he unencrypted match, the email is not altered.

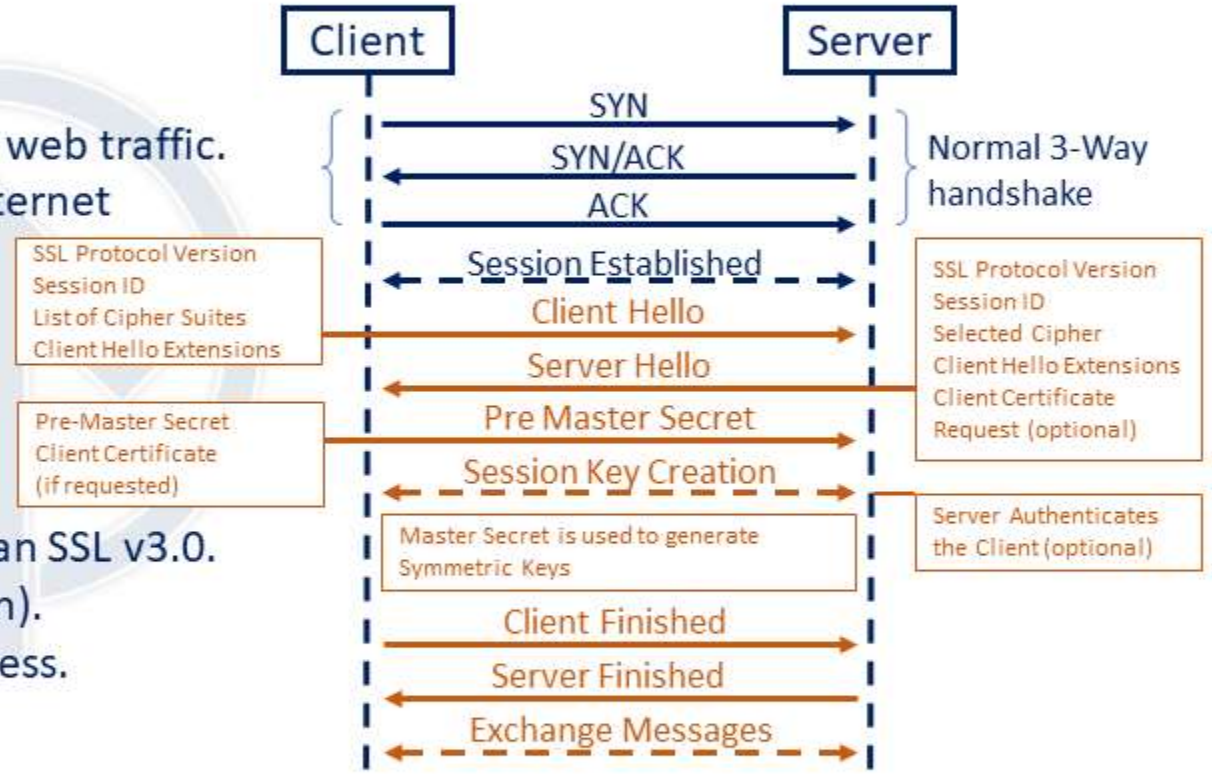


Cryptography

- **Implementing Cryptography:**
  - **MAC** (Message Authentication Code)– The exam uses MAC for several concepts; it will be spelled out which one it is.
    - Hash function using a key.
    - CBC-MAC, for instance, uses Cipher Block Chaining from a symmetric encryption (like DES).
    - Provides integrity and authenticity.
  - **HMAC** (Hashed Message Authentication Code) combines a shared key with hashing.
    - A pre-shared key is exchanged.
    - The sender uses XOR to combine the plaintext with a shared key, then hashes the output using a hashing algorithm (Could be HMAC-MD5 or HMAC-SHA-1).
    - That hash is then combined with the key again, creating an HMAC.
    - The receiver does the same and compares their HMAC with the sender's HMAC.
    - If the two HMACs are identical, the sender is authenticated.

Cryptography

- **Implementing Cryptography:**
  - **SSL** and **TLS** – Confidentiality and Authentication for web traffic.
  - Cryptographic protocols for web browsing, email, internet faxing, instant messaging, and VOIP.
  - You download the server's digital certificate, which includes the site's public key.
    - **SSL** (Secure Socket Layer) - Currently on v3.0.
      - Mostly used for web traffic.
    - **TLS** (Transport Layer Security) - More secure than SSL v3.0.
      - Used for securing web traffic (less common).
      - Used for internet chat and email client access.



- **Preventive and Detective Controls:**
  - **Configuration Management:**
    - When we receive or build new systems they often are completely open, before we introduce them to our environment we harden them.
    - We develop a long list of ports to close, services to disable, accounts to delete, missing patches and many other things.
    - Often it is easier to have OS images that are completely hardened and use the image for the new system, we then update the image when new vulnerabilities are found or patches need to be applied, often though we use a standard image and just apply the missing patches.
    - We do this for any device on our network, servers, workstations, phones, routers, switches ...

- **Preventive and Detective Controls:**
  - **Configuration Management:**
    - Pre introduction into our production environment we run vulnerability scans against the system to ensure we didn't miss anything (Rarely done on workstations, should be done on servers/network equipment).
    - Having a standard hardening baseline for each OS ensures all servers are similarly hardened and there should be no weak links, we also have the standardized hardening making troubleshooting much easier.
    - Once a system is introduced to our production environment we monitor changes away from our security baseline, most changes are administrators troubleshooting or making workarounds, which may or may not be allowed, but it could also be an attacker punching a path out of our network.

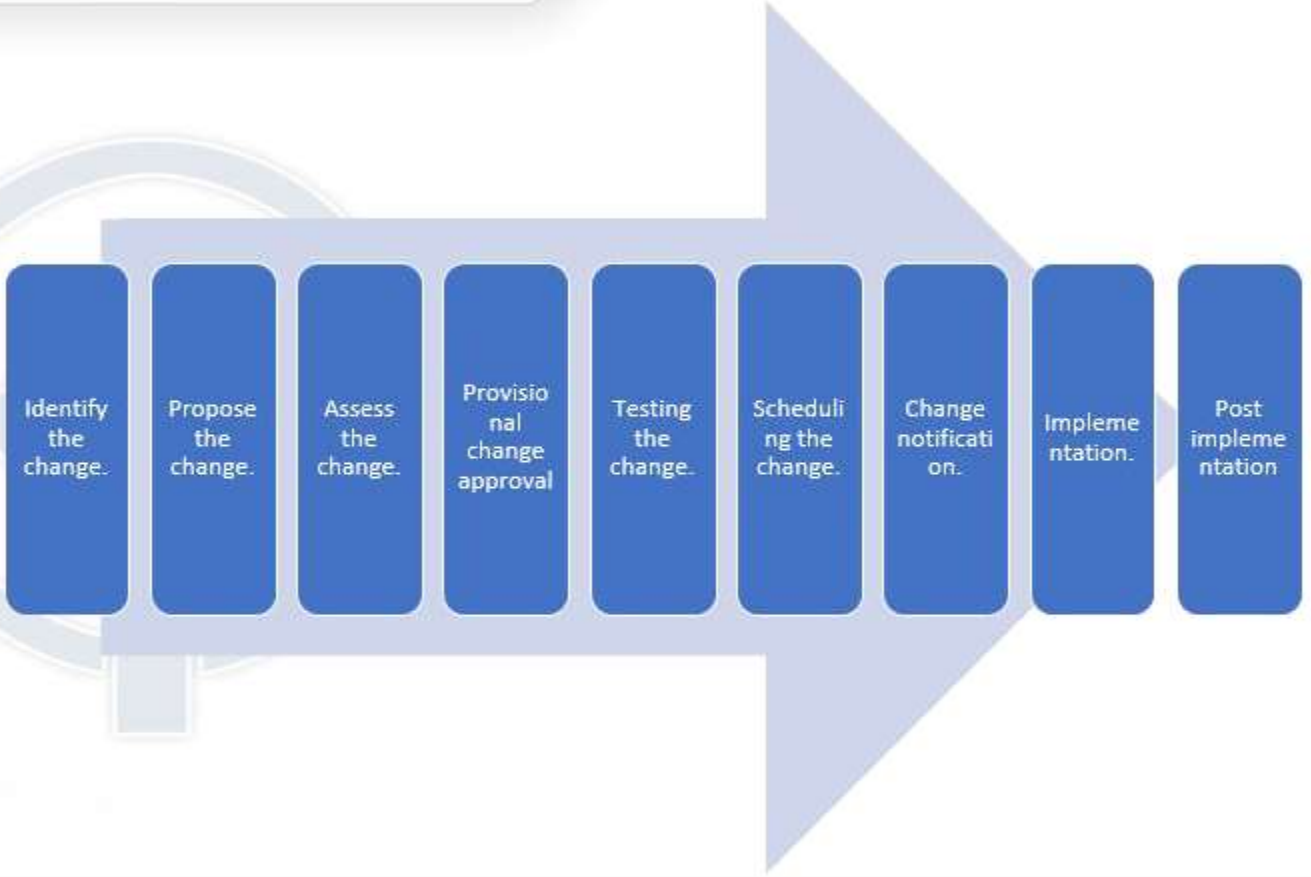
- **Asset Management:**
  - **Patch Management:**
    - In order to keep our network secure we need to apply patches on a regular basis.
    - Whenever a vulnerability is discovered the software producer should release a patch to fix it.
    - Microsoft for instance have "Patch Tuesday" (2nd Tuesday of the month).
      - They release all their patches for that month.
      - If critical vulnerabilities are discovered they push those patches outside of Patch Tuesday.
      - Most organizations give the patches a few weeks to be reviewed and then implement them in their environment.
    - We normally remember the OS patches, but can often forget about network equipment updates, array updates, IoT updates and so on, if they are not patched we are not fully using defense in depth and we can expose ourselves to risk.

- **Asset Management:**
  - **Patch Management:**
    - I have seen places where full rack disk arrays were not encrypted and had not been patched since installation over 10 years prior, the reasoning was poorly designed data storage and updating would take the disks offline for up to an hour, which for the organization was unacceptable.
    - We use software to push our patches to all appropriate systems, this is easier, we ensure all systems gets patched and they all get the same parts of the patch, we may exclude some parts that have an adverse effect on our network.
    - Common tools could be SCCM or WSUS, they do not only push patches, but any software we want to distribute to our organization.
    - We do the pushes after hours to not impact the availability during working hours, normally done Friday or Saturday night somewhere between 01:00 am and 04:00 am.
    - Most places avoid midnight as a lot of backups and jobs run at that time, and end no later than 04:00 am or 05:00 am to ensure systems are online by the start of business the following day.

- **Asset Management:**
  - **Change Management:** 
    - Our formalized process on how we handle changes to our environments.
    - If done right we will have full documentation, understanding and we communicate changes to appropriate parties.
    - The change review board should be comprised of both IT and other operational units from the organization, we may consider impacts on IT, but we are there to serve the organization, they need to understand how it will impact them and raise concerns if they have any.
    - A change is proposed to the change board, they research in order to understand the full impact of the change.
    - The person or group submitting the change should clearly explain the reasons for the change, the pro's and con's of implementing and not implementing, any changes to systems and processes they know about and in general aide and support the board with as much information as needed.
    - The board can have senior leadership on it or they can have a predefined range of changes they can approve and anything above that threshold they would make recommendations but changes require senior leadership approval.

- **Asset Management:**
  - **Change Management:**
    - There are many different models and process flows for change management, some are dependent on organization structure, maturity, field of business and many other factors.
      - A generalized flow would look like this:
        - Identifying the change.
        - Propose the change.
        - Assessing risks, impacts and benefits of implementing and not implementing.
        - Provisional change approval, if testing is what we expect this is the final approval.
        - Testing the change, if what we expected we proceed, if not we go back.
        - Scheduling the change.
        - Change notification for impacted parties.
        - Implementing the change.
        - Post implementation reporting of the actual change impact.

- **Asset Management:**
  - **Change Management:**
    - We closely monitor and audit changes, remember changes can hold residual risk which we would then have to mitigate.
    - Everything in the change control process should be documented and kept, often auditors want to see that we have implemented proper change controls, and that we actually follow the paper process we have presented them with.

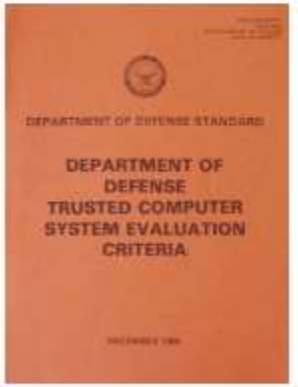


Evaluation Methods, Certification and Accreditation:

- Choosing the security systems and products we implement in our organization can be a daunting task.
- How do we know the vendor is trustworthy, how do we know the systems and products were tested well and what the tests revealed?
- There are many evaluation models in use today.
  - The earliest one, which most security models are based on today is **"The Orange Book"** - The Trusted Computer System Evaluation Criteria – (TCSEC).
  - It was developed by the U.S. Department of Defense in the 1980s. The Orange book was part of a Rainbow Series (or Rainbow Books).
  - The series also had a "The Red Book" Trusted Network Interpretation - (TNI). It addresses Network systems, whereas "The Orange Book" does not address Network Systems.

Rainbow Series

The Orange Book



Evaluation Methods, Certification and Accreditation:

- **ITSEC** (The European Information Technology Security Evaluation Criteria):
  - Was the first successful international model. Contains a lot of references from The Orange Book, but both are retired now.
- **The International Common Criteria** (ISO/IEC 15408):
  - Common Criteria evaluations are performed on computer security products and systems.
  - To be of practical use, the evaluation must verify the target's security features. This is done through the following:
    - **Target Of Evaluation (TOE)** – The product or system that is the subject of the evaluation.
    - **Protection Profile (PP)** – A document which identifies security requirements for a class of security devices. Products can comply with more than one PP. Customers looking for particular types of products can focus on those products certified against the PP that meet their requirements.
    - **Security Target (ST)** – The document that identifies the security properties of the target of evaluation. The ST may have one or more PP's.

Evaluation Methods, Certification and Accreditation:

- **The International Common Criteria** (ISO/IEC 15408):
  - **Evaluation Assurance Level (EAL)** – How did the system or product score on the testing?
    - EAL Level 1-7:
      - EAL1: Functionally Tested.
      - EAL2: Structurally Tested.
      - EAL3: Methodically Tested and Checked.
      - EAL4: Methodically Designed, Tested and Reviewed
      - EAL5: Semi-formally Designed and Tested.
      - EAL6: Semi-formally Verified Design and Tested.
      - EAL7: Formally Verified Design and Tested.

- **Assessment and test strategies.**
  - **Security Assessments:**
    - A full picture approach to assessing how effective our access controls are, they have a very broad scope.
    - Security assessments often span multiple areas, and can use some or all of these components:
      - Policies, procedures, and other administrative controls.
      - Assessing the real world-effectiveness of administrative controls.
      - Change management.
      - Architectural review.
      - Penetration tests.
      - Vulnerability assessments.
      - Security audits.

- **Assessment and test strategies.**
  - **Security audit:** A test against a published standard.
    - SOC 2 Type 1 or 2, PCI-DSS, …
      - SOC 2 Type 1 report on management's description of a service organization's system and the **suitability** of the design of controls.
      - SOC 2 Type 2 report on management's description of a service organization's system and the suitability of the design and operating **effectiveness** of controls.
    - Purpose is to validate/verify that an organization meets the requirements as stated in the published standard.
  - **Internal and 3rd-Party Audits:**
    - **Structured audits** (3rd party):
      - External auditors who validate our compliance, they are experts and the audit adds credibility.
      - Can also be a knowledge transfer for the organization, required annually in many organizations.
    - **Unstructured audits:**
      - Internal auditors to improve our security and find flaws, often done before an external audit.

- **Assessment and test strategies.**
  - **Security Audit Logs:**
    - Reviewing security audit logs in an IT system is one of the easiest ways to verify that access control mechanisms are working as intended.
    - Reviewing audit logs is primarily a detective control.
    - NIST Special Publication 800-92 suggests the following log types should be collected and audited:
      - **Network Security Software/Hardware:**
        - Antivirus logs, IDS/IPS logs, remote access software (such as VPN logs), web proxy, vulnerability management, authentication servers, routers and firewalls.
      - **Operating System:**
        - System events, audit records, applications, client requests and server responses, usage information, significant operational actions.

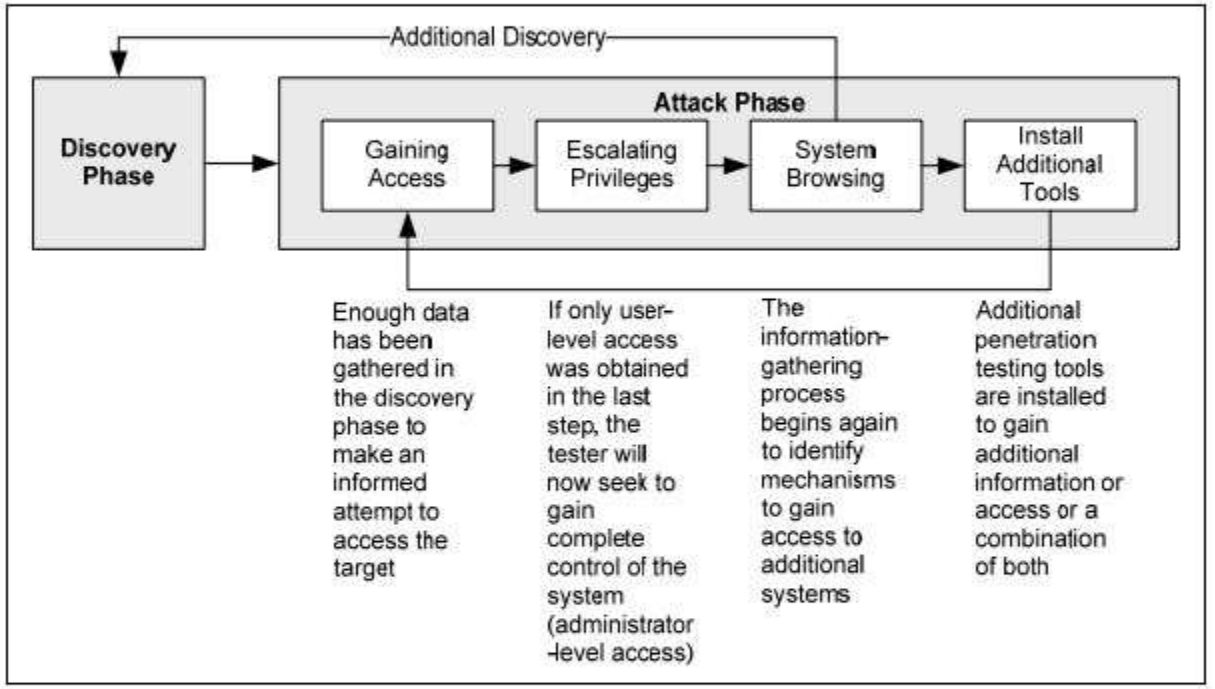
- **Assessment and test strategies.**
  - **Security Audit Logs:**
    - **Centralized Logging:**
      - Should be automated, secure and even administrators should have limited access.
      - Often a central repository is hashed and never touched, and a secondary copy is analyzed to ensure integrity.
      - Logs should have a retention policy to ensure we are compliant and we keep the logs as long as we need them.
      - Checking logs is often an afterthought and rarely done, where do we start?
      - Since they are often keeping everything, there can be 10's of millions of lines of log info, we need to implement systems to automate this as much as makes sense.

- **Assessment and test strategies.**
  - **Security Audit Logs** (Audit trail):
    - Audit record management typically faces five distinct problems:
      - Logs are not reviewed on a regular and timely basis.
      - Audit logs and audit trails are not stored for a long enough time period.
      - Logs are not standardized or viewable by correlation toolsets - they are only viewable from the system being audited.
      - Log entries and alerts are not prioritized.
      - Audit records are only reviewed for the bad stuff.

- **Assessment and test strategies.**
  - **Vulnerability scanning/testing:**
    - A vulnerability scanner tool is used to scan a network or system for a list of predefined vulnerabilities such as system misconfiguration, outdated software, or a lack of patching.
    - It is very important to understand the output from a vulnerability scan, they can be 100's of pages for some systems, and how do the vulnerabilities map to Threats and Risks (Risk = Threat x Vulnerability).
    - When we understand the true Risk, we can then plan our mitigation.
    - Common vulnerability scanners could be Nessus or OpenVAS, both list vulnerabilities in Critical, High, Medium, Low, and Informational.

- **Assessment and test strategies.**
  - **Penetration Testing** (Pen Testing), often called Ethical Hacking.
    - Test if the vulnerabilities are exploitable.
    - An authorized simulated attack on our organization that looks for security weaknesses, potentially gaining access to the systems, buildings and data.
    - It is very important to have very clear rules of engagement defined in a SOW (Statement Of Work)
      - Which IP ranges, time frame, tools, POC, how to test, what to test, …
      - We confirm with our legal team before hiring Pen Testers, even if you allow it what they do may still be illegal.
    - Senior management set the goals for the Pen testing.
      - Why are we doing it? What are we trying to achieve? They have to sign off on it.
    - If we are the pen testers, we are there to test and document the vulnerabilities, not to fix them.
      - We provide the report to senior management and they decide which vulnerabilities they want to address.
    - Use multiple attack vectors and Pen testing uses an iterative process that is similar to Agile project planning.

- **Assessment and test strategies.**
  - **Penetration Testing:**
    - **Discovery** (planning): Finding the vulnerabilities, design the attacks.
    - **Gaining Access**: Access the network.
    - **Escalate Privileges**: Get higher level access, ultimately we want admin access.
    - **System Browsing**: Gain additional access, often back to discovery again with our new knowledge level and access.
    - **Install Additional tools**: With our elevated access we can install more tools and exploit new attack surfaces, can go back to Gaining Access.
    - Finally when done, they report the findings.



- **Assessment and test strategies.**
  - **Penetration Testing:**
    - Planning > Reconnaissance > Scanning (enumeration) > Vulnerability assessment > Exploitation > Reporting.
    - Very similar to a black hat methodology.
    - Black hats often spend less time planning, and instead of reporting they cover their tracks.
      - They delete/modify logs and any other tracks they left and if possible install backdoors, so they can keep exploiting our environments.
    - A Pen tester has a very clear SOW and they do not compromise system and data integrity.
    - The Pen tester may also not be allowed to access certain files (PII/PHI), but a dummy file is created in the same location, if the Pen tester can get to the target file, they could get to the actual data file.
    - The Pen testing is done in clearly defined time windows, often in maintenance windows after hours, the point is to prove we are vulnerable, not disrupt our business.
    - Some low impact Pen tests can also be done on DR environments, to not effect our live environments, but they are often less useful since most DR environments are not a mirror copy of the production environment.

- **Assessment and test strategies.**
  - **Penetration Testing:**
    - **Black box** Pen testing (Zero Knowledge):
      - The attacker had no knowledge about the organization other than publicly available information.
      - They start from the point an external attacker would.
    - **White box** (Crystal/Clear) Pen testing: (Full Knowledge):
      - The attacker has knowledge of the internal network and access to it like a privileged employee would.
      - Normally Administrator access employee with full knowledge of our environment.
    - **Gray (Grey) box** (Partial Knowledge) Pen testing:
      - The attacker has limited knowledge, a normal user, vendor or someone with limited environment knowledge.
    - Do not confuse these with Black, Gray, or White Hat Hackers.
      - **White Hat hackers:** Professional Pen Testers trying to find flaws so we can fix it (Ethical Hackers).
      - **Black Hat hackers:** Malicious hackers, trying to find flaws to exploit them (Crackers - they crack the code).
      - **Gray/Grey Hat hackers:** They are somewhere between the white and black hats.

- **Assessment and test strategies.**
  - **Penetration Testing:**
    - Think like an attacker would, start with the easiest attack first, the users.
    - Low technical tools can be just as effective as sophisticated tools,
    - Many organizations have strong perimeter defense, but no defense in depth, once you get past 1 or 2 barriers you can access most things.
    - **Social engineering** uses people skills to bypass security controls.
      - Can be used in a combination with many other attacks, especially client-side attacks or physical tests.
      - Attacks are often more successful if they use one or more of these approaches:
        - **Authority** (someone you trust or are afraid of) - Look and sound like an authority figure, be in charge, this can be in a uniform or a suit. Most effective with impersonation, whaling, and vishing attacks.
        - **Intimidation** (If you don't bad thing happens) - Virus on the network, credit card compromised, lawsuit against your company, intimidation is most effective with impersonation and vishing attacks.

- **Assessment and test strategies.**
  - **Penetration Testing:**
    - **Social engineering attacks:**
      - **Consensus** (Following the crowd, everyone else was doing it) - Fake reviews on a website, using consensus/social proof is most effective with Trojans and hoaxes.
      - **Scarcity** (If you don't act now, it is too late) - New iPhone out, only 200 available, often effective with phishing and Trojan attacks.
      - **Urgency** (It has to happen now or else) - The company will be sued for $1,000,000 if these papers are not filled out before Friday, often used with Phishing.
      - **Familiarity** (Have a common ground, or build it) - Knowing something about the victim ahead of time and then reference it can raises chances of a successful attack drastically. People want to be helpful, if they feel like they know you they want to even more. Often successful with vishing and in-person social engineering.

- **Assessment and test strategies.**
  - **Penetration Testing:**
    - **War dialing:**
      - Uses modem to dial a series of phone numbers, looking for an answering modem carrier tone, the penetration tester then attempts to access the answering system.
      - Not really done anymore, but know it for the exam.
    - **War driving** (access point mapping):
      - Driving or walking around, mapping access points and trying to gain access to them.
    - **Network attacks**
      - Client-side attacks, server-side attacks, or Web application attacks.
    - **Wireless tests:**
      - Evaluate the risk related to potential access to your wireless network.
      - Uses the password combination & sniffing technique for cracking unsecured wireless network, so a proper set up is required for making the whole process semi-automated and automated.

- **Assessment and test strategies.**
  - **Penetration Testing:**
    - **Penetration Testing Tools and Methodology:**
      - Just like hackers, Pen testers use many different tools to test, both published tools and own creations.
      - Be VERY careful if testing these out, do not use them outside your own network, and only on internal networks with written permission.
      - Penetration testing tools:
        - Open source Metasploit - http://www.metasploit.org/
        - Closed source Core Impact - http://www.coresecurity.com/
        - Immunity Canvas - http://www.immunitysec.com/
        - Top 125 Network Security Tools - http://sectools.org/
        - Kali Linux - https://www.kali.org/

- **Assessment and test strategies.**
  - **Penetration Testing:**
    - Real-time map of detected hacks.
    - http://map.norsecorp.com/#/

- **Assessment and test strategies.**
  - **Software testing:**
    - Historically we have built functional software and tested it for just that stability and functionality, security has been an afterthought if considered at all. Software needs to be designed securely, built in not bolted on.
    - Normal software can have millions of line of code and about 1% of that contains vulnerabilities.
    - Many security breaches happen because our software is easy to compromise.
    - **Static testing** - Passively testing the code, it is not running.
      - This is walkthroughs, syntax checking, and code reviews.
      - Looks at the raw source code itself looking for evidence of known insecure practices, functions, libraries, or other characteristics having been used in the source code.
      - There are 100's of static code analysis tools available depending on programming language.
    - **Dynamic testing** – Actively testing the code while executing it.
      - Can uncover flaws that exist in the particular implementation and interaction of code that static analysis missed. Software can run and code execute with flaws.

- **Assessment and test strategies.**
  - **Software testing:**
    - Code testing uses white and black box terms just like in Pen testing.
      - **White box software testing:**
        - The tester has full access to program source code, data structures, variables, ...
      - **Black box software testing:**
        - The tester has no details, just the software, they then test for functionality and security flaws.

- **Assessment and test strategies.**
  - **Software testing:**
    - **TM/RTM** (Requirements Traceability Matrix):
      - Normally a table, used to map customer requirements to the testing plan using a many-to-many relationship comparison.
      - A requirements traceability matrix may be used to check if the current project requirements are being met, and to help in the creation of a request for proposal, software requirements specification, various deliverable documents, and project plan tasks.

Test Cases	321	3	2	3	1	1	
Tested Implicitly	77						
1.1.1	1	x					
1.1.2	2		x	x			
1.1.3	2	x					
1.1.4	1			x			
1.1.5	2	x					
...							

- **Assessment and test strategies.**
  - **Software testing:**
    - **Software Testing levels:**
      - **Unit testing:**
        - Tests that verify the functionality of a specific section of code.
        - In an object-oriented environment, this is usually at the class level, and the minimal unit tests include the constructors and destructors.
        - Usually written by developers as they work on code (white-box), to ensure that the specific function is working as expected.
      - **Integration testing:**
        - Seeks to verify the interfaces between components against a software design.
        - Integration testing works to expose defects in the interfaces and interaction between integrated components/modules.
        - Progressively larger groups of software components are tested until the software works as a system.

- **Assessment and test strategies.**
  - **Software testing:**
    - **Software Testing levels:**
      - **Component interface testing:**
        - Testing can be used to check the handling of data passed between various units, or subsystem components, beyond full integration testing between those units.
        - Tests a completely integrated system to verify that the system meets its requirements.
      - **Operational acceptance:**
        - Used to conduct operational readiness (pre-release) of a product, service or system as part of a quality management system.

- **Assessment and test strategies.**
  - **Software testing:**
    - **Software Testing types:**
      - **Installation testing:**
        - Assures that the system is installed correctly and working at actual customer's hardware.
      - **Regression testing:**
        - Finding defects after a major code change has occurred.
        - Looks for software regressions, as degraded or lost features, including old bugs that have come back.

- **Assessment and test strategies.**
  - **Software testing:**
    - **Software Testing types:**
      - **Fuzzing (Fuzz testing):**
        - Testing that provides a lot of different inputs in order to try to cause unauthorized access or for the application to enter unpredictable state or crash.
        - If the program crashes or hangs the fuzz test failed.
        - The Fuzz tester can enter values into the script or use pre-compiled random or specific values.
        - **Mutating fuzzing** – The tester analyses real info and modifies it iteratively.

- **Assessment and test strategies.**
  - **Software testing:**
    - **Software Testing types:**
      - **All-pairs testing (Pairwise testing):**
        - All-Pairs Testing is defined as a black-box test design technique in which test cases are designed to execute all possible discrete combinations of each pair of input parameters.
        - The most common bugs in a program are generally triggered by either a single input parameter or an interaction between pairs of parameters.
        - It uses carefully chosen test vectors, this can be done much faster than an exhaustive search of all combinations of all parameters, by parallelizing the tests of parameter pairs.
        - If we have a very simple piece of software with 3 input parameters:
        - Server type A: Physical B: VM Vendor: A: Dell B: HP Serial number: A Valid (5000) B Invalid
          - If we test all possible combinations we would test 2x2x5000 = 20000 combinations.
          - If we test all-pairs we would test 2x2x2 = 8 combinations – we only look at valid or invalid input.

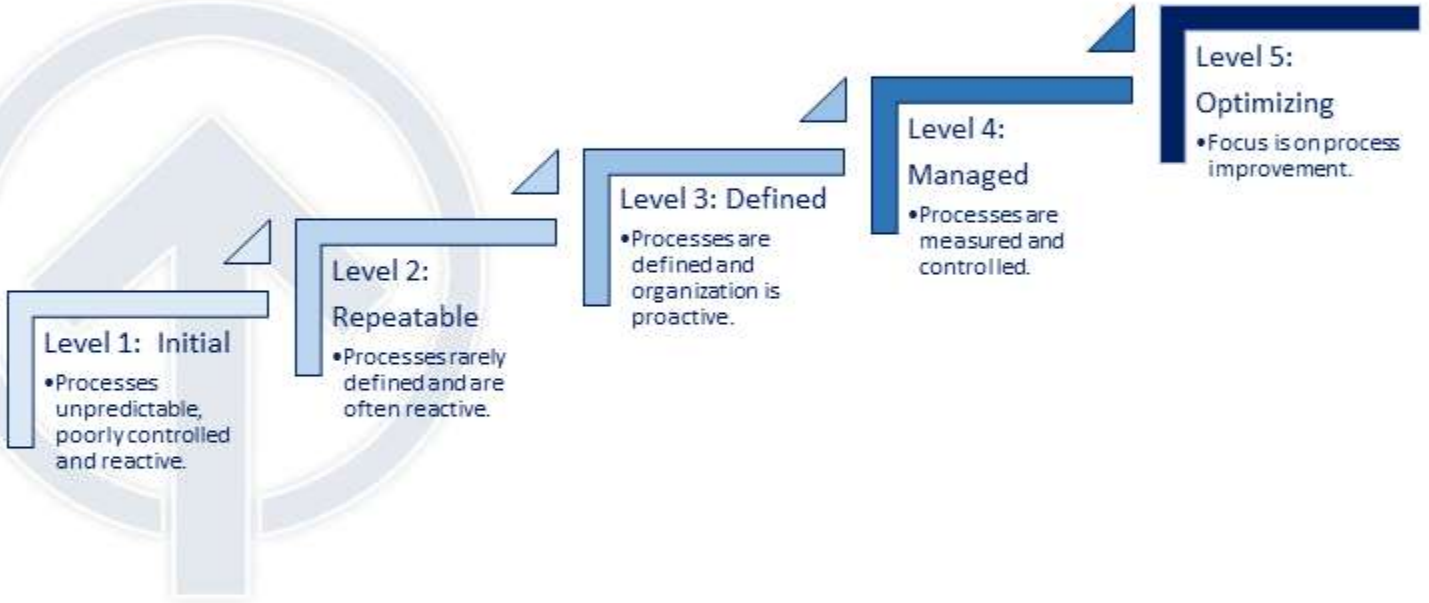
- **Assessment and test strategies.**
  - **Software testing:**
    - **Software Testing types:**
      - **Misuse Case Testing:**
        - Executing a malicious act against a system, attackers won't do what normal users would, we need to test misuse to ensure our application or software is safe.
      - **Test Coverage Analysis:**
        - Identifies how much of the code was tested in relation to the entire application.
        - To ensure there are no significant gaps where a lack of testing could allow for bugs or security issues to be present that otherwise should have been discovered.
        - With 50+ millions line of code in a Windows OS, often spot checks on critical areas are only enforced.

- **Assessment and test strategies.**
  - **Software testing:**
    - Now that we have completed our tests, just like on our log reviews, we need to use it and analyze the data we got from the testing.
    - It can be huge amounts of data, and we need to prioritize what we act on first, what is acceptable and what is not.
    - Think of the qualitative risk analysis, if it is low likelihood and low impact we may leave it alone and focus on higher priority items.

- **Software vulnerabilities and Attacks**
  - **CMM** (Capability Maturity Model):
    - The maturity relates to the degree of formality and optimization of processes, from ad hoc practices, to formally defined repeatable steps, to managed result metrics, to active optimization of the processes.
    - There are five levels defined in the model and, which describes where an organization is, it also has practical steps to how to mature the organization to get to the next level.
    - **Level 1: Initial**
      - Processes at this level are normally undocumented and in a state of dynamic change, tending to be driven in an ad hoc, uncontrolled and reactive manner by users or events.
      - This provides a chaotic or unstable environment for the processes.
    - **Level 2: Repeatable**
      - This level of maturity that some processes are repeatable, possibly with consistent results.
      - Process discipline is unlikely to be rigorous, but where it exists it may help to ensure that existing processes are maintained during times of stress.

- **Software vulnerabilities and Attacks**
  - **CMM:**
    - **Level 3: Defined**
      - This level that there are sets of defined and documented standard processes established and subject to some degree of improvement over time.
      - These standard processes are in place.
      - The processes may not have been systematically or repeatedly utilized enough for the users to become competent or the process to be validated in a range of situations.
    - **Level 4: Managed (Capable)**
      - Processes at this level uses process metrics, effective achievement of the process objectives can be evidenced across a range of operational conditions.
      - The suitability of the process in multiple environments has been tested and the process refined and adapted.
      - Process users have experienced the process in multiple and varied conditions, and are able to demonstrate competence.

- **Software vulnerabilities and Attacks**
  - **CMM:**
    - **Level 5: Optimizing**
      - Processes at this level focus on continually improving process performance through both incremental and innovative technological changes/improvements.
      - Addressing statistical common causes of process variation and changing the process to improve process performance.



- **Software vulnerabilities and Attacks**
  - **CMM:**
    - **Acceptance testing:**
      - There are many different testing types we use throughout the development lifecycle.
      - At the end of development we also use acceptance testing, we need to test it to ensure it does what it is supposed to and it is robust and secure.
      - **The User Acceptance test:**
        - Is the software functional for the users who will be using it? it is tested by the users and application managers.
      - **Operational acceptance testing:**
        - Does the software and all of the components it interacts with ready requirements for operation.
        - Tested by system administrators are the backups in place, do we have a DR plan, how do we handle patching, is it checked for vulnerabilities, ...?

- **Software vulnerabilities and Attacks**
  - **CMM:**
    - **Acceptance testing:**
      - **Contract Acceptance testing:**
        - Does the software fulfil the contract specifications? The what/where/how of the acceptance is defined in the contract.
      - **Compliance acceptance testing:**
        - Is the software compliant with the rules, regulations and laws of our industry?
      - **Compatibility/production testing:**
        - Does the software interface as expected with other applications or systems?
        - Does the software perform as expected in our production environment vs. the development environment?

- **Software vulnerabilities and Attacks**
  - **Buying software from other companies:**
    - When we buy software from vendors either COTS (Commercial Off The Shelf) or custom built software we need to ensure it is as secure as we need it to be.
    - Vendors claims of security posture should until proven be seen as marketing claims.
    - We need to do our due care and due diligence, as well as use outside council if needed.
    - Many organizations deal with C-level executives going to conferences and buying software that the organization may not want or need.
    - Software development and procurement as well as any other project should be carefully scoped, planned be based on a clear analysis of what the business needs and wants.

- **Software vulnerabilities and Attacks**
  - **Buying software from other companies:**
    - **COTS** (Commercial Off-the-Shelf) Software:
      - When buying COTS software we can, depending on how widely the software is used, look at reviews, talk to current customers and users to get a clearer understanding of the software capabilities and security.
      - Software roadmaps are nice, but only buy the software for what it can actually do now, not what it can maybe do in the future.
      - We can use a clear RTM (requirements traceability matrix), requirements are divided into "Must have, nice to have and maybe should have".
      - We would then score the software candidates on the "Have's" and from that we should be able to see feasible candidates, other factors such as cost, maintenance also play a big part in the decision.
      - For large/expensive implementations it may also be possible for the vendor to provide references to talk to.
      - We would also look at how financially sound the vendor looks to be, if we spend $2,000,000 on software and the vendor goes out bankrupt in 3 months, we may have to spend another $2,000,000 all over again.

- **Software vulnerabilities and Attacks**
  - **Buying software from other companies:**
    - **Custom-Developed Third Party Products:**
      - Having someone else develop the software we need is also an option.
      - This is higher cost than COTS software, but also far more customizable.
      - The same questions and then some should be asked:
      - How good are they? Have they done this before? How secure are they?, ...
      - Do we own the code or do we rent it when it is done?
      - What happens if they go out of business?
      - Who will support it?
      - Do you have capable staff, that can support and tweak the software?
      - Is it secure or is it security through obscurity?
      - Many code shops are just that, only code shops, once the software is accepted it is your problem to do the day to day maintenance, they may contract for updates, but that is it.

What we covered in Domain 3

- Access control. - MAC, DAC, RBAC, ABAC
- Objects and subjects.
- IAAA - Identification and Authentication, Authorization and Accountability
- Type 1, 2, and 3 authentication - Something you know, something you have, and something you are
- The history of cryptography, symmetric/asymmetric encryption, hashing, and digital signatures
- Patch management, configuration management, and change management
- Security assessment and security audits.
- Social engineering.
- Software testing.
- Buying software from other companies.
- Data remanence and destruction.

- Thank you for staying here with me!