CISM®

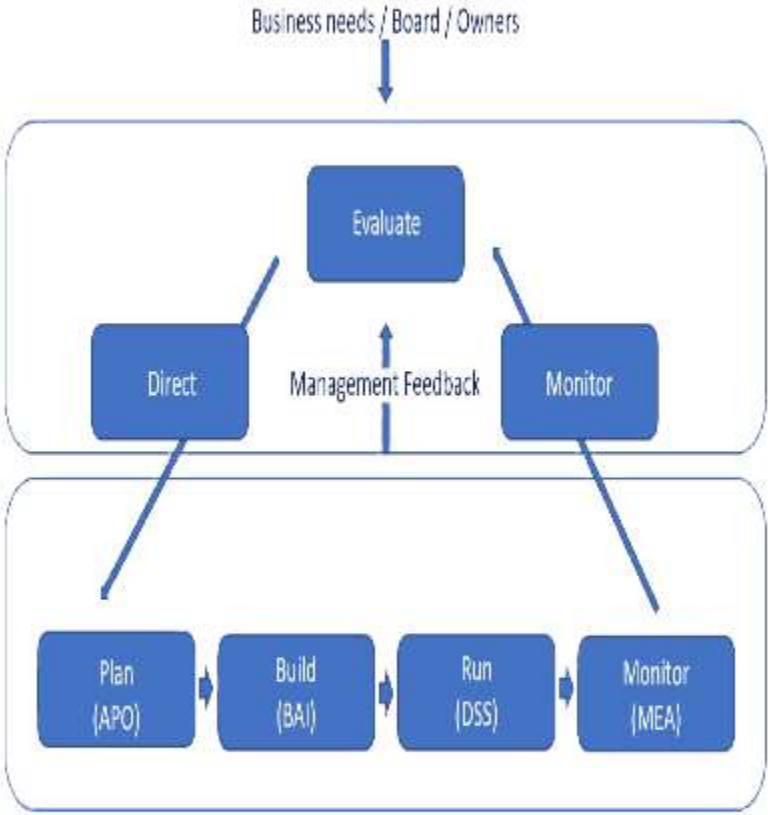
DOMAIN 1

Welcome to the first CISM Domain.

- This chapter is **VERY** important because:
  - Every other knowledge domain builds on top of this chapter
  - This is the **foundation**.
- 24% of the exam questions on the certification are from this domain.
- We will be covering our ethics, values, vision and our mission.
- We look at the policies, the procedures, the laws we need to adhere to.
- SWOT and GAP analysis.
- OpEx, CapEx, Fiscal years, KGIs, KPIs, and KRIs
- Secure software design.

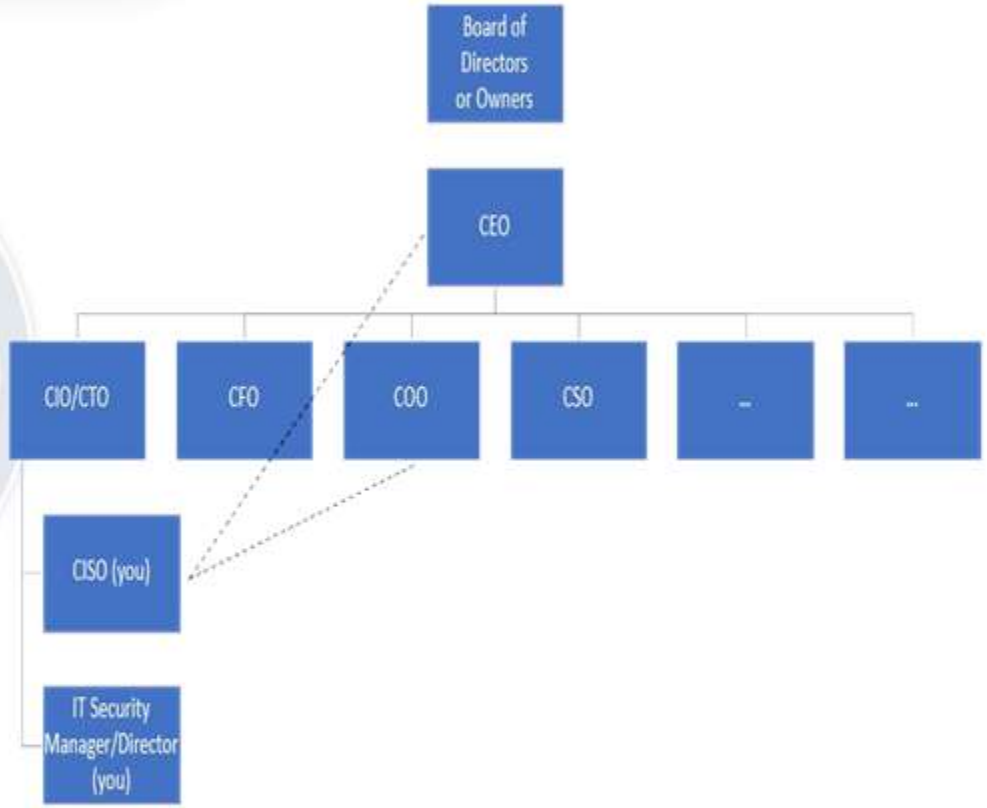
Security governance principles.

- **Governance vs. Management**
  - **Governance** – This is C-level Executives (Not you).
    - Stakeholder needs, conditions and options are evaluated to define:
      - Balanced agreed-upon enterprise objectives to be achieved.
      - Setting direction through prioritization and decision making.
      - Monitoring performance and compliance against agreed-upon direction and objectives.
      - Risk appetite – Aggressive, neutral, adverse.
  - **Management** – How do we get to the destination (This is you).
    - Plans, builds, runs and monitors activities in alignment with the direction set by the governance to achieve the objectives.
    - Risk tolerance – How are we going to practically work with our risk appetite and our environment.



Security governance principles.

- **Bottom-Up:** IT Security is seen as a nuisance and not a helper, this often changes when a security breach happens.
- **Top-Down:** IT leadership understands the importance of IT Security, they lead and set the direction. (The exam).
- **C-Level Executives (Senior Leadership) They are ultimately liable.**
  - **CEO**: Chief Executive Officer.
  - **CSO**: Chief Security Officer.
  - **CIO**: Chief Information Officer.
  - **CFO**: Chief Financial Officer.
  - Normal organizations obviously have more C-Level executives, the ones listed here you need to know.
  - Also know where you fit in the organization and on the exam.



Security governance principles.

- **Governance standards and control frameworks.**
  - **PCI-DSS** - Payment Card Industry Data Security Standard (While a standard it is required: more on this one later).
  - **OCTAVE®** - Operationally Critical Threat, Asset, and Vulnerability Evaluation.
    - **Self Directed** Risk Management.
  - **COBIT** - Control Objectives for Information and related Technology.
    - **Goals** for IT – Stakeholder needs are mapped down to IT related goals.
  - **COSO** – Committee Of Sponsoring Organizations.
    - **Goals** for the entire organization.
  - **ITIL** - Information Technology Infrastructure Library.
    - IT Service Management **(ITSM).**
  - **FRAP** - Facilitated Risk Analysis Process.
    - Analyses one business unit, application or system at a time in a roundtable brainstorm with **internal** employees. Impact analyzed, threats and risks prioritized.

Security governance principles.

- **Governance standards and control frameworks.**
  - **ISO 27000 series**:
    - **ISO 27001:** Establish, implement, control and improvement of the ISMS. Uses PDCA (Plan, Do, Check, Act)
    - **ISO 27002:** (From BS 7799, 1/2, ISO 17799) Provides practical advice on how to implement security controls. It has 10 domains it uses for **ISMS** (Information Security Management Systems).
    - **ISO 27004:** Provides metrics for measuring the success of your ISMS.
    - **ISO 27005:** Standards based approach to risk management.
    - **ISO 27799:** Directives on how to protect PHI (Personal Health Information).

Links on all these as well as ones from previous slides in the “Extras” lecture.

Security governance principles.

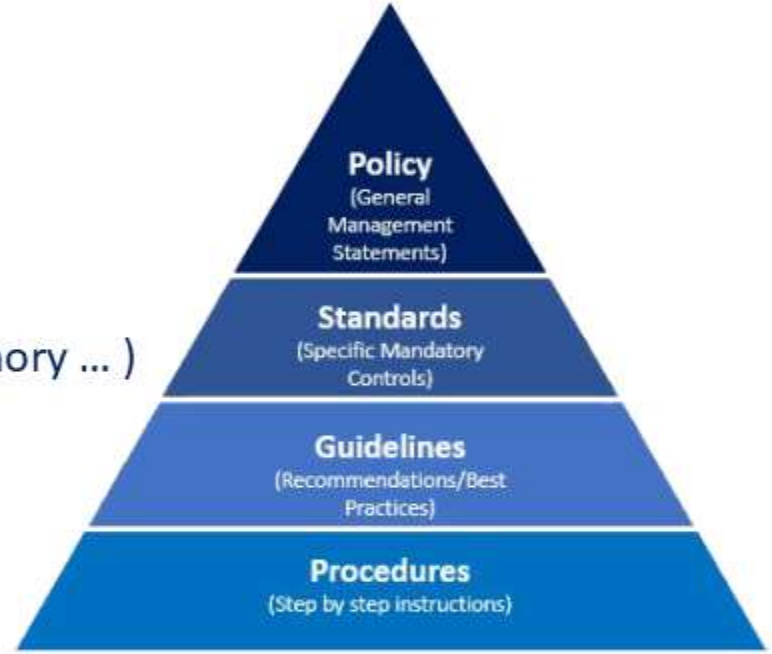
- **Defense in Depth** – Also called Layered Defense or Onion Defense.
  - We implement multiple overlapping security controls to protect an asset.
  - This applies both to physical and logical controls.
    - To get to a server you may have to go through multiple locked doors, security guards, man traps.
    - To get to data you may need to get past firewalls, routers, switches, the server, and the applications security.
    - Each step may have multiple security controls.
  - No single security control secures an asset.
  - By implementing Defense in Depth you improve your organization's Confidentiality, Integrity and Availability.

Security governance principles.

- **Values:**
  - What do are our values? Ethics, Principles, Beliefs.
- **Vision:**
  - What do we aspire to be? Hope and Ambition.
- **Mission:**
  - Who do we do it for? Motivation and Purpose.
- **Strategic Objectives:**
  - How are we going to progress? Plans, goals, Sequencing.
- **Action & KPI's**
  - What do we need to do and how do we know when we achieved it? Actions, Recourses, Outcomes, Owners, and Timeframes.

Information Security Governance:

- **Policies – Mandatory.**
  - High level, non-specific.
  - They can contain "Patches, updates, strong encryption"
  - They will not be specific to "OS, encryption type, vendor Technology"
- **Standards – Mandatory.**
  - Describes a specific use of technology (All laptops are W10, 64bit, 8gig memory …)
- **Guidelines – non-Mandatory.**
  - Recommendations, discretionary – Suggestions on how you would to do it.
- **Procedures – Mandatory.**
  - Low level step-by-step guides, specific.
  - They will contain "OS, encryption type, vendor Technology"
- **Baselines (Benchmarks) - Mandatory.**
  - Benchmarks for server hardening, apps, network. Minimum requirement, we can implement stronger if needed.



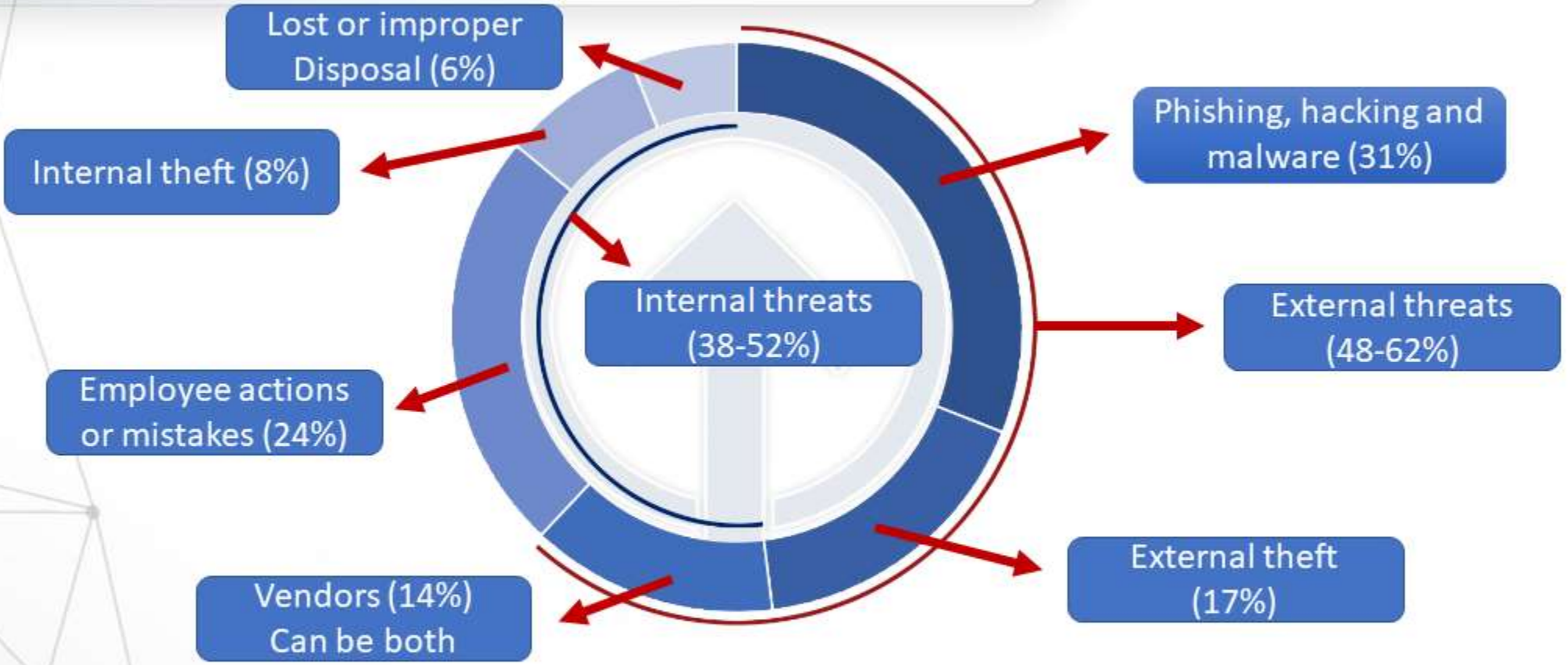
Information Security Governance:

- **Personnel Security – Users often pose the largest security risk:**
  - **Awareness** – Change user behavior - this is what we want, we want them to change their behavior.
  - **Training** – Provides users with a skillset - this is nice, but if they ignore the knowledge, it does nothing.
  - **Hiring Practices** – We do background checks where we check: References, degrees, employment, criminal, credit history (less common, more costly). We have new staff sign a NDA (Non-Disclosure Agreement).
  - **Employee Termination Practices** – We want to coach and train employees before firing them. They get warnings.
    - When terminating employees, we coordinate with HR to shut off access at the right time.
  - **Vendors, Consultants and Contractor Security.**
    - When we use outside people in our environments, we need to ensure they are trained on how to handle data. Their systems need to be secure enough for our policies and standards.
  - **Outsourcing and Offshoring** - Having someone else do part of your (IT in our case) work.
    - This can lower cost, but a thorough and accurate Risk Analysis must be performed. Offshoring can also pose problems with them not having to comply with the same data protection standards.

CISM: Certified Information Security Manager

Domain 1: Information Security Governance.

ThorTeaches.com



SWOT (Strengths, Weaknesses, Opportunities, and Threats):

- **Internal factors:**
  - **Strengths:** What we do well, skilled staff, assets, and advantages over competitors.
  - **Weaknesses:** Things we are missing, resource limitations.
    - Human resources , physical resources, financials, activities and processes, and past experiences.
- **External factors:**
  - **Opportunities:** Elements in the environment that the business or project could exploit to its advantage.
  - **Threats:** Elements in the environment that could cause trouble for the business or project.
    - Future trends, the economy, funding, our physical environment, legislation, national, or international events

Gap analysis:

- **Identify the existing process:**
  - What are we doing?
- **Identify the existing outcome:**
  - How well do we do it?
- **Identify the desired outcome:**
  - How well do we want to do?
- **Identify and document the gap:**
  - What is the difference between now and desired result?
- **Identify the process to achieve the desired outcome:**
  - How can we possibly get to the desired result?
- **Develop the means to fill the gap:**
  - Build the tool or processes to get the result.
- **Develop and prioritize Requirements to bridge the gap.**

Area	Current state	Target state	Difference	Action plan	Priority
Lemonade	$10 per day	$30 per day	$20	Build new stand	High
Cookies	$0 per day	$25 per day	$25	Add to lemonade stand	Medium

Organizational finances.

- **OPEX vs. CAPEX:**
  - **OPEX** (Operating Expense) is the ongoing cost for running a product, business, or system. (Keeping the lights on).
  - **CAPEX** (Capital Expenditure) is the money a company spends to buy, maintain, or improve its fixed assets, such as buildings, vehicles, equipment, or land.
- **Business plans, road-maps:**
  - We build our organizational business plans based on the organizations mission statement and vision at the direction of senior leadership.
  - We have 1-year, 3-year, and 5-year business plans and roadmaps.
- **Fiscal years (budget year):**
  - We plan our budgets according to our organizations fiscal year.

KGI's, KPI's, and KRI's:

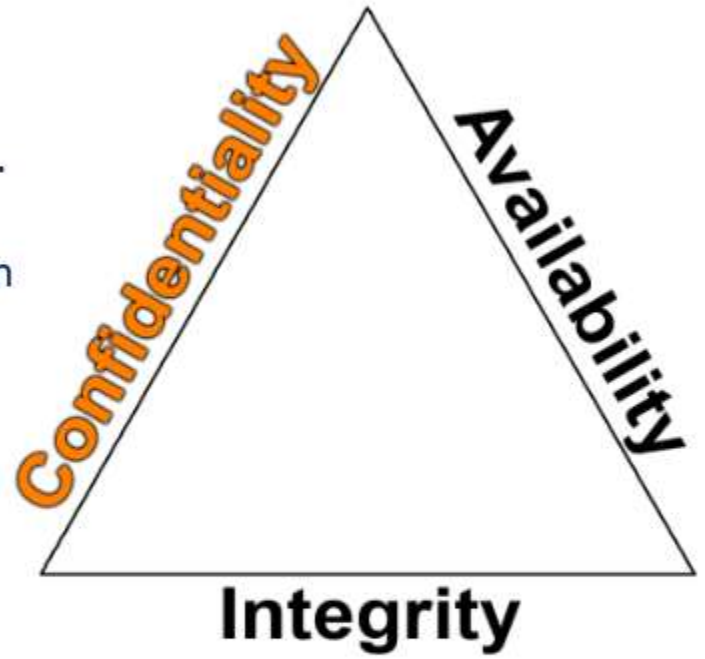
- **KGI (Key Goal Indicator):**
  - Define measures that tell management, after the fact—whether an IT process has achieved its business requirements.
- **KPI (Key Performance Indicators):**
  - Define measures that determine how well the IT process is performing in enabling the goal to be reached.
- **KRI (Key Risk Indicators):**
  - Metrics that demonstrate the risks that an organization is facing or how risky an activity is.
  - They are the mainstay of measuring adherence to and establishing enterprise risk appetite.
  - Key risk indicators are metrics used by organizations to provide an early signal of increasing risk exposures in various areas of the enterprise.
  - KRI give an early warning to identify potential event that may harm continuity of the activity/project.

Confidentiality, Integrity and Availability.

- **The CIA Triad (AIC)**
  - **Confidentiality**
    - This is what most people think IT Security is.
    - We keep our data and secrets secret.
    - We ensure no one unauthorized can access the data.
  - **Integrity**
    - How we protect against modifications of the data and the systems.
    - We ensure the data has not been altered.
  - **Availability**
    - We ensure authorized people can access the data they need, when they need to.

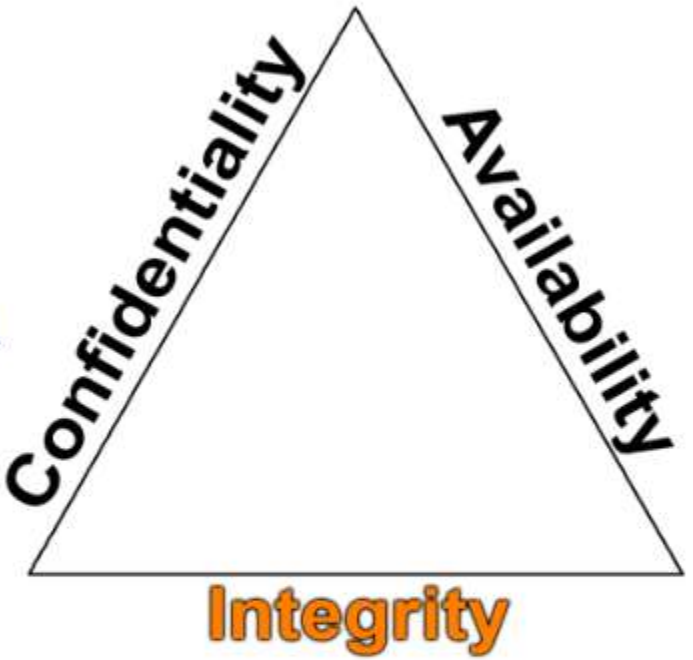
Confidentiality, Integrity and Availability.

- **We use:**
  - Encryption for **data at rest** (for instance AES256), full disk encryption.
  - Secure transport protocols for **data in motion**. (SSL, TLS or IPSEC).
  - Best practices for **data in use** - clean desk, no shoulder surfing, screen view angle protector, PC locking (automatic and when leaving).
  - Strong passwords, multi factor authentication, masking, access control, need-to-know, least privilege.
- **Threats:**
  - Attacks on your encryption (cryptanalysis).
  - Social engineering.
  - Key loggers (software/hardware), cameras, Steganography.
  - IOT (Internet Of Things) – The growing number of connected devices we have pose a new threat, they can be a backdoor to other systems.



Confidentiality, **Integrity** and Availability.

- **System integrity and Data integrity**
  - **We use:**
    - Cryptography (again).
    - Check sums (This could be CRC).
    - Message Digests also known as a hash (This could be MD5, SHA1 or SHA2).
    - Digital Signatures – non-repudiation.
    - Access control.
  - **Threats:**
    - Alterations of our data.
    - Code injections.
    - Attacks on your encryption (cryptanalysis).



Confidentiality, Integrity and **Availability.**

- **System integrity and Data availability.**
  - **We use:**
    - IPS/IDS.
    - Patch Management.
    - Redundancy on hardware power (Multiple power supplies/UPS'/generators), Disks (RAID), Traffic paths (Network design), HVAC, staff, HA (high availability) and much more.
    - SLA's – How high uptime to we want (99.9%?) – (ROI)
  - **Threats:**
    - Malicious attacks (DDOS, physical, system compromise, staff).
    - Application failures (errors in the code).
    - Component failure (Hardware).

Confidentiality, Integrity and Availability

- Finding **the right mix** of Confidentiality, Integrity and Availability is a balancing act.
- This is really the cornerstone of IT Security – finding the RIGHT mix for your organization.
  - Too much Confidentiality and the Availability can suffer.
  - Too much Integrity and the Availability can suffer.
  - Too much Availability and both the Confidentiality and Integrity can suffer.

- The opposites of the CIA Triad is DAD (Disclosure, Alteration and Destruction).
  - Disclosure – Someone not authorized getting access to your information.
  - Alteration – Your data has been changed.
  - Destruction – Your data or systems have been destroyed or rendered inaccessible.

Sensitive Information and Media Security:

- **Sensitive information**
- Any organization has data that is considered sensitive for a variety of reasons.
- We want to protect the data from Disclosure, Alteration and Destruction (**DAD**).

  - **Data has 3 States:** We want to protect it as well as we can in each state.
    - **Data at Rest** (Stored data): This is data on disks, tapes, CDs/DVDs, USB sticks
      - We use disk encryption (full/partial), USB encryption, tape encryption (avoid CDs/DVDs).
      - Encryption can be hardware or software encryption.
    - **Data in Motion** (Data being transferred on a network).
      - We encrypt our network traffic, end to end encryption, this is both on internal and external networks.
    - **Data in Use:** (We are actively using the files/data, it can't be encrypted).
      - Use good practices: Clean desk policy, print policy, allow no 'shoulder surfing', may be the use of view angle privacy screen for monitors, locking computer screen when leaving workstation.

Sensitive information and Media Security:

- **Sensitive Information**
  - **Data handling:**
    - Only trusted individuals should handle our data; we should also have policies on how, where, when, why the data was handled. Logs should be in place to show these metrics.
  - **Data storage:**
    - Where do we keep our sensitive data? It should be kept in a secure, climate-controlled facility, preferably geographically distant or at least far enough away that potential incidents will not affect that facility too.
      - Many older breaches were from bad policies around tape backups.
      - Tapes were kept at the homes of employees instead of at a proper storage facility or in a storage room with no access logs and no access restrictions (often unencrypted).

Sensitive information and Media Security:

- **Sensitive information**
  - **Data retention:**
    - Data should not be kept beyond the period of usefulness or beyond the legal requirements (whichever is greater).
    - Regulation (HIPAA or PCI-DSS) may require a certain retention of the data (1, 3, 7 years or infinity).
    - Each industry has its own regulations and company policies may differ from the statutory requirements.
    - Know your retention requirements!

Data Classification Policies:

- **Labels: Objects have Labels** assigned to them.
  - The label is used to allow Subjects with the right clearance to access them.
  - Labels are often more granular than just "Top Secret" they can be "Top Secret – Nuclear."
- **Clearance: Subjects have Clearance** assigned to them.
  - A formal decision on a subject's current and future trustworthiness.
  - The higher the clearance, the more in-depth the background checks should be (always in military, not always in business).

Military Classification
TOP SECRET
SECRET
CONFIDENCIAL
UNCLASSIFIED

Business Classification
HIGHLY SENSITIVE
SENSITIVE
INTERNAL
PUBLIC

Data Classification Policies:

- **Formal Access Approval:**
  - Document from the data owner approving access to the data for the subject.
  - Subject must understand all requirements for accessing the data and the liability involved if compromised, lost or destroyed.
  - Appropriate Security Clearance is required as well as the Formal Access Approval.
- **Need to know:**
  - Just because you have access does not mean you are allowed the data.
  - You need a **valid** reason for accessing the data. If you do not have one you can be terminated/sued/jailed/fined.
    - Leaked information about Octomom Natalie Suleman cost 15 Kaiser employees fines or terminations because they had no valid reason for accessing her file.
    - We may never know who actually leaked the information. It may not be one of the 15, but they violated HIPAA by accessing the data.
- **Least privilege:** Users have the minimum necessary access to perform their job duties.

Data, system, mission ownership, custodians and users:

Each role has unique roles and responsibilities to keep the data safe.

- **Mission/business owner:** Senior executives make the policies that govern our data security.
- **Data/information owner:** Management level, they assign sensitivity labels and backup frequency.
  - This could be you or a data owner from HR, payroll or other departments.
- **System owner:** Management level and the owner of the systems that house the data.
  - Often a data center manager or an infrastructure manager.
- **Data custodian:** These are the technical hands-on employees who do the backups, restores, patches, system configuration. They follow the directions of the data owner.
- **Users:** These are the users of the data. User awareness must be trained; they need to know what is acceptable and what is not acceptable, and the consequences for not following the policies, procedures and standards.
- **Data controllers and data processors:**
  - Controllers create and manage sensitive data in the organization (HR/Payroll)
  - Processors manage the data for controllers (Outsourced payroll)

Data Security Controls and Frameworks:

- We use standards, baselines, scoping and tailoring to decide which controls we use, and how we deploy them.
- Different controls are deployed for data at rest and data in motion.
- Some of the standards and frameworks used could be PCI-DSS, ISO27000, OCTAVE, COBIT or ITIL.
- **Scoping** is determining which portion of a standard we will deploy in our organization.
  - We take the portions of the standard that we want or that apply to our industry, and determine what is in scope and what is out of scope for us.
- **Tailoring** is customizing a standard to your organization.
  - This could be: we will apply this standard, but we use a stronger encryption (AES 256bit).
- **Classification**: A system, and the security measures to protect it, meet the security requirements set by the data owner or by regulations/laws.
- **Accreditation**: The data owner accepts the certification and the residual risk. This is required before the system can be put into production.

Ethics:

- **ISACA professional Code of Ethics**: You sign this before the exam.
  1. Support the implementation of, and encourage compliance with, appropriate standards and procedures for the effective governance and management of enterprise information systems and technology, including: audit, control, security and risk management.
  2. Perform their duties with objectivity, due diligence and professional care, in accordance with professional standards.
  3. Serve in the interest of stakeholders in a lawful manner, while maintaining high standards of conduct and character, and not discrediting their profession or the Association.
  4. Maintain the privacy and confidentiality of information obtained in the course of their activities unless disclosure is required by legal authority. Such information shall not be used for personal benefit or released to inappropriate parties.
  5. Maintain competency in their respective fields and agree to undertake only those activities they can reasonably expect to complete with the necessary skills, knowledge and competence.
  6. Inform appropriate parties of the results of work performed including the disclosure of all significant facts known to them that, if not disclosed, may distort the reporting of the results.
  7. Support the professional education of stakeholders in enhancing their understanding of the governance and management of enterprise information systems and technology, including: audit, control, security and risk management.

Failure to comply with this Code of Professional Ethics can result in an investigation into a member's or certification holder's conduct and, ultimately, in disciplinary measures.

Ethics:

- **Computer Ethics Institute**
  - **Ten Commandments of Computer Ethics:**
    - Thou shalt not use a computer to harm other people.
    - Thou shalt not interfere with other people's computer work.
    - Thou shalt not snoop around in other people's computer files.
    - Thou shalt not use a computer to steal.
    - Thou shalt not use a computer to bear false witness.
    - Thou shalt not copy or use proprietary software for which you have not paid.
    - Thou shalt not use other people's' computer resources without authorization or proper compensation.
    - Thou shalt not appropriate other people's' intellectual output.
    - Thou shalt think about the social consequences of the program you are writing or the system you are designing.
    - Thou shalt always use a computer in ways that ensure consideration and respect for your fellow humans.

Ethics:

- **IAB's Ethics and the Internet**
  - Defined as a Request For Comment (RFC), #1087 - Published in 1987
  - Considered unethical behavior:
    - Seeks to gain unauthorized access to the resources of the Internet.
    - Disrupts the intended use of the Internet.
    - Wastes resources (people, capacity, computer) through such actions :
      - Destroys the integrity of computer-based information.
      - Compromises the privacy of users.

- **Your Organization's Ethics:**
  - You need to know the Internal Code of Ethics of your organization
    - If you don't, how can you adhere to it?

Legal and regulatory issues.

As IT Security Professionals we need to understand that laws and regulations have a huge influence on how we work.
We need to know some of them and understand how the rest work.

- There are 4 types of laws covered on the exam and important to your job as an IT Security Professional.
  - **Criminal Law:**
    - "Society" is the victim and proof must be "Beyond a reasonable doubt".
    - Incarceration, death and financial fines to "Punish and deter".
  - **Civil Law** (Tort Law):
    - Individuals, groups or organizations are the victims and proof must be "the majority of proof".
    - Financial fines to "Compensate the victim(s)".
  - **Administrative Law** (Regulatory Law):
    - Laws enacted by government agencies (FDA Laws, HIPAA, FAA Laws etc.) Proof "More likely than not".
  - **Private Regulations**:
    - Compliance is required by contract (For instance PCI-DSS).

Legal and regulatory issues.

- **Liability:**
  - If the question is who is ULTIMATELY liable, the answer is Senior Leadership. This does not mean you are not liable; you may be, that depends on Due Care. Who is held accountable, who is to blame, who should pay?
- **Due Diligence and Due Care:**
  - Due Diligence – The research to build the IT Security architecture of your organization. Best practices and common protection mechanisms. Research of new systems before implementing.
  - Due Care – Prudent person rule – What would a prudent person do in this situation?
    - Implementing the IT Security architecture, keep systems patched. If compromised: fix the issue, notify affected users (Follow the Security Policies to the letter).
- **Negligence** (and gross negligence) is the opposite of Due Care.
  - If a system under your control is compromised and you can prove you did your Due Care, you are most likely not liable.
  - If a system under your control is compromised and you did NOT perform Due Care, you are most likely liable.

Legal and regulatory issues.

- **Evidence:**
  - How you obtain and handle evidence is VERY important.
  - **Types of evidence:**
    - **Real Evidence:** Tangible and physical objects in IT Security: Hard disks, USB drives – NOT the data on them.
    - **Direct Evidence:** Testimony from a first hand witness, what they experienced with their 5 senses.
    - **Circumstantial Evidence:** Evidence to support circumstances for a point or other evidence.
    - **Collaborative Evidence:** Supports facts or elements of the case: not a fact on its own, but support other facts.
    - **Hearsay:** Not first-hand knowledge – normally inadmissible in a case.
      - Computer-generated records and with that log files were considered hearsay, but case law and updates to the Federal Rule of Evidence have changed that. Rule 803 provides for the admissibility of a record or report that was "made at or near the time by, or from information transmitted by, a person with knowledge, if kept in the course of a regularly conducted business activity, and if it was the regular practice of that business activity to make the memorandum, report, record or data compilation."

Legal and regulatory issues.

- **Evidence:**
  - **Best Evidence Rule** – The courts prefer the best evidence possible.
    - Evidence should be accurate, complete, relevant, authentic, and convincing.
  - **Secondary Evidence** – This is common in cases involving IT.
    - Logs and documents from the systems are considered secondary evidence.
  - **Evidence Integrity** – It is vital that the evidence's integrity cannot be questioned.
    - We do this with hashes. Any forensics is done on copies and never the originals.
    - We check hash on both original and copy before and after the forensics.
  - **Chain of Custody** – This is done to prove the integrity of the data; that no tampering was done.
    - Who handled it?
    - When did they handle it?
    - What did they do with it?
    - Where did they handle it?

Legal and regulatory issues.

- **Reasonable Searches:**
  - The Fourth Amendment to the United States Constitution protects citizens from unreasonable search and seizure by the government.
  - In all cases, the court will determine if evidence was obtained legally. If not, it is inadmissible in court.
  - Exigent circumstances apply if there is an immediate threat to human life or of evidence destruction.
    - This will later be decided by a court if it was justified.
    - Only applies to law enforcement and those operating under the "color of law" – Title 18. U.S.C. Section 242 – Deprivation of Rights Under the Color of Law.
  - Your organization needs to be very careful when ensuring that employees are made aware in advance that their actions are monitored, that their equipment, and maybe even personal belongings, can be subjected to searches.
    - Notifications like that should only be made if your organization has security policies in place for it, and it must take into account the privacy laws in your county/state/country.

Legal and regulatory issues.

- **Entrapment and Enticement:**
  - **Entrapment** (Illegal and unethical): When someone is persuaded to commit a crime they had no intention of committing and is then charged with it.
    - Openly advertising sensitive data and then charging people when they access them.
    - Entrapment is a solid legal defense.
  - **Enticement** (Legal and ethical): Making committing a crime more enticing, but the person has already broken the law or at least has decided to do so. Honeypots can be a good way to use Enticement.
    - Have open ports or services on a server that can be attacked.
    - Enticement is not a valid defense.

  - If there is a gray area in some cases between Entrapment and Enticement, it is ultimately up to the jury to decide which one it was.
  - Check with your legal department before using honeypots. They pose both legal and practical risks.

- **GDPR** (General Data Protection Regulation):
  - GDPR is a regulation in EU law on data protection and privacy for all individuals within the European Union (EU) and the European Economic Area (EEA).
  - It does **not** matter where we are based, if we have customers in EU/EEA we have to adhere to the GDPR.
  - Violators of the GDPR may be fined up to €20 million or up to 4% of the annual worldwide turnover of the preceding financial year in case of an enterprise, whichever is greater.
  - Unless a data subject has provided informed consent to data processing for one or more purposes, personal data may not be processed unless there is at least one legal basis to do so.
  - **Restrictions:** Lawful interception, national security, military, police, justice.
  - **Personal data** covers a variety of data types including: Names, Email Addresses, Addresses, Unsubscribe confirmation URLs that contain email and/or names, IP Addresses

- **GDPR** (General Data Protection Regulation):
  - **Restrictions:** Lawful interception, national security, military, police, justice.
  - **Right to access:** Data controllers must be able to provide a free copy of an individual's data if requested.
  - **Right to erasure:** All users have a 'right to be forgotten'.
  - **Data portability:** All users will be able to request access to their data 'in an electronic format'.
  - **Data breach notification:** Users and data controllers must be notified of data breaches within 72 hours.
  - **Privacy by design:** When designing data processes, care must be taken to ensure personal data is secure. Companies must ensure that only data is 'absolutely necessary for the completion of duties'.
  - **Data protection officers:** Companies whose activities involve data processing and monitoring must appoint a data protection officer.

Legal and regulatory issues.

Intellectual Property:

- **Copyright ©** - (Exceptions: first sale, fair use).
  - Books, art, music, software.
  - Automatically granted and lasts **70 years after creator's death or 95 years after creation by/for corporations.**
- **Trademarks ™** and ® (Registered Trademark).
  - Brand names, logos, slogans – Must be registered, is valid for 10 years at a time, can be renewed indefinitely.
- **Patents: Protects inventions for 20 years** (normally) – **Cryptography algorithms can be patented.**
    - Inventions must be:
      - **Novel** (New idea no one has had before).
      - **Useful** (It is actually possible to use and it is useful to someone).
      - **Nonobvious** (Inventive work involved).
- **Trade Secrets.**
  - You tell no one about your formula, your secret sauce. If discovered anyone can use it; you are not protected.

Legal and regulatory issues.

Attacks on Intellectual Property:

- **Copyright.**
  - Piracy - Software piracy is by far the most common attack on Intellectual Property.
  - Copyright infringement – Use of someone else's copyrighted material, often songs and images.
- **Trademarks.**
  - Counterfeiting – Fake Rolexes, Prada, Nike, Apple products – Either using the real name or a very similar name.
- **Patents.**
  - Patent infringement – Using someone else's patent in your product without permission.
- **Trade Secrets.**
  - While a organization can do nothing if their Trade Secret is discovered, *how* it is done can be illegal.

- **Cyber Squatting** – Buying an URL you know someone else will need (To sell at huge profit – not illegal).
- **Typo Squatting** – Buying an URL that is VERY close to real website name (Can be illegal in certain circumstances).

- **BCP and DRP:**
  - **Warfare, Terrorism and Sabotage (Human):**
    - We still see plenty of conventional conflicts and wars, but there is much more happening behind the veil of the internet, hacking for causes, countries, religion and many more reasons.
    - It makes sense to cripple a country's or region's infrastructure if you want to invade or just destabilize that area.
    - This could be for war, trade, influence or many other reasons, everything is so interconnected we can shut down water, electricity or power from across the world.
    - The targets are not always the obvious targets, hospitals, air travel, shipping, production, ... are potential targets.
    - **State, cause or religious hacking (Human):**
      - Common, we often see the attacks happening 9-5 in that time zone, this is a day job.
      - Approximate 120 countries have been developing ways to use the internet as a weapon to target financial markets, government computer systems and utilities.
      - Famous attacks: US elections (Russia), Sony websites (N. Korea), Stuxnet (US/Israel), US Office of Personnel Management (China) ...

- **BCP and DRP:**
  - **Financially Motivated Attackers (Human):**
    - We are seeing more and more financially motivated attacks, they can be both highly skilled or not.
    - The lower skilled ones could be normal phishing attacks, social engineering or vishing, these are often a numbers game, but only a very small percentage needs to pay to make it worth the attack.
    - The ones requiring more skills could be stealing cardholder data, identity theft, fake anti-malware tools, or corporate espionage, ...
    - Ransomware is a subtype of financially motivated attacks, it will encrypt a system until a ransom is paid, if not paid the system is unusable, if paid the attacker may send instructions on how to recover the system.
    - Attackers just want the payday, they don't really care from whom.

WannaCry ransomware screenshot from an infected system.

- **Administrative Personnel Security Controls:**
  - **Administrative Security:** 
    - Provides the means to control people's operational access to data.
    - **Least Privilege:**
      - We give employees the minimum necessary 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. (Kaiser employees).
    - **Separation of duties:**
      - More than one individual in one single task is an internal control intended to prevent fraud and error.
      - We do not allow the same person to enter the purchase order and issue the check.
      - For the exam assume the organization is large enough to use separation of duties, in smaller organizations where that is not practical, compensating controls should be in place.

- **Administrative Personnel Security Controls:**
  - **Administrative Security:** 
    - **Job rotation:**
      - For the exam think of it to detect errors and frauds. It is easier to detect fraud and there is less chance of collusion between individuals if they rotate jobs.
      - It also helps with employees burnout and it helps employees understand the entire business.
      - This can be to cost prohibitive for the exam/real life, make sure on the exam the cost justifies the benefit.
    - **Mandatory vacations:**
      - Done to ensure one person is not always performing the same task, someone else has to cover and it can keep fraud from happening or help us detect it.
      - Their accounts are locked and an audit is performed on the accounts.
      - If the employee has been conducting fraud and covering it up, the audit will discover it.
      - The best way to do this is to not give too much advance notice of vacations.
  - With the combination of all 5 we minimize some of the insider threats we may have.

- **Administrative Personnel Security Controls:**
  - **NDA (non-disclosure agreement):**
    - We covered NDA's between our and other organizations, it is also normal to have them for internal employees.
    - Some employment agreements will include a clause restricting employees' use and dissemination of company-owned confidential information.
  - **Background checks:**
    - References, Degrees, Employment, Criminal, Credit history (less common, more costly).
    - For sensitive positions the background check is an ongoing process.
  - **Privilege monitoring:**
    - The more access and privilege an employee has the more we keep an eye on their activity.
    - They are already screened more in depth and consistently, but they also have access to many business critical systems, we need to audit their use of that access.
    - With more access comes more responsibility and scrutiny.

- **Designing security into our software:**
  - The more breaches and compromises, the more we see the move towards security being part of the scope of the software design project.
  - We use software at our jobs, our personal lives, our homes, cars, power, water...
  - It is everywhere. And it has been, and still is, common to write functional code. Security is an afterthought, or not considered at all.
  - A large part of our defense-in-depth is to protect our assets, but ultimately most of it is to protect our data/software.
  - Software with security built in is much securer than software where it is added on later.
  - It is common for programmers to make 15-50 mistakes per 1,000 lines of code. If using a programming maturity framework, we can lower that to 1 error per 1,000 lines of code.
  - Most of the errors are not a vulnerability, or really a concern, but the more we use software in everything, the more critical the vulnerabilities become.
  - Hacks have accelerated and stopped cars on highways, had planes change course (hacked through bad security on the in-flight entertainment), power grids, elections...

- **Programming concepts:**
  - **Machine Code:**
    - Software executed directly by the CPU, 0's and 1's understood by the CPU.
  - **Source Code:**
    - Computer programming language, written in text and is human understandable, translated into machine code.
  - **Assembler Language:**
    - Short mnemonics like ADD/SUB/JMP, which are matched with the full length binary machine code; assemblers convert assembly language into machine language. A disassembler does the reverse.
  - **Compiler Languages:**
    - Translates the higher level language into machine code and saves, often as executables, compiled once and run multiple times.
  - **Interpreted languages:**
    - Similar to compiler languages, but interprets the code each time it is run into machine code.

- **Programming concepts:**
  - **Bytecode:**
    - An interpreted code, in intermediary form, converted from source code to interpreted, but still needs to be converted into machine code before it can run on the CPU.
  - **Procedural languages** (Procedure-oriented):
    - Uses subroutines, procedures and functions.
  - **Object-oriented Programming (OOP):**
    - Based on the concept of objects, which may contain data, in the form of fields, often known as attributes, and code, in the form of procedures, often known as methods.
    - An object's procedures can access and often modify the data fields of the objects with which they are associated.
    - In OOP, computer programs are designed by making them out of objects that interact with one another.

- **Programming concepts:**
  - **4th Generation languages (4GL):**
    - Fourth-generation languages are designed to reduce programming effort and the time it takes to develop software, resulting in a reduction in the cost of software development.
    - Increases the efficiency by automating the creation of machine code.
    - Often uses a GUI, drag and drop, and then generating the code, often used for websites, databases and reports.
  - **Programming languages and generations:**
    - **1st generation:** Machine Code
    - **2nd Generation:** Assembler
    - **3rd Generation:** Cobol, basic, C, C++, C#, Java, JavaScript, …
    - **4th Generation:** ColdFusion, Progress 4GL, SQL, PHP, Perl, …

- **Programming concepts:**
  - **CASE (Computer-Aided Software Engineering):**
    - Similar to and were partly inspired by computer-aided design (CAD) tools used for designing hardware products.
    - Used for developing high-quality, defect-free, and maintainable software.
    - Often associated with methods for the development of information systems together with automated tools that can be used in the software development process.
    - **CASE software** is classified into 3 categories:
      - **Tools** support specific tasks in the software life-cycle.
      - **Workbenches** combine two or more tools focused on a specific part of the software life-cycle.
      - **Environments** combine two or more tools or workbenches and support the complete software life-cycle.

- **Programming concepts:**
  - **Top-Down Programming:**
    - Starts with the big picture, then breaks it down into smaller segments.
    - An overview of the system is formulated, specifying, but not detailing, any first-level subsystems.
    - Each subsystem is then refined in yet greater detail, sometimes in many additional subsystem levels, until the entire specification is reduced to base elements.
    - Procedural programming leans toward Top-Down, you start with one function and add to it.
  - **Bottom-Up Programming:**
    - Piecing together of systems to build more complex systems, making the original systems a sub-system of the overarching system.
    - The individual base elements of the system are first specified in great detail, they are then linked together to form larger subsystems, which then in turn are linked, sometimes in many levels, until a complete top-level system is formed.
    - OOP leans tends toward Bottom-Up, you start by developing your objects and build up.

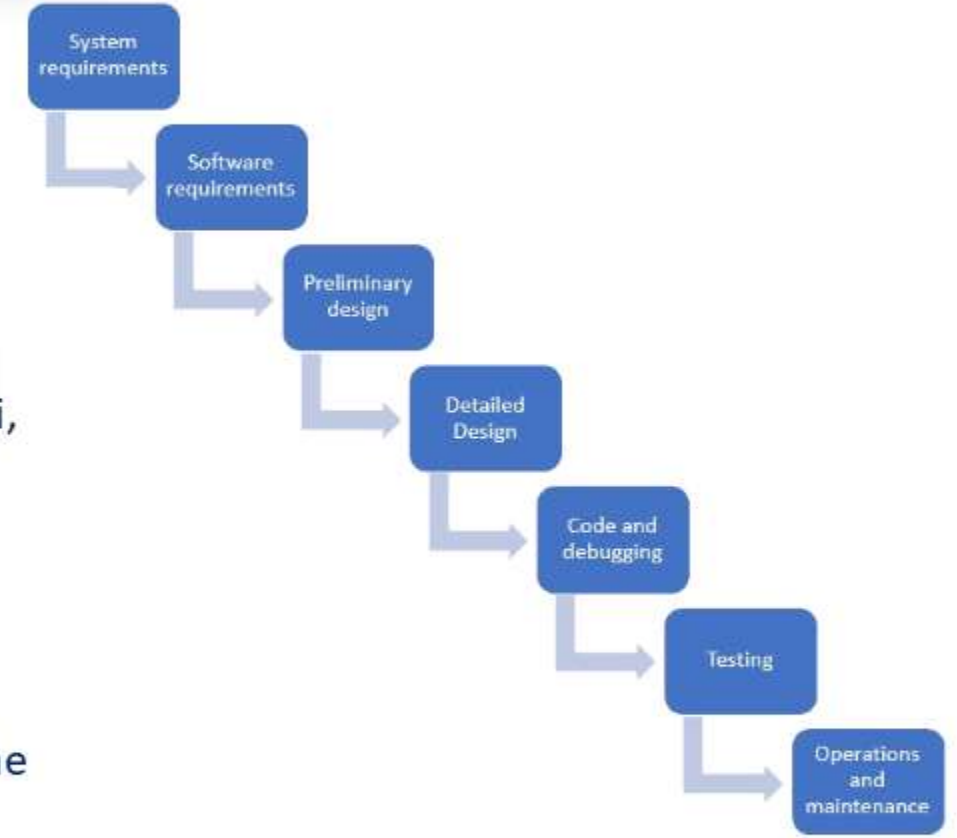
- **Programming concepts:**
  - **Software release:**

    Open source:
    - We release the code publicly, where it can be tested, improved and corrected, but it also allows attackers to find the flaws in the code.
  - **Closed Source:**
    - We release the software, but keep the source code a secret, may be sound business practice, but can also be security through obscurity.
  - **Proprietary software:**
    - Software protected by intellectual property and/or patents, often used interchangeably with Closed Source software, but it really is not. It can be both Open and Closed Source software.
    - Any software not released into the public domain is protected by copyright.

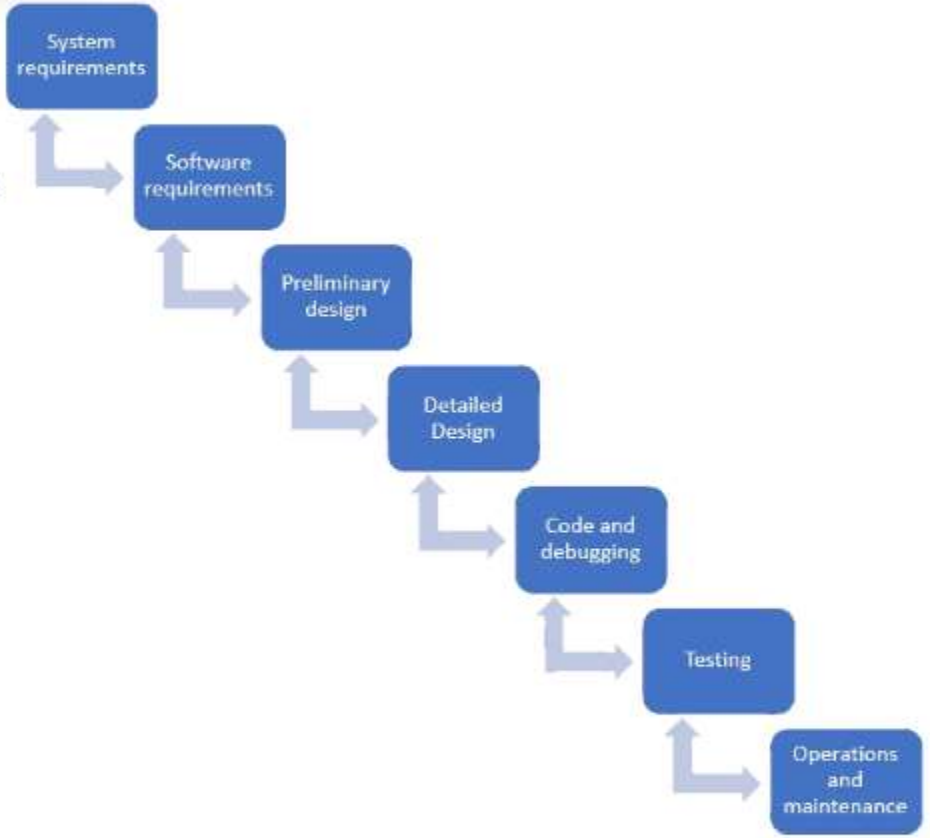
- **Programming concepts:**
  - **Software release:**
    - **Free software:**
      - **Freeware:**
        - Actually free software, it is free of charge to use.
      - **Shareware:**
        - Fully functional proprietary software that is initially free to use.
        - Often for trials to test the software, after 30 days you have to pay to continue to use.
      - **Crippleware:**
        - Partially functioning proprietary software, often with key features disabled.
        - The user is required to make a payment to unlock the full functionality.
    - **EULAs (End-User License Agreements):**
      - Electronic form where the user clicks "I agree" to the software terms and conditions while installing the software.

- **Programming concepts:**
  - **Software licenses:**
    - Open source software can be protected by a variety of licensing agreement.
      - **GNU** (General Public License): Also called GPL or GPL
        - Guarantees end users the freedom to run, study, share and modify the software.
        - A copyleft license, which means that derivative work can only be distributed under the same license terms.
      - **BSD** (Berkeley Software Distribution):
        - A family of permissive free software licenses, imposing minimal restrictions on the use and redistribution of covered software.
        - This is different than copyleft licenses, which have reciprocity share-alike requirements.
    - **Apache:**
      - Software must be free, distribute, modify and distribute the modified software.
      - Requires preservation of the copyright notice and disclaimer.

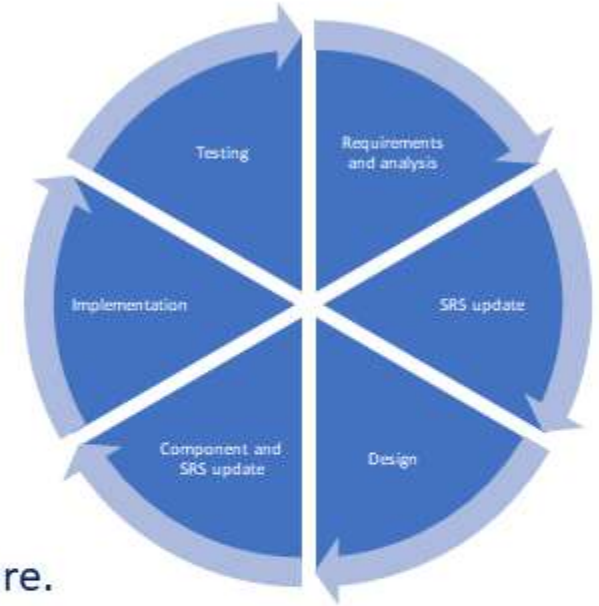
- **Software Development Methodologies:**
  - There is a wide range of software development methodologies used today.
  - In the past the Waterfall method was widely used, it is a very linear process, and does not work very well with the iterative nature of software development.
  - To remedy that problem other methods were developed Spiral, Sashimi, Agile and Scrum.
  - The individual phases are different from organization to organization, understand how each methodology works and the phases flows.
  - **Waterfall:**
    - Very linear, each phase leads directly into the next.
    - The unmodified waterfall model does not allow us to go back to the previous phase.



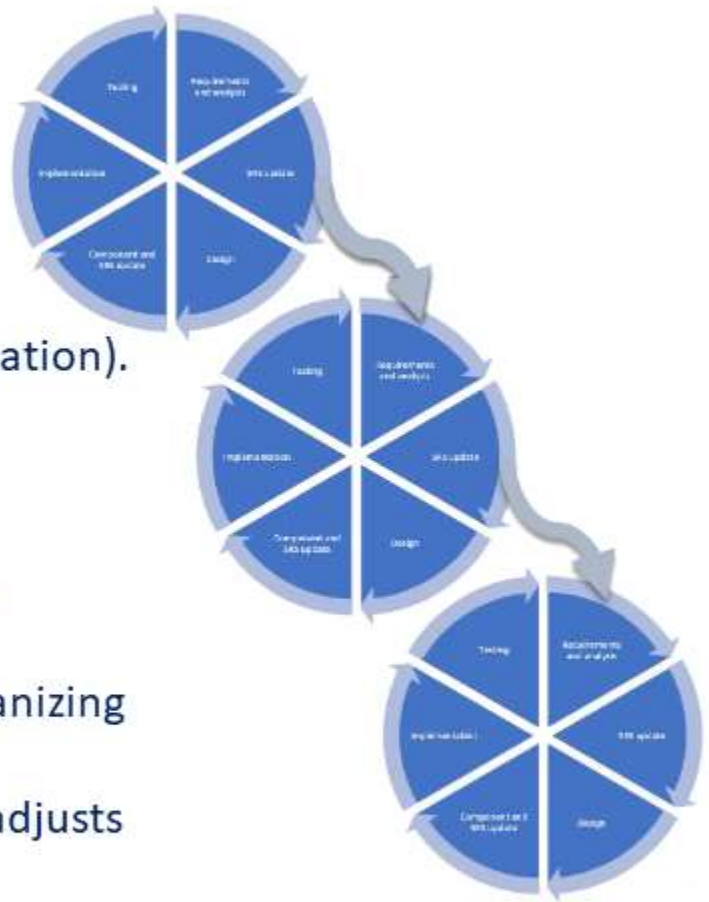
- **Software Development Methodologies:**
  - **Sashimi model** (Waterfall with overlapping phases):
    - Similar to waterfall, but we always have 2 overlapping phases, if we close one phase, we add the next phase.
    - The modified waterfall model allows us to go back to the previous phase but no further.
  - **Agile software development:**
    - Describes a set of values and principles for software development under which requirements and solutions evolve through the collaborative effort of self-organizing cross-functional teams.
    - Uses adaptive planning, evolutionary development, early delivery, and continuous improvement, and it encourages rapid and flexible response to change.
    - There are many types of agile, for the exam know the flow.



- **Software Development Methodologies:**
  - **Agile software development:**
    - **Manifesto for Agile Software Development:**
      - **What is valued in the manifesto:**
        - Individuals and Interactions more than processes and tools.
        - Working Software more than comprehensive documentation.
        - Customer Collaboration more than contract negotiation.
        - Responding to Change more than following a plan.
      - **The twelve principles in the manifesto:**
        - Customer satisfaction by early and continuous delivery of valuable software.
        - Welcome changing requirements, even in late development.
        - Working software is delivered frequently (weeks rather than months).
        - Close, daily cooperation between business people and developers.
        - Projects are built around motivated individuals, who should be trusted.



- **Software Development Methodologies:**
  - **Agile software development:**
    - **Manifesto for Agile Software Development:**
      - **The twelve principles in the manifesto:**
        - Face-to-face conversation is the best form of communication (co-location).
        - Working software is the primary measure of progress.
        - Sustainable development, able to maintain a constant pace.
        - Continuous attention to technical excellence and good design.
        - Simplicity—the art of maximizing the amount of work not done—is essential.
        - Best architectures, requirements, and designs emerge from self-organizing teams.
        - Regularly, the team reflects on how to become more effective, and adjusts accordingly.

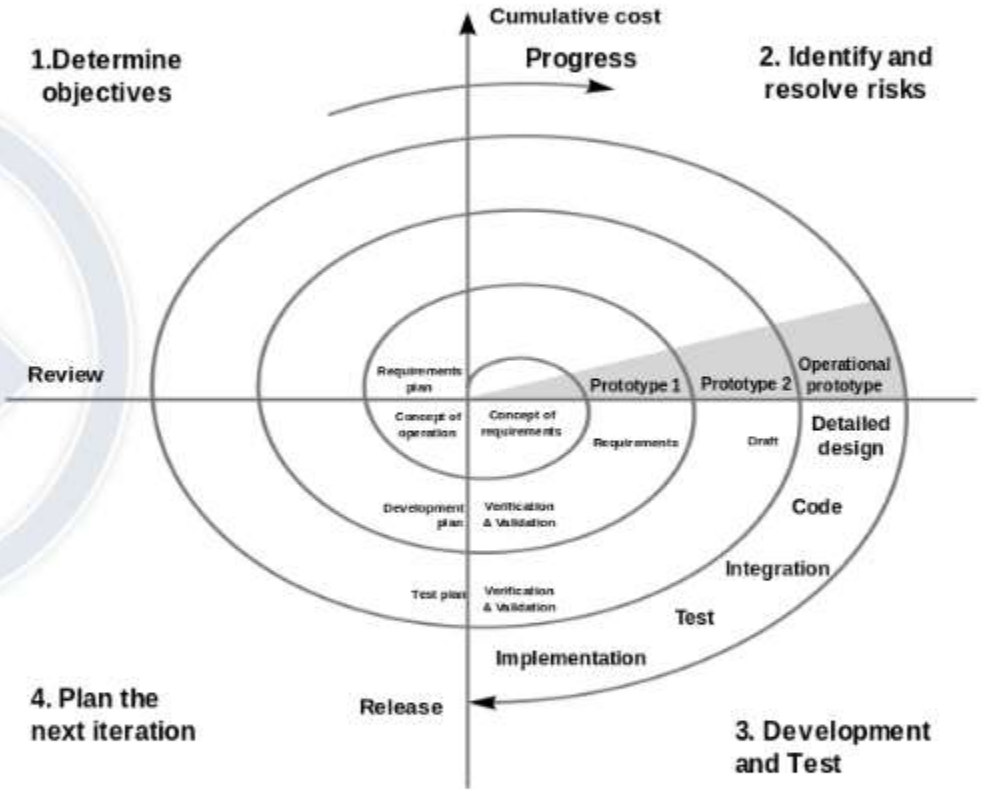


- **Software Development Methodologies:**
  - **Agile software development:**
    - **Scrum:**
      - Scrum is a framework for managing software development. Scrum is designed for teams of approximately 10 individuals, and generally relies on two-week development cycles, called "sprints", as well as short daily stand-up meetings.
      - The three core roles in the Scrum framework.
        - **The product owner:**
          - Representing the product's stakeholders, the voice of the customer, and is accountable for ensuring that the team delivers value to the business.
        - **Development team:**
          - Responsible for delivering the product at the end of each sprint (sprint goal).
          - The team is made up of 3–9 individuals who do the actual work (analysis, design, develop, test, technical communication, document, etc.).

- **Software Development Methodologies:**
  - **Agile software development:**
    - **Scrum:**
      - The three core roles in the Scrum framework.
        - **Development team:**
          - Development teams are cross-functional, with all of the skills as a team necessary to create a product increment.
        - **Scrum master:**
          - Facilitates and accountable for removing impediments to the ability of the team to deliver the product goals and deliverables.
          - Not a traditional team lead or project manager but acts as a buffer between the team and any distracting influences.
          - The scrum master ensures that the Scrum framework is followed.

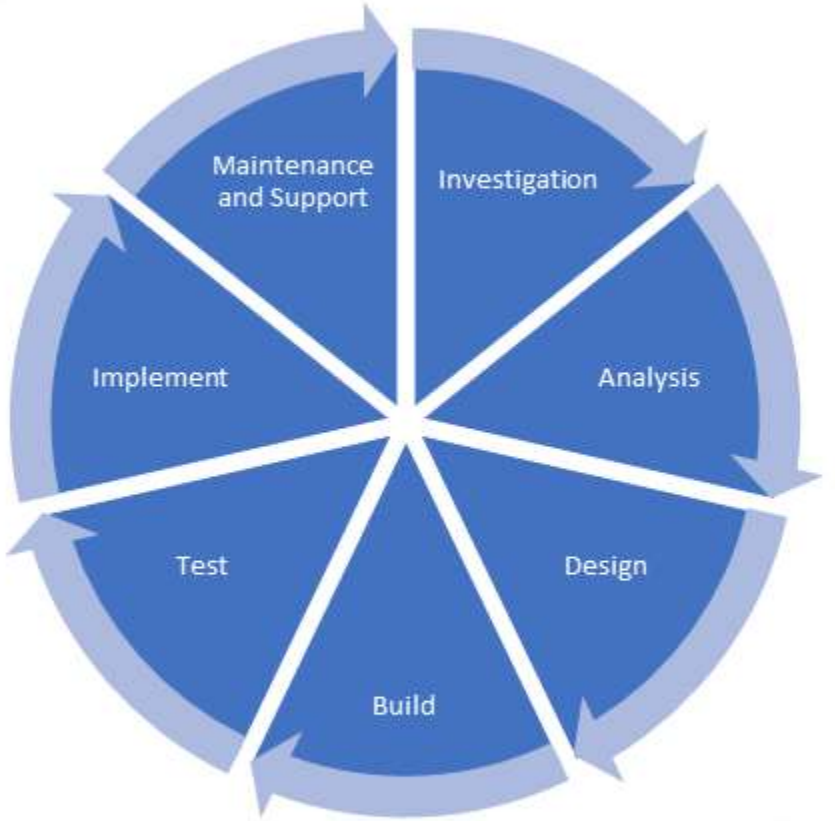
- **Software Development Methodologies:**
  - **Agile software development:**
    - **XP (Extreme programming):**
      - Intended to improve software quality and responsiveness to changing customer requirements.
      - Uses advocates frequent releases in short development cycles, intended to improve productivity and introduce checkpoints at which new customer requirements can be adopted.
      - **XP uses:**
        - Programming in pairs or doing extensive code review.
        - Unit testing of all code.
        - Avoiding programming of features until they are actually needed.
        - Flat management structure.
        - Code simplicity and clarity.
        - Expecting changes in the customer's requirements as time passes and the problem is better understood.
        - Frequent communication with the customer and among programmers.

- **Software Development Methodologies:**
  - **The spiral model:**
    - A risk-driven process model generator for software projects.
    - The spiral model has four phases: Planning, Risk Analysis, Engineering and Evaluation.
    - A software project repeatedly passes through these phases in iterations (called Spirals in this model).
    - The baseline spiral, starting in the planning phase, requirements are gathered and risk is assessed.
    - Each subsequent spiral builds on the baseline spiral.



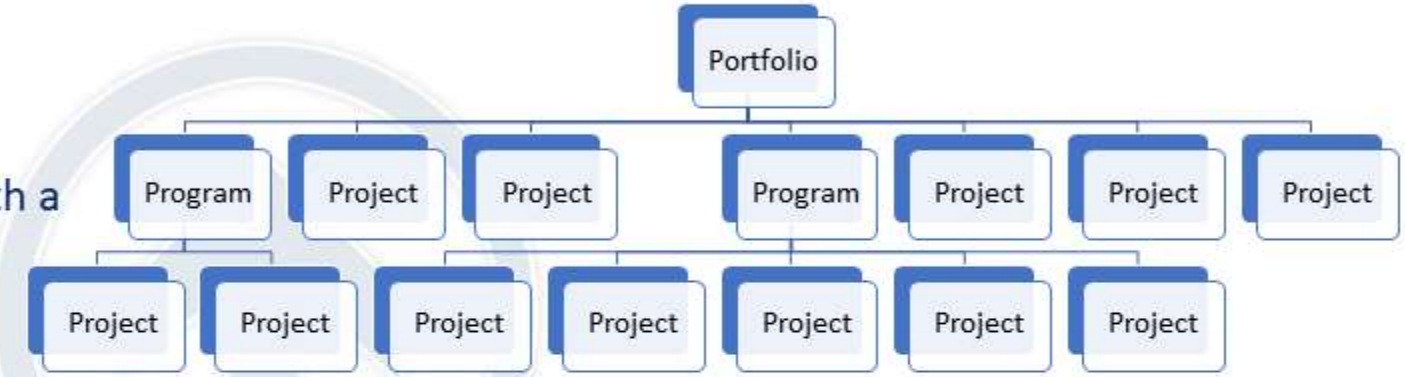
- **Software Development Methodologies:**
  - **RAD (Rapid Application Development):**
    - Puts an emphasis on adaptability and the necessity of adjusting requirements in response to knowledge gained as the project progresses.
    - Prototypes are often used in addition to or sometimes even in place of design specifications.
    - Very suited for developing software that is driven by user interface requirements.
    - GUI builders are often called rapid application development tools.
  - **Prototyping:**
    - Breaks projects into smaller tasks, creating multiple prototypes of system design features.
    - A working model of software with some limited functionality, rather than designing the full software up front.
    - Has a high level of customer involvement, the customer inspects the prototypes to ensure that the project is on track and meeting its objective.

- **Software Development Methodologies:**
  - **SDLC** (Software Development Life Cycle):
    - The SDLC is not really a methodology, but a description of the phases in the life cycle of software development.
    - These phases are (in general), investigation, analysis, design, build, test, implement, maintenance and support (and disposal).
    - Can have security built into each step of the process, for the exam it always does.
    - If an answer about SDLC does not list secure or security, it would be wrong and can be eliminated.
    - Has a number of clearly defined and distinct work phases which are used by systems engineers and systems developers to plan for, design, build, test, and deliver information systems.



- **Software Development Methodologies:**
  - **SDLC:**
    - The aim is to produce high-quality systems that meet or exceed customer expectations, based on customer requirements, by delivering systems which move through each clearly defined phase, within scheduled time frames and cost estimates.
    - SDLC is used during the development of a project, it describes the different stages involved in the project from the drawing board, through the completion of the project.
    - All software development methodologies follow the SDLC phases but the method of doing that varies vastly between methodologies.
    - Many different SDLC methodologies have been created, Waterfall, Spiral, Agile, Rapid Prototyping, ...
    - In Scrum project a single user story goes through all the phases of the SDLC within a single two-week sprint, where Waterfall projects can take many months or several years to get through the phases.
    - While very different they both contain the SDLC phases in which a requirement is defined, then pass through the life cycle phases ending in the final phase of maintenance and support.

- **Software Development Methodologies:**
  - **Projects, programs and portfolios.**
    - A **project** is a temporary endeavor, with a finite start and end, that is focused on creating a unique product, service, or result.
    - A **program** is a collection of related projects.
      Like a project, a program is temporary, when the collection of projects are complete, the program is complete.



    - A **portfolio** is a collection of projects and programs that are managed as a group to achieve strategic objectives.

- **Software Development Methodologies:**
  - **IPT** (Integrated Product Team):
    - A multidisciplinary group of people who are collectively responsible for delivering a defined product or process.
    - IPTs are used in complex development programs/projects for review and decision making.
    - The emphasis of the IPT is on involvement of all stakeholders (users, customers, management, developers, contractors) in a collaborative forum.
    - IPTs can be addressed at the program level, there may also be Oversight IPTs (OIPTs), or Working-level IPTs (WIPTs).
    - IPTs are created most often as part of structured systems engineering methodologies, focusing attention on understanding the needs and desires of each stakeholder.

- **Software Development Methodologies:**
  - **Source code escrow:**
    - The deposit of the source code of software with a third party escrow agent.
    - Escrow is typically requested by a party licensing software (the licensee), to ensure maintenance of the software instead of abandonment or orphaning.
    - The software source code is released to the licensee if the licensor files for bankruptcy or otherwise fails to maintain and update the software as promised in the software license agreement.
  - **Source code repositories:**
    - Using public third party code repositories comes with some security concerns.
    - Other than the provider security, one of the most important controls is using multi-factor authentication.
    - File archive and web hosting facility where a large amount of source code, for software or for web pages, is kept, either publicly or privately.
    - They are often used by open-source software projects and other multi-developer projects to handle various versions. They help developers submit patches of code in an organized fashion.

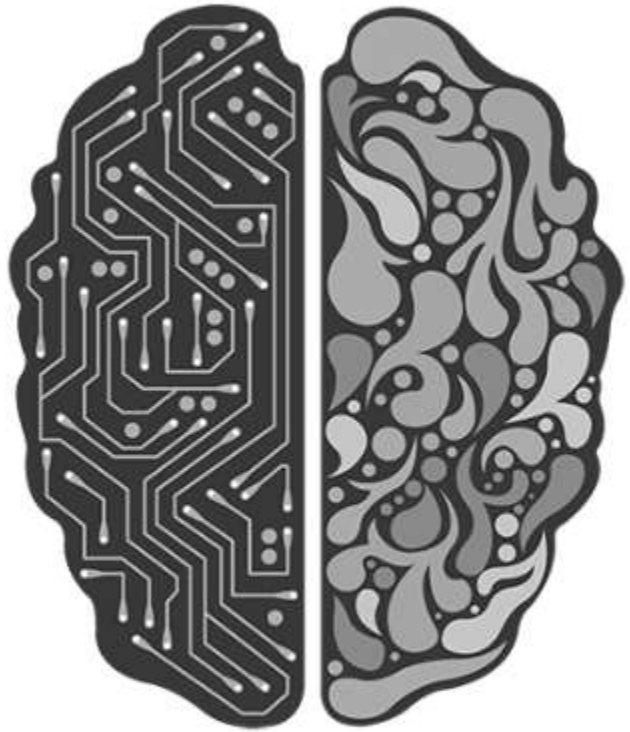
- **Software Development Methodologies:**
  - **API Security** (Application Programming Interface):
    - Allows an application to communicate with another application, operating systems, databases, networks, ...
    - Many applications use API's, this could be to add super sign-on, integrate 2 applications, or many other things, ...
    - They are a good example of how we integrate for better usability, but often security is overlooked.
    - API's are the cause of a number of recent high-profile website security breaches including SnapChat, Pinterest and Instagram.
    - We will cover the OWASP top 10 web vulnerabilities in domain 2.
    - OWASP also has an Enterprise Security API Toolkits project, which includes these critical API controls:
    - Authentication, Access control, Input validation, Output encoding/escaping, Cryptography, Error handling and logging, Communication security, HTTP security and Security configuration.

- **Software Development Methodologies:**
  - **Software Change and Configuration Management:**
    - Earlier in this domain we covered how software development has a lifecycle, and in Domain 3 we will cover configuration and change management.
    - Both change and configuration management are very applicable to our software development process, all the way from investigation/initiation to disposal of the software.
    - As with many of the concepts we cover they are to some extend logical, configuration management tracks changes to a specific piece of software where change management is all changes in the entire software development process.

- **Software Development Methodologies:**
  - **Software Change and Configuration Management:**
    - NIST 80-128: Guide for Security-Focused Configuration Management of Information Systems uses these terms:
    - A Configuration Management Plan (CM Plan) is a comprehensive description of the roles, responsibilities, policies, and procedures that apply when managing the configuration of products and systems.
    - The basic parts of a CM Plan include:
    - Configuration Control Board (CCB) – Establishment of and charter for a group of qualified people with responsibility for the process of controlling and approving changes throughout the development and operational lifecycle of products and systems, may also be referred to as a change control board.
    - Configuration Item Identification – for selecting and naming configuration items that need to be placed under CM.
    - Configuration Change Control – Process for managing updates to the baseline configurations for the configuration items.
    - Configuration Monitoring – Process for assessing or testing the level of compliance with the established baseline configuration and mechanisms for reporting on the configuration status of items placed under CM

- **Software Development Methodologies:**
  - **DevOps:**
    - A software development and delivery process that emphasizes communication and collaboration between product management, software development, and operations professionals in the entire service lifecycle, from design through
    - the development process to production support.
    - It does this by automating and monitoring the process of software integration, testing, deployment, and infrastructure changes by establishing a culture and environment where building, testing, and releasing software can happen rapidly, frequently, and more reliably.

- **AI (Artificial intelligence):**
  - Intelligence exhibited by machines, rather than humans or other animals.
  - What true AI is, is a topic of discussion, what was considered AI years ago we have achieved and when once goal is reached the AI definition is tweaked a little.
  - From what we are seeing published we do in my mind not currently have true AI, but very highly simulated intelligence, that being said IBM and Google do seem to be getting a lot closer.
  - It is also used when a machine mimics cognitive functions that humans associate with other human minds, such as learning and problem solving.
  - AI currently defined as advice that perceives its environment and takes actions that maximize its chance of success at some goal, not through experience/programming, but through reasoning.



- **AI (Artificial intelligence):**
  - **Expert systems:**
    - A computer system that emulates the decision-making ability of a human expert.
    - Designed to solve complex problems by reasoning about knowledge, represented mainly as if–then rules rather than through conventional procedural code.
    - An expert system is divided into two subsystems:
    - The knowledge base represents facts and rules.
    - The inference engine applies the rules to the known facts to deduce new facts, and can also include explanation and debugging abilities.

- **AI (Artificial intelligence):**
  - **ANN's** (Artificial neural networks):
    - Computing systems inspired by the biological neural networks that constitute animal brains, we make decisions based on 1000's of memories, stories, the situation and many other factors, the ANN tries to emulate that.
    - The systems learn and progressively improve their performance, to do tasks, generally without task-specific programming.
    - They can learn to identify images that contain geckos by analyzing example images that have been manually labeled as "gecko" or "no gecko" and using the analytic results to identify geckos in other images.
    - They are mostly used in areas that are difficult to express in a traditional computer algorithm using rule-based programming.
    - An ANN is based on a collection of connected units called artificial neurons.
    - Each connection (synapse) between neurons can transmit a signal to another neuron.
    - Typically, neurons are organized in layers, different layers may perform different transformations on their inputs.
    - Signals travel from the first input, to the last output layer, at times after traversing the layers multiple times.

- **AI (Artificial intelligence):**
  - **GP** (Genetic Programming):
    - A technique where computer programs are encoded as a set of genes that are then modified (evolved) using an evolutionary algorithm often a GA (Genetic Algorithm).
    - The results are computer programs able to perform well in a predefined task.
    - The methods used to encode a computer program in an artificial chromosome and to evaluate its fitness with respect to the predefined task are central in the GP technique and still the subject of active research.
    - GP evolves computer programs, traditionally represented in memory as tree structures.
    - Trees can be easily evaluated in a recursive manner.
    - Every tree node has an operator function and every terminal node has an operand, making mathematical expressions easy to evolve and evaluate.
    - Traditionally GP favors the use of programming languages that naturally embody tree structures for example, Lisp or other functional programming languages.

- **AI (Artificial intelligence):**
  - **GP** (Genetic Programming):
    - The process is in its simple form like this:
      - Generate an initial population of random computer programs.
      - Execute each program in the population and assign it a fitness value according to how well it solves the problem.
      - Create a new population of computer programs.
      - Copy the best existing programs
      - Create new computer programs by mutation.
      - Create new computer programs by crossover.
    - Genetic Algorithms and Genetic Programming have been used to program a Pac-Man playing program, robotic soccer teams, networked intrusion detection systems, and many others.

What we covered in Domain 1.

- This chapter is **VERY** important because:
  - Every other knowledge domain builds on top of this chapter
  - This is the **foundation**.
- 24% of the exam questions on the certification are from this domain.
- We covered our ethics, values, vision and our mission.
- We looked at the policies, the procedures, and the laws we need to adhere to.
- SWOT and GAP analysis.
- OpEx, CapEx, Fiscal years, KGIs, KPIs, and KRIs
- Secure software design.