Jeffery Moore's CISSP Domain Objectives Study Notes

Domain 1 **Security and Risk Management**

- **Glass-Steagall Act**: passed in 1933 and separated investment and commercial banking activities in response to involvement in stock market investment
    - The Gramm-Leach-Bliley Act eliminated the Glass-Steagall Act's restrictions against affiliations between commercial and investment banks in 1999
    - **Security Control Assessment (SCA)**: an evaluation process of the different type of controls such as management, operational and security control within an information system, with the purpose of validation of the requirement of a control, correct implementation, operationally being followed as intended, and result is as desired
        - basically a formal evaluation of a defined set of controls, which may be conducted with the Security Test and Evaluation (ST&E); NIST Special Publication 800-53A Security and Privacy Controls for Federal Information Systems and Organizations ensure the security requirements and enforcement of appropriate security controls
- RFC 1087: activitiy that is unethical and unacceptable:
    - (a) seeks to gain unauthorized access to the resources of the Internet
    - (b) disrupts the intended use of the Internet
    - (c) wastes resources (people, capacity, computer) through such actions
    - (d) destroys the integrity of computer-based information, and/or
    - (e) compromises the privacy of users

1.1 Understand, adhere to, and promote professional ethics (OSG-9 Chpts 1,19)

- As a CISSP, you must understand and follow the (ISC)² code of ethics, as well as your organization's own code
- 1.1.1 (ISC)² Code of Professional Ethics
    - (ISC)² Code of Professional Ethics -- take the time to read the code of ethics
    - At a minimum, know and understand the ethics canons:
        - **Protect society, the common good, necessary public trust and confidence, and the infrastructure**
            - this is "do the right thing"; put the common good ahead of yourself
            - ensure that the public can have faith in your infrastructure and security
        - **Act honorably, honestly, justly, responsibly, and legally**
            - always follow the laws
            - but if you find yourself working on a project where conflicting laws from different countries or jurisdictions apply, you should prioritize the local jurisdiction from which you are performing the services
        - **Provide diligent and competent service to principles**
            - avoid passing yourself as an expert or as qualified in areas that you aren't
            - maintain and expand your skills to provide competent services
        - **Advance and protect the profession**
            - don't bring negative publicity to the profession
            - provide competent services, get training and act honorably

        - think of it like this: If you follow the first three canons in the code of ethics, you automatically comply with this one
- 1.1.2 Organizational code of ethics
    - You must also support ethics at your organization; this can be interpreted to mean evangelizing ethics throughout the organization, providing documentation and training around ethics, or looking for ways to enhance the existing organizational ethics
        - some organizations might have slightly different ethics than others, so be sure to familiarize yourself with your org's ethics and guidelines

1.2 Understand and apply security concepts (OSG-9 Chpt 1)

- 1.2.1 Confidentiality, integrity, and availability, authenticity and nonrepudiation
    - **Confidentiality**:
        - Principle that objects are not disclosed to unauthorized subjects
        - Concept of measures used to ensure the protection of the secrecy of data, objects, and resources
        - Confidentiality protections prevent disclosure while protecting authorized access
        - Preserving authorized restrictions on information access and disclosure, including the means for protecting personal privacy and priopríetary information
        - Sensitive data, including personally identifiable information (PII) must be kept confidential; confidentiality is different from secrecy
        - Preserving confidentiality means protecting an asset or data, even if it's not a secret
    - **Integrity**:
        - Principle that objects retain their veracity and are intentionally modified only by authorized subjects
        - Concept of protecting the reliability and correctness of data; guarding against improper info modification/destruction; includes ensuring non-repudiation and authenticity
        - Integrity protection prevents unauthorized alterations of data
        - Preventing authorized subjects from making unauthorized modifications, such as mistakes
        - Maintaining the internal and external consistency of objects
    - **Availability**:
        - Principle that authorized subjects are granted timely and uninterrupted access to objects
        - To ensure high availability of services and data, use techniques like failover clustering, site resiliency, automatic failover, load balancing, redundancy of hardware and software components, and fault tolerance
    - **Authenticity**: ensuring a transmission, message or sender is legitimate
        - See the NIST glossary for examples: https://csrc.nist.gov/glossary/term/authenticity
    - **Nonrepudiation**:

        - Ensures that the subject of activity or who caused an event cannot deny that the event occurred
        - Nonrepudiation is made possible through identification, authentication, authorization, accountability, and auditing

    - **AAA Services**:

        - Identification: claiming to be an identity when attempting to access a secured area or system
        - Authentication: proving that you are that claimed identity via one or more factors (something you have, something you know, something you are)
        - Authorization: defining the needed resources, permissions (i.e. allow/grant and/or deny) to a resource, and object access for a specific identity or subject
        - Auditing: recording a log of the events and activities related to the system and subjects
        - Accounting: (aka accountability) access control process which records info about attempts by all entities to access resources; reviewing log files to check for compliance and violations in order to hold subjects accountable for their actions, especially violations of organizational security policy

1.3 Evaluate and apply security governance principles (OSG-9 Chpt 1)

- **Security governance**: the collection of policies, roles, processes/practices used to make security decisions in an org; related to supporting, evaluating, defining, and directing the security efforts of an org; it involves making sure that security strategies align with business goals, and that they are comprehensive and consistent across the organization
    - Security governance is the implementation of a security solution and a management method that are tightly interconnected
    - There are numerous security frameworks and governance guidelines providing a structured approach to security governance:
        - ISO/IEC 27001: a widely recognized international standard for information security management systems (ISMS); it provides a risk-based approach, and emphasizes coontinual improvement of the ISMS
        - NIST Cybersecurity Framework (CSF): built around six core functions: govern, identify, protect, detect, respond, and recover to provide guidance to industry, government agencies, and other orgs to manage cybersecurity risks
        - SP 800-53: a comprehensive set or catelog of security and privacy controls across multiple security domains, covering areas such as risk management, access control, incident response, and system maintenance
        - SP 800-100: Titled Information Security Handbook: a guide for managers, NIST hasn't released an update since 2006, although they appear to have an update in progress
        - **COBIT (Control Objectives for Information and Related Technologies)**: COBIT focuses enterprise IT, aligning IT and business strategies, and providing a comprehensive framework for managing risks (see additional below)
        - **CIS Critical Security Controls**: the CIS (Center for Internet Security) Critical Security Controls provides a prioritize set of actions to defend against threats; it focuses on practical steps to reduce the attack surface, like implementing secure configurations, managing admin privileges, and monitoring logs

        - **ITIL (Information Technology Infrastructure Library)**: ITIL is a set of practices for IT service management (ITSM) that focuses on aligning IT services with business needs; it includes elements of security governance, particularly in managing security incidents, changes, and service continuity, and is often integrated with other frameworks like ISO 27001
- **The security function**: the aspect of operating a business that focuses on the task of evaluating and improving security over time
    - To manage security, an org must implement proper and sufficient security governance
    - The act of performing a risk assessment to drive the security policy is the clearest and most direct example of management of the security function
- **Third-party governance**: external entity oversight that may be mandated by law, regulation, industry standards, contractual obligation, or licensing requirement; outside investigator or auditors are often involved
- 1.3.1 Alignment of security function to business strategy, goals, mission, and objectives
    - **Security Management Planning**: ensures proper creation/implementation/enforcment of a security policy, and alignment with org strategy, goals, mission, and objectives; security management is based on three types of plans: strategic, tactical, and operational
    - **Strategic Plan**: a strategic plan is a long-term plan (useful for 5 years); it defines the org's security purpose
        - A strategic plan should include a risk assessment
    - **Tactical Plan**: mid-term plan (1 year or less) developed to provide more details on accomplishing the goals set forth in the strategic plan
    - **Operational Plan**: a short-term, highly detailed plan based on strategic or tactical plans
        - Strategy, goals, missions,and objectives — support each other in a heirarchy
        - **Objectives**: are closest to the ground-level and represent small efforts to help you achieve a mission
        - **Missions**: represent a collection of objectives, and one or more missions lead to goals; when you reach your goals, you are achieving the strategy
    - A security framework must closely tie to mission and objectives, enabling the business to complete its objectives and advance the mission while securing the environment based on risk tolerance
- 1.3.2 Organizational processes (e.g., acquistions, divestitures, governance committees)
    - Security governance should address every aspect of an org, including organizational processes of acquisitions, divestitures, and governance
    - Be aware of the risks in acquisitions (since the state of the IT environment to be integrated is unknown, due diligence is key) and divestitures (how to split the IT infrastructure and what to do with identities and credentials)
    - Understand the value of governance committees (vendor governance, project governance, architecture governance, etc.)
    - Executives, managers and appointed individuals meet to review architecture, projects and incidents (security or otherwise),and provide approvals for new strategies or directions
        - The goal is a fresh set of eyes, often eyes that are not purely focused on information security
    - When evaluating a third-party for your security integration, consider the following:
        - on-site assessment
        - document exchange and review

        - process/policy review
        - third-party audit
- 1.3.3 Organizational Roles and Responsibilities
    - Primary security roles are senior manager, security professional, asset owner, custodian, user, and auditor
    - Senior Manager: has a responsibility for organizational security and to maximize profits and shareholder value
    - Security Professional: has the functional responsibility for security, including writing the security policy and implementing it
    - Asset Owner: responsible for classifying information for placement or protection within the security solution
    - Custodian: responsible for the task of implementing the proscribed protection defined by the security policy and senior management
    - Auditor: responsible for reviewing and verifying that the security policy is properly implemented
- 1.3.4 Security control frameworks
    - A **security control framework**: outlines the org's approach to security, including a list of specific security processes, procedures, and solutions used; it is important in planning the structure of an org's security solution; many frameworks to choose from, such as:
        - COBIT Control Objectives for Information and Related Technology ("moderately referenced" on the exam):
            - COBIT is a documented set of best IT security practices by ISACA; a "security concept infrastructure" used to organize the complex security solutions of companies
                - COBIT is commonly used as an audit framework for orgs
                - Six key principles:
                    - provide stakeholder value
                    - holistic approach
                    - dynamic governance system
                    - governance distinct from management
                    - tailored to enterprise needs
                    - end-to-end governance system
        - ISO 27000 series (27000, 27001, 27002, etc.).
        - NIST **CyberSecurity Framework (CSF)**
            - designed for commerical orgs and critical infrastructure, CSF 1.1 consists of five functions:
                - identify
                - protect
                - detect
                - respond
                - recovery
            - note: updated (2024) CSF 2.0 functions:
                - govern
                - identify
                - protect
                - detect

            - respond
            - recover
- 1.3.5 Due care/due diligence
    - **Due diligence**: establishing a plan, policy, and process to protect the interests of the organization; due diligence is knowing what should be done and planning for it; understanding your security governance principles (policies and procedures) and the risks to your organization; actions taken by a vendor to demonstrate or provide due care
        - Due diligence often involves:
            - gathering information through discovery, risk assessments and review of existing documentation
            - developing a formalized security structure containing a security policy, standards, baselines guidlines, and procedures
            - documentation to establish written policies
            - disseminating the information to the organization
    - **Due care**: practicing the individual activities that maintain the due diligence effort; due care is about your legal responsibility within the law or within org policies to implement your org's controls, follow security policies, do the right thing and make reasonable choices
    - Security documentation is the security policy
    - After establishing a framework for governance, security awareness training should be implemented, including all new hires, who complete the security awareness training as they come on board, and existing employees who should recertify regularly (typically yearly)
    - Due care is the responsible protection of assets
    - Due diligence is the ability to prove due care

1.4 Determine compliance and other requirements (OSG-9 Chpt 4)

- 1.4.1 Contractual, legal, industry standards, and regulatory requirements
    - Understand the difference between criminal, civil, and administrative law.
        - **Criminal law**: protects society against acts that violate the basic principles we believe in; violations of criminal law are prosecuted by federal and state governments
        - **Civil law**: provides the framework for the transaction of business between people and organizations; violations of civil law are brought to the court and argued by the two affected parties
        - **Administrative law**: used by government agencies to effectively carry out their day-to-day business
    - **Compliance**: Organizations may find themselves subject to a wide variety of laws, and regulations imposed by regulatory agencies or contractual obligation
        - **Payment Card Industry Data Security Standard (PCI DSS)** - governs the security of credit card information and is enforced through the terms of a merchant agreement between a business that accepts CC payments, and the bank that processes the business' transactions
        - **Sarbanes-Oxley (SOX)** - financial systems may be audited to ensure security controls are sufficient to ensure compliance with SOX
        - **Gramm-Leach-Bliley Act (GLBA)** - affects banks, insurance companies, and credit providers; included a number of limitations on the types of information that could be exchanged even among subsidiaries of the same corp, and required financial institutions to provide written privacy policies to all their customers

        - **Health Insurance Portability and Accountability Act (HIPAA)** - privacy and security regulations requiring strict security measures for hospitals, physicians, insurance companies, and other organizations that process or store private medical information about individuals; also clearly defines the rights of individuals who are the subject of medical records and requires organizations that maintain such records to disclose these rights in writing
        - **Federal Information Security Management Act (FISMA)** - requires federal agencies to implement an information security program that covers the agency's operations and contractors
        - **Computer Fraud and Abuse Act (CFAA)** (as amended) - protects computers used by the government or in interstate commerce from a variety of abuses
        - **Electronic Communications Privacy Act (ECPA)** - in short, makes it a crime to invade the electronic privacy of an individual; passed in 1986 to expand and revise federal wiretapping and electronic eavesdropping provisions, making it a crime to intercept or procure electronic communications, and includes important provisions that protect a person’s wire and electronic communications from being intercepted by another private individual
        - **Digital Millennium Copyright Act (DMCA)** - prohibits the circumvention of copyright protection mechanisms placed in digital media and limits the liability of internet service providers for the activities of their users
- 1.4.2 Privacy requirements
    - European Union's **General Data Protection Regulation (GDPR)** - replaced Data Protection Directive (DPD), purpose is to provide a single, harmonized law that covers data throughout the EU
        - Lawfulness, fairness, and transparency
        - Purpose Limitation
        - Data Minimization
        - Accuracy
        - Storage Limitation
        - Security
        - Accountability
    - Personal Information Protection and Electronic Documents Act (PIPEDA): Canadian law that governs the use of personal information
    - California Consumer Privacy Act (CCPA)
    - Be familiar with the EU Data Protection Directive
    - Be familiar with the requirements around healthcare data, credit card data and other PII data as it relates to various countries and their laws and regulations

1.5 Understand legal and regulatory issues that pertain to information security in a holistic context (OSG-9 Chpt 4)

- 1.5.1 Cybercrimes and data breaches
    - Understand the notification requirements placed on organizations that experience a data breach
    - California's SB 1386 implemented the first statewide requirement to notify individuals of a breach of their personnel information; all other states eventually followed suit with similar laws

  - Currently, federal law only requires notification of individuals when a HIPAA-covered entity breaches their protected health information (likely to soon change)
  - Before an org expands to other countries, perform due diligence to understand legal systems and what changes might be required to the way that data is handled and secured
  - In particular, be familiar with:
    - **Council of Europe Convention on Cybercrime**: a treaty signed by many countries that establishes standards for cybercrime policy
    - Laws about data breaches, including notification requirements
    - In the US, the **Health Information Technology for Economic and Clinical Health (HITECH)** Act requires notification of a data breach in some cases, such as when the personal health information was not protected as required by HIPAA
    - GLBA (Gramm-Leach-Bliley Act) applies to insurance and financial orgs, requiring notification to federal regulators, law enforcement agencies and customers when a data breach occurs
    - Certain states also impose their own requirements concerning data breaches
    - the EU and other countries have their own requirements, for instance, the GDPR has very strict data breach notification requirements: A data breach must be reported to the competent supervisory authority within 72 hours of its discovery
    - **Communications Assistance to Law Enforcement Act (CALEA)**: requires all communication carriers make wiretaps possible for law enforcement officials who have an appropriate court order
    - Some countries do not have any reporting requirements
- 1.5.2 Licensing and intellectual property (IP) requirements
  - **Intellectual property**: intangible assets (e.g. software, data)
  - **Trademarks**: words, slogans, and logos used to identify a company and its products or services
  - **Patents**: provide protection to the creators of new inventions; a temporary monopoly for producing a specific item such as a toy, which must be novel and unique to qualify for a patent
    - **Utility**: protect the intellectual property rights of inventors
    - **Design**: cover the appearance of an invention and last for 15 years; note design patents don't protect the idea of an invention only its form, and are generally seen as weaker
    - Software: area of on-going controversy; Google vs Oracle; given to rise of "patent trolls"
  - **Copyright**: protects original works of authoriship, such as books, articles, poems, and songs; exclusive use of artistic, musical or literary works which prevents unauthorized duplication, distribution or modification
  - **Licensing**: a contract between the software producer and the consumer which limits the use and/or distribution of the software
  - **Trade Secrets**: trade secret laws protect the operating secrets of a firm; trade secrets are intellectual property that is critical to a business, and significant damage would result if it were disclosed to competitors or the public; the Economic Espionage Act imposes fines and jail sentences on someone found guilty of stealing trade secrets from a US corp
- 1.5.3 Import/export controls
  - Every country has laws around the import and export of hardware and software; e.g. the US has restrictions around the export of cryptographic technology, and Russia requires a license to import encryption technologies manufactured outside the country
- 1.5.4 Transborder data flow

  - Orgs should adhere to origin country-specific laws and regulations, regardless of where data resides
  - Also be aware of applicable laws where data is stored and systems are used
  - **International Traffic in Arms Regulations (ITAR)**: a US regulation that was built to ensure control over any export of items such as missiles, rockets, bombs, or anything else existing in the United States Munitions List (USML)
  - **Export Administration Regulations (EAR)**: EAR predominantly focuses on commercial use-related items like computers, lasers, marine items, and more; however, it can also include items that may have been designed for commercial use but actually have military applications
- 1.5.5 Privacy

  - Many laws include privacy protections for personal data

    - The EU's GDPR has strong privacy rules that apply to any org anywhere that stores or processes the personal data of EU residents; these individuals must be told how their data is collected and used, and they must be able to opt out

  - The privacy guidelines of the **Organization for Economic Co-operation and Development (OECD)** require orgs to avoid unjustified obstacles to trans-border data flow, set limits to personal data collection, protect personal data with reasonable security and more

  - Fourth Amendment to the US Constitution: the right of the people to be secure in their persons, houses, papers, effects against unreasonable search and seizure

  - **Electronic Communication Privacy Act (ECPA)**: as amended, protects wire, oral, and electronic communications while those communications are being made, are in transit, and when they are stored on computers; makes it a crime to invade electronic privacy of an individual, and it broadened the Federal Wiretap Act

  - HIPAA: see above

  - HITECH: see above

  - California SB 1386 (2002): immediate disclosure to individuals for PII breach

  - **California Consumer Privacy Act (CCPA)**: The CCPA applies to:

    - For-profit businesses that collect consumers' personal information (or have others collect personal information for them),
    - Determine why and how the information will be processed,
    - Do business in California and meet any of the following:
      - have a gross annual revenue > $25 million;
      - buy, sell, or share the personal information of 100k or more California residents or households; or
      - get 50% or more of their annual revenue from selling or sharing California residents' personal information
    - The CCPA imposes separate obligations on service providers and contractors (who contract with businesses to process personal info) and other recipients of personal information from businesses
    - The CCPA does not generally apply to nonprofit orgs or government agencies

- California residents have the right to:
  - (L)imit use and disclosure of personal info
  - (O)pt-out of sale or cross-context advertising
  - (C)orrect inaccurate info
  - (K)now what personal info business have and share
  - (E)qual treatment / nondiscrimination
  - (D)elete info business have on them

- **Children's Online Privacy Protection Act (COPPA)** of 1998:
  - COPPA makes a series of demands on websites that cater to children or knowingly collect information from children:
    - Websites must have a privacy notice that clearly states the types of info they collect and what it's used for (including whether infor is disclosed to third parties); must also include contact info for site operators
    - Parents must be able to review any info collected from children and permanently delete it from the site's records
    - Parents must give verifiable consent to the collection of info about children younger than the age of 13 prior to any such collection

- GLBA: see above

- US Patriot Act of 2002: enacted following the September 11 attacks with the stated goal of tightening U.S. national security, particularly as it related to foreign terrorism
  - The act included three main provisions:
    - expanded surveillance abilities of law enforcement, including by tapping domestic and international phones
    - easier interagency communication to allow federal agencies to more effectively use all available resources in counterterrorism efforts
    - increased penalties for terrorism crimes and an expanded list of activities which would qualify for terrorism charges

- **Family Education Rights and Privacy Act (FERPA)**: Grants privacy rights to students over 18, and the parents of minor students

- EU's Data Protection Directive (DPD): see above

- EU's General Data Protection Regulation (GDPR): key provisions
  - lawfulness, fairness, and transparency
  - purpose limitation
  - data minimization
  - accuracy
  - storage limitation
  - security
  - accountability

- The EU-US **Privacy Shield** (formerly the EU-US Safe Harbor agreement): controls data flow from the EU to the United States; the EU has more stringent privacy protections and without the Privacy Shield, personal data flow from the EU to the United States would not be allowed

1.6 Understand requirements for investigation types (i.e., administrative, criminal, civil, regulatory, industry standards) (OSG-9 Chpt 19)

- An investigation will vary based on incident type; e.g. for a financial services company, a financial system compromise might cause a regulatory investigation; a system breach or website compromise might cause a criminal investigation; each type of investigation has special considerations:
    - **Administrative**: an administrative investigation has a primary purpose of providing the appropriate authorities with incident information; thereafter, authorities will determine the proper action, if any
        - Administrative investigations are often tied to HR scenarios, such as when a manager has been accused of improprieties
    - **Criminal**: a criminal investigation occurs when a crime has been committed and you are working with a law enforcement agency to convict the alleged perpetrator; in such a case, it is common to gather evidence for a court of law, and to share the evidence with the defense
        - You need to gather and handle the information using methods that ensure the evidence can be used in court
        - In a criminal case, a suspect must be proven guilty beyond a reasonable doubt; a higher bar compared to a civil case, which is showing a preponderance of evidence
    - **Civil**: in a civil case, one person or entity sues another; e.g. one company could sue another for a trademark violation
        - A civil case is typically about monetary damages, and doesn't involve criminality
        - In a civil case, a preponderance of evidence is required to secure a victory; differing from criminal cases, where a suspect is innocent until proven guilty beyond a reasonable doubt
    - **Industry Standards**: an industry standards investigation is intended to determine whether an org is adhering to a specific industry standard or set of standards, such as logging and auditing failed logon attempts
        - Because industry standards represent well-understood and widely implemented best practices, many orgs try to adhere to them even when they are not required to do so in order to improve security, and reduce operational and other risks
    - **Regulatory**: A regulatory investigation is conducted by a regulatory body, such as the Securities and Exchange Commission (SEC) or Financial Industry Regulatory Authority (FINRA), against an org suspected of an infraction
        - Here the org is required to comply with the investigation, e.g., by not hiding or destroying evidence

1.7 Develop, document, and implement security policy, standards, procedures and guidelines (OSG-9 Chpt 1)

- To create a comprehensive security plan, you need the following items: security policy, standards, baselines, guidelines, and procedures

- The top tier of a formalized hierarchical organization security documentation is the security policy

    - **Policy**: docs created by and published by senior management describing organizational strategic goals

    - A security policy is a document that defines the scope of security needed by the org, discussing assets that require protection and the extent to which security solutions should go to provide the necessary protections
    - It defines the strategic security objectives, vision, and goals and outlines the security framework of the organization

- **Acceptable User Policy**: the AUP is a commonly produced document that exists as part of the overall security documentation infrastructure

    - This policy defines a level of acceptable performance and expectation of behavior and activity; failure to comply with the policy may result in job action warnings, penalties, or termination

- Security Standards, Baselines and Guidelines: once the main security policies are set, the remaining security docuemntation can be crafted from these policies

    - **Policies**: these are high-level documents, usually written by the management team; policies are mandatory, and a policy might provide requirements, but not the steps for implementation
    - **Standards**: specific mandates explicity stating expections of performance/conformance; more descriptive than policies, standards define compulsary requirements for the homogenous use of hardware, software, technology, and security controls, uniformly implemented throughout the org
    - **Baseline**: defines a minimum level of security that every system throughout the organization must meet; baselines are usually system specific and refer to industry / government standards
        - e.g. a baseline for server builds would be a list of configuration areas that should be applied to every server that is built
        - A Group Policy Object (GPO) in a Windows network is sometimes used to comply with standards; configuration management solutions can also help you establish baselines and spot configurations that are not in alignment
    - **Guideline**: offers recommendations on how standards and baselines should be implemented & serves as an operational guide for security professionals and users
        - Guidelines are flexible, and can be customized for unique systems or conditions; they state which security mechanism should be deployed instead of prescribing a specific product or control; they are not complusory; suggested practices and expectations of activity to best accomplish tasks and goals
    - **Procedure** (AKA Standard Operating Procedure or SOP): detailed, step-by-step how-to doc that describes the exact actions necessary to implement a specific security mechanism, control, or solution

1.8 Identify, analyze, and prioritize Business Continuity (BC) requirements (OSG-9 Chpt 3)

- **Business Continuity Planning (BCP)**: involves assessing the risk to organizational processes and creating policies, plans, and procedures to minimize the impact those risks might have on the organization if they were to occur

    - BCP is used to maintain the continuous operation of a business in the event of an emergency, with a goal to implement a combination of policies, procedures, and processes
    - BCP has four distinct phases:
        - project scope and planning
        - business impact analysis

        - continuity planning
        - approval and implementation
    - Business continuity requires a lot of planning and preparation; actual implementation of business continuity processes occur quite infrequently
    - The primary facets of business continuity are:
        - Resilience: (e.g. within a data center and between sites or data centers)
        - Recovery: if a service becomes unavailable, you need to recover it as soon as possible
        - Contingency: a last resort in case resilience and recovery prove ineffective

- BCP vs DR:

    - BCP activities are typically strategically focused at a high level and center themselves on business processes and operations
    - DR plans tend to be more tactical and describe technical activities such as recovery sites, backups, and fault tolerance

- The overall goal of BCP is to provide a quick, calm, and efficient response in the event of an emergency and to enhance a company's ability to quickly recover from a distruptive event

- 1.8.2 Develop and document the scope and the plan

    - The BCP process has four main steps:

        1. Project scope and planning
        2. Business Impact Analysis
        3. Continuity planning
        4. Approval and implementation

        - **Project scope and planning**: Developing the project scope and plan starts with gaining support of the management team, making a business case (cost/benefit analysis, regulatory or compliance reasons etc.) and gaining approval to move forward
            - Next, you need to form a team with representatives from the business as well as IT
            - Next, start with a business continuity policy statement,conduct a business impact analysis (see next item), and then develop the remaining components:
                - preventive controls
                - relocation
                - the actual continuity plan
                - testing
                - training and maintenance

- 1.8.1 Business Impact Analysis (BIA)

    - **Business impact analysis (BIA)**: Identify the systems and services that the business relies on and assess the impacts that a disruption or outage would cause, including the impacts on business processes like accounts receivable and sales
        - Step 1: Identification of priorities
        - Step 2: Risk identification
        - Step 3: Likelihood assessment
        - Step 4: Resource prioritization

        - deciding which systems and services you need to get things running again (think foundational IT services such as the network and directory, which many other systems rely on)
        - and prioritize the order in which critical systems and services are recovered or brought back online
    - As part of the BIA, establish:
        - **recovery time objectives (RTO)**: how long it takes to recover
        - **recovery point objectives (RPO)**: the maximum tolerable data loss
        - **maximum tolerable downtime (MTD)**: (AKA maximum allowable downtime or MAD) how long an org can survive an interruption of critical functions
        - along with the costs of downtime and recovery
  - **Continuity planning**: The first two phases of the BCP process (project scope and planning and the business impact analysis) focus on determining how the BCP process will work and prioritizing the business assets that need to be protected against interruption
    - The next phase of BCP development, continuity planning, focuses on the development and implementation of a continuity strategy to minimize the impact realized risks might have on protected assets
    - There are two primary subtasks/phases involved in continuity planning:
        - **Strategy development**: in this phase, the BCP team determines which risks they will mitigate
        - **Provisions and processes**: in this phase, the team designs mechanisms and procedures that will mitigate identified risks
    - The goal of this process is to create a **continuity of operations plan** (COOP), which focuses on how an org will carry out critical business functions starting shortly after a disruption occurs and extending up to one month of sustained operations
  - **Approval and implementation**:
    - BCP plan now needs sr. management buy-in (should be endorsed by the org's top exec)
    - BCP team should create an implementation schedule, and all personnel involed should receive training on the plan

- The top priority of BCP and DRP is people: **Always prioritize people's safety**; get people out of harm's way, and then address IT recovery and restoration issues

1.9 Contribute to and enforce personnel security policies and procedures (OSG-9 Chpt 2)

- People are often considered the weakest element in any security solution; no matter what physical or logical controls are deployed, humans can discover ways of to avoid them, circumvent/subvert them, or disable them

  - Malicious actors are routinely targeting users with phishing and spear phishing campaigns, social engineering, and other types of attacks, and everybody is a target
  - Once attackers compromise an account, they can use that entry point to move around the network and elevate their privileges
  - People can also become a key security asset when they are properly trained and are motivated to protect not only themselves but the security of the organization as well
  - Part of planning for security includes having standards in place for job descriptions, job classifications, work tasks, job responsibilities, prevention of collusion, candidate screening, background checks, security clearances, employment and nondisclosure agreements

- 1.9.1 Candidate screening and hiring
    - The following strategies can reduce your risk:
        - **Candidate screening and hiring**: To properly plan for security, you shold have standards in place for job descriptions, job classification, work tasks, job responsibilities, prevention of collusion, candidate screening, background checks, security clearances, employment agreements, and nondisclosure agreements
            - screening employment candidates thoroughly is a key part of the hiring process
            - be sure to conduct a full background check that includes a criminal records check, job history verification, education verification, certification validation and confirmation of other accolades when possible
            - all references should be contacted
- 1.9.2 Employement agreements and policies
    - **Employment agreement**: specifies job duties, expectations, rate of pay, benefits and info about termination; sometimes, such agreements are for a set period (for example, in a contract or short-term job).
        - Employment agreements facilitate termination when needed for an underperforming employee
        - The more info and detail in an employment agreement, the less risk (risk of a wrongful termination lawsuit, for example) the company has during a termination proceeding
        - e.g. a terminated employee might take a copy of their email with them without thinking of it as stealing, but they are less likely to do so if an employment agreement or another policy document clearly prohibits it
        - example employee agreements:
            - non-compete
            - codes of conduct such as an acceptable use policy (AUP), which defines what is and isn’t acceptable acitivty, practice, or use for company equipemnt and resources
            - nondisclosure agreement (NDA), which is a doc used to protect confidential information from being disclosed by a current or former employee
- 1.9.3 Onboarding, transfers and termination processes
    - **Onboarding**: process of bringing a new employee into the org
        - creating documented processes allowing the new employee to be intgrated quickly and consistently
    - **Transfer**: an employee moves from one job to another, likely requiring adjusted account access to maintain appropriate least privilege
    - **Termination or offboarding**: offboarding is the removal of an employee's identity from the IAM system, once that person has left the org; can also be an element used when an employee transfers into a new role
        - whether cordial or abrupt, the ex-employee should be escorted off the premises and not allowed to return
- 1.9.4 Vendor, consultant, and contractor agreements and controls

    - Orgs commonly outsource many IT functions, particularly data center hosting, contact-center support, and application development
    - Info security policies and procedures must address outsourcing secuity and the use of service providers, vendors and consultants
        - e.g. access control, document exchange and review, maintenance, on-site assessment, process and policy review, and Service Level Agreements (SLAs) are examples of outsourcing security considerations

- 1.9.5 Compliance policy requirements

    - **Compliance**: the act of confirming or adhering to rules, policies, regulations, standards, or requirements
        - on a personnel level, compliance is related to individual employees following company policies and procedures
        - employees need to be trained on company standards as defined in the security policy and remain in compliance with any contractual obligations (e.g. with PCI DSS)
    - Compliance is a form of administrative or managerial security control
    - **Compliance enforcement**: the application of sanctions or consequences for failing to follow policy, training, best practices, or regulations

- 1.9.6 Privacy policy requirements

    - Personally identifiable information (PII) about employees, partners, contractors, customers and others should be stored in a secure way, accessible only to those who require the information to perform their jobs
    - Orgs should maintain a documented privacy policy outlining the type of data covered by the policy and who the policy applies to
    - Employees and contractors should be required to read and agree to the privacy policy upon hire and on a regular basis thereafter (such as annually)

1.10 Understand and apply risk management concepts (OSG-9 Chpt 2)

- 1.10.1 Privacy policy requirements

    - **Risk Management**: process of identifying factors that could damage or disclose data, evaluating those factors in light of data value and countermeasure cost, and implementing cost-effective solutions for mitigating or reducing risk
    - **Threats**: any potential occurrence that many cause an undersirable or unwanted outome for a specific asset; they can be intentional or accidental; loosely think of a threat as a weapon that could cause harm to a target
    - **Vulnerability**: the weakness in an asset, or weakness (or absense) of a safeguard or countermeasure; a flaw, limitation, error, frailty, or susceptibility to harm
    - Threats and vulnerabilities are related: a threat is possible when a vulnerability is present
        - Threats exploit vulnerabilities, which results in exposure
        - Exposure is risk, and risk is mitigated by safeguards
        - Safeguards protect assets that are endangered by threats
        - **Threat Agent/Actors**: intentionally exploit vulnerabilities
        - **Threat Events**: accidential occurrences and intentinoal exploitations of vulnerabilities

        - **Threat Vectors**: (AKA attack vector) is the path or means by which an attack or attacker can gain access to a target in order to cause harm
        - **Exposure**: being susceptible to asset loss because of a threat; the potential for harm to occur
        - **Exposure Factor (EF)**: derived from this concept; an element of quantitative risk analysis that represents the percentage of loss than org would experience if a specific asset were violated by a realized risk
        - **Single Loss Expectancy (SLE)**: an element of quantitative risk analysis that represents the cost associated with a single realized risk against a specific asset; SLE = asset value (AV) * exposure factor (EF)
        - **Annualized rate of occurrence (ARO)**: an element of quantitative risk analysis that represent the expected frequency with which a specific threat or risk will occur within a single year
        - **Annualized loss expectancy (ALE)**: an element of quantitative risk analysis that represent the possible yearly cost of all instances of a specific realized threat against a specific asset; ALE = SLE * ARO
        - **Safeguard evaluation**: ALE for an asset if a safeguard is implemented; ALE before safeguard - ALE with safeguard - annual cost of safeguard, or (ALE1 - ALE2) - ACS
        - **Risk**: the possiblity or likelihood that a threat will exploit a vulnerability to cause harm to an asset and the severity of damage that could result; the > the potential harm, the > the risk

- 1.10.1 Risk assessment/analysis

    - **Risk Assessment**: used to identify the risks and set criticality priorities, and then risk response is used to determine the best defense for each identified risk
    - Risk is threat with a vulnerability
    - Risk = threat * vulnerability (or probability of harm multiplied by severity of harm)
    - Addressing either the threat or threat agent or vulnerability directly results in a reduction of risk (known as threat mitigation)
    - All IT systems have risk; all orgs have risk; there is no way to elminiate 100% of all risks
        - Instead upper management must decide which risks are acceptable, and which are not; there are two primary risk-assessment methodologies:
            - **Quantitative Risk Analysis**: assigns real dollar figures to the loss of an asset and is based on mathematical calculations
            - **Qualitative Risk Analysis**: assigns subjective and intangible values to the loss of an asset and takes into account perspectives, feelings, intuition, preferences, ideas, and gut reactions; qualitative risk analys is based more on scenarios than calculations, and threats are ranked to evaluate risks, costs, and effects
        - Most orgs employ a hybrid of both risk assessment methodologies
        - The goal of risk assessment is to identify risks (based on asset-threat parings) and rank them in order of criticality

- 1.10.3 Risk response

    - **Risk response**: the formulation of a plan for each identified risk; for a given risk, you have a choice for a possible risk response:

        - **Risk Mitigation**: reducing risk, or risk mitigation, is the implementation of safeguards, security controls, and countermeasures to reduce and/or eliminate vulnerabilities or block threats
        - **Risk Assignment**: assigning or transferring risk is the placement of the responsibility of loss due to a risk onto another entity or organization; AKA assignment of risk and transference of risk
        - **Risk Deterrence**: deterrence is the process of implementing deterrents for would-be violators of security and policy
            - the goal is to convince a threat agent not to attack
            - e.g. implementing auditing, security cameras, and warning banners; using security guards
        - **Risk Avoidance**: determining that the impact or likelihood of a specific risk is too great to be offset by potential benefits, and not performing a particular business function due to that determiniation; the process of selecting alternate options or activities that have less associated risk than the default, common, expedient, or cheap option
        - **Risk Acceptance**: the result after a cost/benefit analysis determines that countermeasure costs would outweigh the possible cost of loss due to a risk
            - also means that management has agreed to accept the consequences/loss if the risk is realized
        - **Risk Rejection**: an unacceptable possible response to risk is to reject risk or ignore risk; denying that risk exists and hoping that it will never be realized are not valid prudent due care/due diligence responses to risk
        - **Risk Transference**: paying an external party to accept the financial impact of a given risk
    - **Inherent Risk**: the level of natural, native, or default risk that exists in an environment, system, or product prior to any risk management efforts being performed (AKA initial or starting risk); this is the risk identified by the risk assessment process
    - **Residual Risk**: consists of threats to specific assets against which management chooses not to implement (the risk that management has chosen to accept rather than mitigate); risk remaining after security controls have been put in place
    - **Total Risk**: the amount of risk an org would face if no safeguards were implemented
    - **Conceptual Total Risk Formula**: threats * vulnerabilities * asset value = total risk
    - **Controls Gap**: amount of risk that is reduced by implementing safeguards, or the difference between total risk and residual risk
    - **Conceptual Residual Risk Formula**: total risk - controls gap = residual risk
    - Risk should be reassessed on a periodic basis to maintain reasonable security because security changes over time

- 1.10.4 Countermeasure selection and implementation

    - **Countermeasure**: AKA a "control" or "safeguard" can help reduce risk
        - For exam prep, understand how the concepts are integrated into your environment; this is not a step-by-step technical configuration, but the process of the implementation — where you start, in which order it occurs and how you finish
        - Bear in mind that security should be designed to support and enable business tasks and functions

            - security controls, countermeasures, and safeguards can be implemented administratively, logically / technically, or physically
            - these 3 categories should be implemented in a conceptual layered defense-in-depth manner to provide maximum benefit
            - based on the concept that policies (part of administrative controls) drive all aspects of security and thus form the initial protection layer around assets
            - then, logical and technical controls provide protection against logical attacks and exploits
            - then, physical controls provide protection against real-world physical attacks against facilities and devices

- 1.10.5 Applicable types of controls (e.g., preventive, detective, corrective)

    - **Administrative**: the policies and procedures defined by an org's security policy and other regulations or requirements
    - **Physical**: security mechanisms focused on providing protection to the facility and real world objects
    - **Preventive**: a preventive or preventative control is deployed to thwart or stop unwanted or unauthorized activity from occurring
    - **Deterrent**: a deterrent control is deployed to discourage security policy violations; deterrent and preventative controls are similar, but deterrent controls often depend on individuals being convinced not to take an unwanted action
    - **Detective**: a detective control is deployed to discover or detect unwanted or unauthorized activity; detective controls operate after the fact
    - **Compensating**: a compensating control is deployed to provide various options to other existing controls, to aid in enforcement and support of security policies
        - they can be any controls used in addition to, or in place of, another control
        - they can be a means to improve the effectiveness of a primary control or as the alternative or failover option in the event of a primary control failure
    - **Corrective**: a corrective control modifies the environment to return systems to normal after an unwanted or unauthorized activity as occurred; it attempts to correct any problems resulting from a security incident
    - **Recovery**: An extension of corrective controls but have more advanced or complex abilities; a recovery control attempts to repair or restore resources, functions, and capabilities after a security policy violation
        - recovery controls typically address more significant damaging events compared to corrective controls, especially when security violations may have occurred
    - **Directive**: A directive control is deployed to direct, confine, or control the actions of subjects to force or encourage compliance with security policies

- 1.10.6 Control assessments (security and privacy)

    - Periodically assess security and privacy controls: what’s working, what isn’t
        - As part of this assessment, the existing documents should be thoroughly reviewed, and some of the controls tested randomly
        - A report is typically produced to show the outcomes and enable the org to remediate deficiencies

        - Often, security and privacy control assessment are performed and/or validated by different teams, with the privacy team handling the privacy aspects

- 1.10.7 Monitoring and measurement

    - Monitoring and measurement are closely aligned with identifying risks
    - While monitoring is used for more than security purposes, monitoring should be tuned to ensure the org is notified about potential security incidents as soon as possible
    - If a security breach occurs, monitored systems and data become valuable from a forensics perspective; rrom the ability to derive root cause of an incident to making adjustments to minimize the chances of reoccurance

- 1.10.8 Reporting

    - Risk Reporting is a key task to perform at the conclusion of risk analysis (i.e. production and presentation of a summarizing report)
    - A Risk Register or Risk Log is a document that inventories all identified risks to an org or system or within an individual project
        - A risk register is used to record and track the activities of risk management, including:
            - identifying risks
            - evaluating the severity of, and prioritizing those risks
            - prescribing responses to reduce or eliminate the risks
            - track the progress of risk mitigation

- 1.10.9 Continuous improvement (e.g., Risk maturity modeling)

    - Risk analysis is performed to provide upper management with the details necessary to decide which risks should be mitigated, which should be transferred, which should be deterred, which should be avoided, and which should be accepted; to fully evaluate risks and subsequently take proper precautions, the following must be analyzed:

        - assets
        - asset valuation
        - threats
        - vulnerabilities
        - exposure
        - risk
        - realized risk
        - safeguards
        - countermeasures
        - attacks
        - breaches

    - An **Enterprise Risk Management** (ERM) program can be evaluated using an RMM

    - **Risk Maturity Model (RMM)**: assesses the key indicators and activities of a mature, sustainable, and repeatable risk management process, typically relating the assessment of risk maturity against a five-level model such as:

        - **Ad hoc**: a chaotic starting point from which all orgs initiate risk management

    - **Preliminary**: loose attempts are made to follow risk management processes, but each department may perform risk assessment uniquely
    - **Defined**: a common or standardized risk framework is adopted organization-wide
    - **Integrated**: risk management operations are integrated into business processes, metrics are used to gather effectiveness data, and risk is considered an element in business strategy decisions
    - **Optimized**: risk management focuses on achieving objectives rather than just reacting to external threats; increased strategic planning is geared toward business success rather than just avoiding incidents; and lessons learned are re-integrated into the risk management process

- 1.10.10 Risk frameworks
  - A risk framework is a guide or recipe for how risk is to be accessed, resolved, and monitored
  - NIST established the **Risk Management Framework** (RMF) and the **Cybersecurity Framework** (CSF): the CSF is a set of guidelines for mitigating organizational cybersecurity risks, based on existing standards, guidelines, and practices
  - The RMF is intended as a risk management process to identify and respond to threats, and is defined in three core, interrelated Special Publications:
    - SP 800-37 Rev 2, Risk Management Framework for Information Systems and Organizations
    - SP 800-39, Managing Information Security Risk
    - SP 800-30 Rev 1, Guide for Conducting Risk Assessments
    - The **RMF has 7 steps**, and **six cyclical phases**:
      - **Prepare** to execute the RMF from an organization and system-level perspective by establishing a context and priorities for managing security and privacy risk
      - **Categorize** the system and the information processed, stored, and transmitted by the system based on an analysis of the impact of loss
      - **Select** an initial set of controls for the system and tailor the controls as needed to reduce risk to an acceptable level based on an assessment of risk
      - **Implement** the controls and describe how the controls are employed within the system and its environment of operation
      - **Assess** the controls to determine if the controls are implemented correctly, operating as intended, and producing the desired outcomes with respect to satisfying the security and privacy requirements
      - **Authorize** the system or common controls based on a determination that the risk to organizational operations and assets, individuals, and other organizations, and the nation is acceptable
      - **Monitor** the system and associated controls on an on-going basis to include assessing control effectiveness, documenting changes to the system and environment of operation, conducting risk assessments and impact analysis, and reporting the security and privacy posture of the system
    - See my overview article, The NIST Risk Management Framework
  - There are other risk frameworks, such as the ISO/IEC 31000, ISO/IEC 31004, COSO, Risk IT, OCTAVE, FAIR, and TARA; be familiar with frameworks and their goals

1.11 Understand and apply threat modeling concepts and methodologies (OSG-9 Chpt 1)

- **Threat Modeling**: security process where potential threats are identified, categorized, and analyzed; can be performed as a proactive measure during design and development (aka defensive approach) or as an reactive measure once a product has been deployed (aka adversarial approach)
    - Threat modeling identifies the potential harm, the probability of occurrence, the priority of concern, and the means to eradicate or reduce the threat
- Microsoft uses the **Security Development Lifecycle** (SDL) with the motto: "Secure by design, secure by default, secure in deployment and communication"
    - It has two objectives:
        - Reduce the number of security-related design and coding defects
        - Reduce the severity of any remaining defects
- A defensive approach to threat modeling takes place during the early stages of development; the method is based on predicting threats and designing in specific defenses during the coding and crafting process
    - Security solutions are more cost effective in this phase than later; this concept should be considered a proactive approach to threat management
- Microsoft developed the **STRIDE threat model**:
    - Spoofing: an attack with the goal of gaining access to a target system through the use of falsified identity
    - Tampering: any action resulting in unauthorized changes or manipulation of data, whether in transit or in storage
    - Repudiation: the ability of a user or attacker to deny having performed an action or activity by maintaining plausible deniability
    - Information Disclosure: the revelation or distribution of private, confidential, or controlled information to external or unauthorized entities
    - Denial of Service (DoS): an attack that attempts to prevent authorized use of a resource; this can be done through flaw exploitation, connection overloading, or traffic flooding
    - Elevation of privilege: an attack where a limited user account is transformed into an account with greater privileges, powers, and access
    - STRIDE is threat categorization model; threat categorization is an important part of app threat modeling
- **Process for Attack Simulation and Threat Analysis (PASTA)**: a seven-stage threat modeling methodology:
    - Stage I: Definition of the Objectives (DO) for the Analysis of Risk
    - Stage II: Definition of the Technical Scope (DTS)
    - Stage III: Application Decomposition and Analysis (ADA)
    - Stage IV: Threat Analysis (TA)
    - Stage V: Weakness and Vulnerability Analysis (WVA)
    - Stage VI: Attack Modeling and Simulation (AMS)
    - Stage VII: Risk Analysis and Management (RAM)
- Each stage of PASTA has a specific list of objectives to achieve and deliverables to produce in order to complete the stage
- **Visual, Agile, and Simple Threat (VAST)**: a threat modeling concept that integrates threat and risk management into an Agile programming environment on a scalable basis
- Part of the job of the security team is to identify threats, using different methods:
    - Focus on attackers: this is a useful method in specific situations;

    - e.g. suppose that a developer's employment is terminated, and that post-offboarding and review of developer's computer, a determination is made that the person was disgruntled and angry
        - understanding this situation as a possible threat, allows mitigation steps to be taken
    - Focus on assets: an org's most valuable assets are likely to be targeted by attackers
    - Focus on software: orgs that develop applications in house, and can be viewed as part of the threat landscape; the goal isn't to identify every possible attack, but to focus on the big picture, identifying risks and attack vectors
- Understanding threats to the org allow the documentation of potential attack vectors; diagramming can be used to list various technologies under threat
- **Reduction analysis**: with a purpose of gaining a greater understanding of the logic of a product and interactions with external elements includes breaking down a system into five core elements: trust boundaries, data flow paths, input points, privileged operations, and security control details; AKA decomposing the application, system, or environment
- **DREAD**: Microsoft developed the DREAD threat modeling approach to detect and prioritize threats so that serious threats can be mitigated first
    - D: Damage potential
    - R: Reproducibility
    - E: Exploitability
    - A: Affected users
    - D: Discoverability

1.12 Apply Supply Chain Risk Management (SRM) concepts (OSG-9 Chpt 1)

- 1.12.1 Risks associated with hardware, software, and services
    - **Supply Chain Risk Management (SCRM)**: the means to ensure that all of the vendors or links in the supply chain are:
        - reliable,
        - trustworthy,
        - reputable organizations that disclose their practices and security requirements to their business partners (not necessarily to the public)
    - Each link in the chain should be responsible and accountable to the next link in the chain; each handoff is properly organized, documented, managed, and audited
        - The goal of a secure supply chain is that the finished product is of sufficient quality, meets performance and operational goals, provides stated security mechanisms, and that at no point in the process was any element counterfeited or subject to unauthorized or malicious manipulation or sabotage
    - The supply chain can be a threat vector, where materials, software, hardware, or data is being obtained from a supposedly trusted source but the supply chain behind the source could have been compromised and asset poisoned or modified
- 1.12.2 Third-party assessment and monitoring
    - Before doing business with another company, an org needs to perform due-dilligence, and third-party assessments can help gather information and perform the assessment
    - An on-site assessment is useful to gain information about physical security and operations
        - During document review, your goal is to thoroughly review all the architecture, designs, implementations, policies, procedures, etc.

        - A good understanding of the current state of the environment, especially to understand any shortcomings or compliance issues prior to integrating the IT infrastructures
        - The level of access and depth of information obtained is usually proportional to how closely the companies will work together
- 1.12.3 Minimum security requirements
    - As part of assessment, the minimum security requirements must be established; in some cases, the minimum security requirements are your company's security requirements, in other cases, new minimum security requirements need to be established
        - In such scenarios, the minimum security requirements should have a defined period
- 1.12.4 Service-level requirements
    - **Service Level Agreements (SLAs)**: companies have SLAs for internal operations (such as how long it takes for the helpdesk to respond to a new ticket), for customers (such as the availability of a public-facing service) and for partner orgs (such as how much support a vendor provides a partner)
        - All SLAs should be reviewed; a company sometimes has an SLA standard that should be applied, when possible, to the service level agreements as part of working with another company
            - this can sometimes take time, as the acquiring company might have to support established SLAs until they expire or are up for renewal

1.13 Establish and maintain a security awareness, education, and training program (OSG-9 Chpt 2)

- 1.13.1 Methods and techniques to present awareness and training (e.g., social engineering, phishing, security champions, gamification)

    - Before actual training takes place, user security awareness needs to take place; from there, training, or teaching employees to perform their work tasks and to comply with the security policy can begin
        - All new employees require some level of training so that they will be able to comply with all standards, guidelines, and procedures mandated by the security policy
        - Education is a more detailed endeavor in which students/users learn much more than they actually need to know to perform their work tasks
        - Education is most often associated with users pursuing certification or seeking job promotion
    - Employees need to understand what to be aware of (e.g. types of threats, such as phishing and free USB sticks), how to perform their jobs securely (e.g. encrypt sensitive data, physically protect valuable assets) and how security plays a role in the big picture (company reputation, profits,and losses)
        - Training should be mandatory and provided both to new employees and yearly (at a minimum) for ongoing training
        - Routine tests of operational security should be performed (such as phishing test campaigns, tailgating at company doors and social engineering tests)
        - **Social engineering**: a form of attack that exploits human nature and behavior; the common social engineering principles are authority, intimidation, consensus, scarcity, familiarity, trust, and urgency;
            - social engineering attacks include phishing, spear phishing, business email compromise (BEC), whaling, smishing, vishing, spam, shoulder surfing, invoice

      scams, hoaxes, impersonation, masquerading, tailgating, piggybacking, dumpster diving, identity fraud, typo squatting, and influence campaigns
      - while many orgs don’t perform social engineering campaigns (testing employees using benign social engineering attempts) as part of security awareness, it is likely to gain traction
      - outside of campaigns, presenting social engineering scenarios and information is a common way to educate
    - Phishing: phishing campaigns are popular, and many orgs use third-party services to routinely test their employees with fake phishing emails
      - such campaigns produce valuable data, such as the percentage of employees who open the phishing email, the percentage who open attachments or click links, and the percentage who report the fake phishing email as malicious
    - Security champions: the term "champion" has been gaining ground; orgs often use it to designate a person on a team who is a subject matter expert in a particular area or responsible for a specific area
      - e.g. somebody on the team could be a monitoring champion — they have deep knowledge around monitoring and evangelize the benefits of monitoring to the team or other teams
      - a security champion is a person responsible for evangelizing security, helping bring security to areas that require attention, and helping the team enhance their skills
    - Gamification: legacy training and education are typically based on reading and then answering multiple-choice questions to prove knowledge; gamification aims to make training and education more fun and engaging by packing educational material into a game
      - gamification has enabled organizations to get more out of the typical employee training

- 1.13.2 Periodic content reviews

  - Threats are complex, so training needs to be relevant and interesting to be effective; this means updating training materials and changing out the ways which security is tested and measured
    - if you always use the same phishing test campaign or send it from the same account on the same day, it isn’t effective, and the same applies to other materials.
    - instead of relying on long/detailed security documentation for training and awareness, consider using internal social media tools, videos and interactive campaigns

- 1.13.2 Program effectiveness evaluation

  - Time and money must be allocated for evaluating the company’s security awareness and training; the company should track key metrics, such as the percentage of employees who click on a fake phishing campaign email links

Also see my articles on risk management:

- Part 1 introduces risk and risk terminology from the lens of the (ISC)² Official Study Guide

- Since the primary goal of risk management is to identify potential threats against an organizaton's assets, and bring those risks into alignment with an organization's risk appetite, in Part2, we cover the threat assessment -- a process of examining and evaluating cyber threat sources with potential system vulnerabilities
    - we look at how a risk assessment helps drive our understanding of risk by pairing assets and their associated potential threats, ranking them by criticality
    - we also discuss quantitative analytic tools to help provide specific numbers for various potential risks, losses, and costs
- In the third installment, we review the outcome of the risk assessment process, looking at total risk, allowing us to determine our response to each risk/threat pair and perform a cost/benefit review of a particular safeguard or control
    - we look at the categories and types of controls and the idea of layering them to provide several different types of protection mechanisms
    - we also review the important step of reporting out our risk analysis and recommended responses, noting differences in requirements for messaging by group

Domain 2 **Asset Security**

- Domain 2 of the CISSP exam covers asset security making up ~10% of the test
- Asset security includes the concepts, principles, and standards of monitoring and securing any asset important to the organization
- The Asset Security domain focuses on collecting, handling, and protecting information throughout its lifecycle; the first step is classifying information based on its value to the organization
- **Anonymization**: replaces privacy data with useful but inaccurate data; the dataset can be shared, but anonymization removes individual identities; anonymization is permanent
- **Asset**: anything of value owned by the organization
- **Asset lifecycle**: phases an asset goes through, from creation (or collection) to destruction
- **EPROM / UVEPROM**: erasable programmable read-only memory, is a type of programmable read-only memory (PROM) chip that retains its data when its power supply is switched off; chips my be erased with ultraviolet light
- **EEPROM**: Electrically Erasable Programmable Read-Only Memory; chips may be erased with electrical current
- **PROM**: programmable read-only memory, a form of digital memory where the contents can be changed once after manufacture of the device
- **RAM**: Random Access Memory - volatile memory that loses contents when the computer is powered off
- **ROM**: nonvolatile memory that can't be written to by end users
- **TEMPEST**: a classification of technology designed to minimize the electromagnetic emanations generated by computing devices; TEMPEST technology makes it difficult, if not impossible, to compromise confidentiality by capturing emanated information; TEMPEST countermeasures to Van Eck phreaking (i.e. eavesdropping), include Faraday cages, white noise, control zones, and shielding

2.1 Identify and classify information assets (OSG-9 Chpt 5)

- 2.1.1 Data classification
    - Managing the data lifecycle refers to protecting it from cradle to grave -- steps need to be taken to protect data when it's first created until it's destroyed

  - One of the first steps in the lifecycle is identifying and classifying information and assets, often within a security policy
  - In this context, assets include sensitive data, the hardware used to process that data, and the media used to store/hold it
  - **Data categorization**: process of grouping sets of data, info or knowledge that have comparable sensativities (e.g. impact or loss rating), and have similar law/contract/compliance security needs
  - **Sensitive data**: any information that isn't public or unclassified, and can include anything an org needs to protect due to its value, or to comply with existing laws and regulations
  - **Personally Identifiable Information (PII)**: any information that can identify an individual
    - more specifically, info about an individual including (1) any info that can be used to distinguish or trace an individual's identity, such as name, social security number, date and place of birth, mother's maiden name, or biometric records; and (2) any other information that is linked or linkable to an individual, such as medical, educational, financial, and employment information (NIST SP 800-122)
  - **Protected Health Information (PHI)**: any health-related information that can be related to a specific person
  - **Proprietary data**: any data that helps an organization maintain a competitive edge
  - Organizations classify data using labels
    - government classification labels include:
      - **Top Secret**: if disclosed, could cause massive damage to national security, such as the disclosure of spy satellite information
      - **Secret**: if disclosed, can adversely affect national security
      - **Unclassified**: not sensitive
    - non-government organizations use labels such as:
      - **Confidential/Proprietary**: only used within the org and, in the case of unauthorized disclosure, it could suffer serious consequences
      - **Private**: may include personal information, such as credit card data and bank accounts; unauthorized disclosure can be disastrous
      - **Sensitive**: needs extraordinary precautions to ensure confidentiality and integrity
      - **Public**: can be viewed by the general public and, therefore, the disclosure of this data would not cause damage
    - labels can be as granular and custom as required by the org
  - It is important to protect data in all states: at rest, in transit, or in use
  - The best way to protect data confidentiality is via use of strong encryption

- 2.1.2 Asset Classification

  - It's important to identify and classify assets, such as systems, mobile devices etc.
  - **Classification**: derived from compliance mandates, the process of recognizing organizational impacts if information suffers any security compromise (whether to confidentiality, integrity, availability, non-repudiation, authenticity, privacy, or safety)
  - Asset classifications should match data classification, i.e. if a computer is processing top secret data, the computer should be classified as a top secret asset
  - **Clearance**: relates to access of certain classfication of data or equipment, and who has access to that level or classification

    - A **formal access approval process** should be used to change user access; the process should involve approval from the data/asset owner, and the user should be informed about rules and limits
        - before a user is granted access they should be educated on working with that level of classification
    - Classification levels can be used by businesses during acquisitions, ensuring only personnel who need to know are involved in the assessment or transition
    - In general, classification labels help users use data and assets properly, for instance by restricting dissemination or use of assets by their classification

2.2 Establish information and asset handling requirements (OSG-9 Chpt 5)

- **Asset handling**: refers to secure transport of media through its lifetime
- The data and asset handling key goal is to prevent data breaches, by using:
    - **Data Maintenance**: on-going efforts to organize and care for data through its life cycle
    - **Data Loss Prevention (DLP)**: systems that detect and block data exfiltration attempts; two primary types:
        - network-based DLP
        - endpoint-based DLP
- **Marking**: (AKA labeling) sensitive information/assets ensures proper handling (both physically and electronically)
- **Data Collection Limitation**: prevent loss by not collecting unnecessary sensitive data
- **Data Location**: keep dup copies of backups, on- and off-site
- **Storage**: define storage locations and procedures by storage type; use physical locks for paper-based media, and encrypt electronic data
- **Destruction**: destroy data no longer needed by the organization; policy should define acceptable destruction methods by type and classification (see NIST SP-800-88 for details)
    - **Erasing**: usually refers to a delete operation on media, leaving data remanence
    - **Clearing**: removal of sensitive data from a storage device such that there is assurance data may not be reconstructed using normal functions or software recovery or software recovery utilities; over-writing existing data; it's not very strong, and there's a chance that the data could be brought back
    - **Purging**: removal of sensitive data from a system or device with the intent that data cannot be reconstructed by any known technique; usually refers to mutliple clearing passes combined with other tools; often means getting rid of data in more reliable ways, like using a strong magnetic field (degaussing) to destroy data on storage devices(see below) -- although not considered acceptable for top secret data
    - **Destruction**: includes physically destroying media through shredding, burning, pulverizing, or incinerating, and also includes the use of strong encryption to logically destroy data; a surer way than even purging
- **Data Remanence**: data remaining on media after typical erasure; to ensure all remanence is removed, the following tools can help:
    - **Degaussing**: used on magentic media, removes data from tapes and magnetic hard drives; no affect on optical media or SSDs
    - **(Physical) destruction**: used for SSD/electronic components, or in combination with other less-secure methods; destruction methods include incineration, crushing, shredding, and disintegration

  - **Cryptographic Erasure**: AKA cryptoshedding, basically destroying encryption key; may be only secure method for cloud storage
  - **File carving**: computer forensics technique that recovers files from a storage device's raw data based on their structure and content, often used to recover files that are not indexed by the file system, such as those that are deleted, formatted, or encrypted; file carving is also a good method for recovering files if an entire directory is missing or corrupt

2.3 Provision resources securely (OSG-9 Chpt 16)

- The primary purpose of security operations practices is to safeguard assets such as information, systems, devices, facilities, and apps; these practices help to identify threats, vulnerabilities, and implement controls to reduce the risk to these asssets

- Implementing common security operations concepts, along with performing periodic security audits and reviews demonstrates a level of due care

- **Need-to-know**: a principle that imposes the requirement to grant users access only to data or resources they need to perform assigned work tasks

- **Least privilege**: a principle stating that subjects are granted only the privileges necessary to perform assigned work tasks and no more

- 2.3.1 Information and asset ownership

  - **Data owner**: the person who has ultimate organizational responsibility for data; usually sr. manager (CEO,president, dept. head); data owners typically delegate data protection tasks to others in the org

- 2.3.2 Asset inventory (e.g., tangible, intangible)

  - **Inventory**: complete list of items
  - **Tangible assets**: include hardware and software assets owned by the company
  - **Intangible assets**: things like patents, copyrights, a company’s reputation, and other assets representing potential revenue
    - an org should keep track of intangible assets, like intellectual property, patents, trademarks, and company’s reputation, and copyrights to protect them
    - note: patents in the US are valid for 20 years

- 2.3.3 Asset management

  - Asset management refers to managing both tangible and intangible assets; this starts with inventories of assets, tracking the assets, and taking additional steps to protect them throughout their lifetime
  - **Accountability**: ensures that account management has assurance that only authorized users are accessing a system and using it properly
  - **Hardware assets**: IT resources such as computers, servers, routers, switches and peripherals
    - use an automated configuration management system (CMS) to help with hardware asset management
    - use barcodes, RFID tags to track hardware assets
  - **Software assets**: operating systems and applications

        - important to monitor license compliance to avoid legal issues
        - software licensing also refers to ensuring that systems do not have unauthorized software installed
    - To protect intangible inventories (like intellectual property, patents, trademarks, and company’s reputation, and copyrights), they need to be tracked

2.4 Manage data lifecycle (OSG-9 Chpt 5)

- 2.4.1 Data roles (i.e., owners, controllers, custodians, processors, users/subjects)
    - **System owner**: controls the computer storing the data; usually includes software and hardware configurations and support services (e.g. cloud implementation)
        - data owner is the person respsonible for classifying, categorizing, and permitting access to the data; the data owner is the person who is best familiar with the importance of the data to the business
        - system owners are responsible for the systems that process the data
        - system owner is responsible for system operation and maintenance, and associated updating/patching as well as related procurement activities
        - per NIST SP 800-18, information system owner has the following responsibilities:
            - develops the system security plan
            - maintains the system security plan and ensures that the system is deployed/operated according to security requirements
            - ensures that system users and support personnel receive the requisite security training
            - updates the system security plan as required
            - assists in the identification, implementation, and assessment of the common security controls
    - **Data controller**: decide what data to process and how to process it
        - the data controller is the person or entity that controls the processing of the data - deciding what data to process, why this data should be processed, and how it is processed
        - e.g. a company that collects personal information on employees for payroll is a data controller (but, if they pass this info to a third-party to process payroll, the payroll company is the data processor, see below)
    - **Data processor**: an entity working on behalf (or the direction) of the data controller, that processes PII; they have a responsibility to protect the privacy of the data and not use it for any purpose other than directed by the data controller; generally, a data processor is any system used to process data
        - a controller can hire a third party to process data, and in this context, the third party is the data processor; data processors are often third-party entities that process data for an org at the direction of the data controller
        - note GDPR definition: "a natural or legal person, public authority, agency, or other body, which processes personal data soley on behalf of the data controller"
            - GDPR also restricts data tranfers to countries outside EU, with fines for violations
            - many orgs have created dedicated roles to oversee GDPR data laws are followed
    - **Data custodian**: a custodian is delegated, from the system owner, day-to-day responsibilities for properly storing and protecting data; responsible for the protection of data through

maintenance activities, backing up and archiving, and preventing the loss or corruption and recovering data
    - **Security administrator**: responsible for ensuring the overall security of entire infrastructure; they perform tasks that lead to the discovery of vulnerabilities, monitor network traffic and configure tools to protect the network (like firewalls and antivirus software)
        - security admins also devise security policies, plans for business continuity and disaster recovery and train staff
    - **Supervisors**: responsible for overseeing the activities of all the above entities and all support personnel; they ensure team activities are conducted smoothly and that personnel is properly skilled for the tasks assigned
    - **Users**: any person who accesses data from a computer device or system to accomplish work (think of users as employees or end users)
        - users should have access to the data they need to perform tasks; users should have access to data according to their roles and their need to access info
        - must comply with rules, mandatory policies, standards and procedures
        - users fall into the category of subjects, and a subject is any entity that accesses an object such as a file or folder
            - note that subjects can be users, programs, processes, services, computers, or anything else that can access a resource (OSG-9 Chpts 8, 13)

- 2.4.2 Data Collection

    - One of the easiest ways of preventing the loss of data is to simply not collect it
    - The **data collection guideline**: if the data doesn't have a clear purpose for use, don't collect it, and don't store it; this is why many privacy regulations mention limiting data collection

- 2.4.3 Data location

    - **Data location**: in this context, refers to the location of data backups or data copies
    - If a company's system is on-prem, keeps data on-site, but regularly backups up data, best practice is to keep a backup copy on site and backup copy off-site
    - Consider distance between data/storage locations to mitigate potential mutual (primary and backup) damage risk

- 2.4.4 Data maintenance

    - **Data maintenance**: managing data through the data lifecycle (creation, usage, retirement); data maintenance is the process (often automated) of making sure the data is available (or not available) based on where it is in the lifecycle
    - Ensuring appropriate asset protection requires that sensitive data be preserved for a period of not less than what is business-required, but for no longer than necessary
    - Encrypt sensitive data
    - Safeguard assets via basic security controls to enforce appropriate levels of confidentiality, integrity and availability and act per security policies, standards, procedures and guidelines

- 2.4.5 Data retention

    - Retention requirements apply to data or records, media holding sensitive data, systems that process sensitive data, and personnel who have access to sensitive data

    - **record retention**: retaining and maintaining info as long as it is needed, and destroying it when its no longer needed
      - note: a current trend in many orgs is to reduce legal liabilities by implementing short retention policies with email
  - Three fundamental retention policy questions:
    - **how to retain**: data should be kept in a manner that makes it accessible whenever required; take taxonomy (or the scheme for data classification) into account
    - **how long to retain data**: general guidelines for business data is 7 years (but can vary by country/region/regulation)
    - **what data**: to retain per org requirements

- 2.4.6 Data remanence

  - **Data remanence**: the data remaining on media after the data is supposedly erased
    - typically refers to data on a hard drive as residual magnetic flux or slack space (unused space within a disk cluster)
      - note that many OSs store files in clusters, which are groups of sectors (the smallest storage unit on a hard disk drive)
    - if media includes any type of private and sensitive data, it is important to eliminate data remanence
    - note that some OSs fill slack space with data from memory, which is why personnel should never process classified data on unclassified systems

- 2.4.7 Data destruction

  - Destroy sensitive data when it is no longer needed
  - An org's security or data policy should define the acceptable methods of destroying data based on the data's classification
  - a degausser can be used on a hard disk drives/magnetic media
  - the best SSD wiping method is destruction -- even when using manufacturers SSD wiping tools, data can remain, and therefore the best SSD wipe method is destruction
  - **Defensible destruction**: eliminating data using a controlled, legally defensible and regulatory compiant way

2.5 Ensure appropriate asset retention (e.g. End-of-Life EOL, End-of-Support (EOS)) (OSG-9 Chpt 5)

- Hardware: even if you maintain data for the appropriate retention period, it won't do you any good if you don't have hardware that can read the data

- Personnel: beyond retaining data for required time periods and maintaining hardware to read the data, you need personnel who know how to operate the hardware to execute restoraton processes

- **End-Of-Life (EOL)**: often identified by vendors as the time when they stop offering a product for sale

- **End-Of-Support (EOS)/End-Of-Service-Life (EOSL)**: often used to identify when support ends for a product

- EOL,EOS/EOSL can apply to either software or hardware

2.6 Determine data security controls and compliance requirements (OSG-9 Chpt 5)

- You need security controls that protect data in each possible state: at rest, in transit or in use
- Each state requires a different approach to security; note that there aren’t as many security options for data in use as there are for data at rest or data in transit
    - keeping the systems patched, maintaining a standard computer build process, and running anti-virus/malware are typically the real-world primary protections for data in use
- 2.6.1 Data states (e.g., in use, in transit, at rest)
    - The three data states are at rest, in transit, and in use
        - **Data at rest**: any data stored on media such as hard drives or external media
        - **Data in transit**: any data transmitted over a network
            - encryption methods protect data at rest and in transit
        - **Data in use**: data in memory and used by an application
            - applications should flush memory buffers to remove data after it is no longer needed
- 2.6.2 Scoping and tailoring
    - **Baseline**: documented, lowest level of security config allowed by a standard or org
    - After selecting a control baseline, orgs fine-tune with tailoring and scoping processes; a big part of the tailoring process is aligning controls with an org's specific security requirements
    - **Tailoring**: refers to modifying the list of security controls within a baseline to align with the org's mission
        - includes the following activities:
            - identifying and designating common controls; specificaion of organization-defined parameters in the security controls via explicit assignment and selection statements
            - applying scoping guidance/considerations
            - selecting/specifying compensating controls
            - assigning control values
    - **Scoping**: setting the boundaries of security control implementation; limiting the general baseline recommendations by removing those that do not apply; part of the tailoring process and refers to reviewing a list of baseline security controls and selecting only those controls that apply to the systems you're trying to protect
        - scoping processes eliminate controls that are recommended in a baseline
- 2.6.3 Standards selection
    - Organizations need to identify the standards (e.g. PCI DSS, GDPR etc) that apply and ensure that the security controls they select fully comply with these standards
    - Even if the org doesn't have to comply with a specific standard, using a well-designed community standard can be helpful (e.g. NIST SP 800 documents)
    - **Standards selection**: the process by which organizations plan, choose and document technologies or architectures for implementation
        - e.g. you evaluate three vendors for a security control; you could use a standards selection process to help determine which solution best fits the org

  - Vendor selection is closely related to standards selection but focuses on the vendors, not the technologies or solutions
  - The overall goal is to have an objective and measurable selection process
    - if you repeat the process with a totally different team, the alternate team should come up with the same selection

- 2.6.4 Data protection methods (e.g., Digital Rights Management (DRM), Data Loss Prevention (DLP), Cloud Access Security Broker (CASB))

  - **Data protection methods** include:
    - **digital rights management (DRM)**: methods used in attempt to protect copyrighted materials; purpose is to prevent the unauthorized use, modification, and distirbution of copyrighted works
    - **Cloud Access Security Brokers (CASBs)**: software placed logically between users and cloud-based resources ensuring that cloud resources have the same protections as resources within a network
      - note that entities must comply with the EU GDPR, and use additional data protection methods such as pseudonymization, tokenization, and anonymization
  - One of the primary methods of protecting the confidentiality of data is encryption
  - Options for protecting your data vary depending on its state:
    - Data at rest: consider encryption for operating system volumes and data volumes, and backups as well
      - be sure to consider all locations for data at rest, such as tapes, USB drives, external drives, RAID arrays, SAN, NAS, and optical media
      - DRM is useful for data at rest because DRM "travels with the data" regardless of the data state
        - DRM is especially useful when you can’t encrypt data volumes
    - Data in transit: think of data in transit wholistically -- moving data from anywhere to anywhere; use encryption for data in transit
      - e.g. a web server uses a certificate to encrypt data being viewed by a user, or IPsec encrypting a communication session
      - most important point is to use encryption whenever possible, including for internal-only web apps
      - DLP solutions are useful for data in transit, scanning data on the wire, and stopping the transmission/transfer, based on the DLP rules set (e.g. outbound data that contains numbers matching a social security number pattern, a DLP rule can be used to block that traffic)
    - Data in use:
      - CASB solution often combines DLP, a web application firewall with some type of authentication and authorization, and a network firewall in a single solution; A CASB solution is helpful for protecting data in use (and data in transit)
  - **Pseudonymization**: refers to the process of using pseudonyms to represent other data; process of replacing data elements with pseudonyms or aliases
    - A pseudonym is an alias, and pseudonymization can prevent data from directly identifying an entity (i.e. person)
  - **Tokenization**: use of a token, typically a random string of characters, to replace other data

    - note that tokenization is similar to pseudonymization in that they are both used to represent other data, and the token or pseudonym have no meaning or value outside the process that creates and links them to that data
  - example of tokenization used in CC transactions:
    - registration: app on user's smart phone securely sends CC info to the credit card processor (CCP)
      - The CCP sends the CC info to a tokenization vault, creating a token and associating it with the user's phone
    - usage: when the user makes a purchase, the POS system sends the token to the CCP for authorization
    - validation: the CCP sends the token to the tokenization vault; the vault replies with the CC info, the charge is processed
    - completing the sale: the CCP sends a reply to the POS indicating the charge is approved
    - this system prevents CC theft at the POS system

Domain 3 **Security Architecture and Engineering**

You may find this domain to be more technical than others, and if you have experience woring in a security engineering role you likely have an advantage. If not, allocate extra time to this domain to ensure you have a good understanding of the topics

- **Advanced Encryption Standard (AES)**: uses the Rijndael algorithm and is the US gov standard for the secure exchange of sensitive but unclassified data; AES uses key lengths of 128, 192, and 256 bits, and a fixed block size of 128 bits, achieving a higher level of security than the older DES algorithm
- **Algorithm**: a mathmatical function that is used in the encryption and decryption process; can be simply or very complex; also defined as a set of instructions by which encryption and decryption is done
- **ASLR**: Address space layout randomization (ASLR) is a memory-protection process for operating systems (OSes) that guards against buffer-overflow attacks by randomizing the location where system executables are loaded into memory
- **Block Mode Encryption**: using fixed-length sequences of input plaintext symbols as the unit of encryption
- **Block ciphers**: take a number of bits and encrypt them in a single unit, padding the plaintext to achieve a multiple of the block size; the Advanced Encryption Standard (AES) algorithm uses 128-bit blocks
- **Ciphers**: always meant to hide the true meaning of a message; types of ciphers include transposition, substitution, stream, and block
- **Ciphertext**: altered form of a plaintext message so as to be unreadable for anyone expect the intended recipients (it's a secret)
- **Cleartext**: any information that is unencrypted, although it might be in an encoded form that is not easily human-readable (such as base64 encoding)
- **Codes**: cryptographic systems of symbols that operate on words or phrases and are sometimes secret, but don't always provide confidentiality
- **Collision**: occurs when a hash function generates the same output for different inputs
- **Cryptanalysis**: study of techniques for attempting to defeat cryptographic techniques and generally information security services; Cryptanalysis is the process of transforming or decoding

communications from non-readable to readable format without having access to the real key

- **Cryptographic Hash function**: process or function that transforms an input plaintext into a unique value called a hash (or hash value); note that they do not use cryptographic algorithms, as hashes are one-way functions where it's infeasible to determine the plaintext; Message digests are an example of cryptographic hash
- **Cryptography**: study of/application of methods to secure the meaning and content of messages, files etc by disguise, obscuration, or other transformations
- **Cryptosystem**: complete set of hardware, software, communications elements and procedures that allow parties to communicate, store or use info protected by cryptographic means; includes algroithm, key, and key management functions
- **Cryptovariables(s)**: parameters associated with a particular cryptogrphic algorithm; e.g. block size, key length and number of iterations
- **Cyber-physical systems**: systems that use 'computational means' to control physical devices
- **Decoding**: the reverse process from encoding, converting the encoded message back to plaintext format
- **Decryption**: the reverse process from encryption
- **Elliptic-curve cryptography (ECC)**: a newer mainstream algorithm, is normally 256 bits in length (a 256-bit ECC key is equivalent to a 3072-bit RSA key), making it securer and able to offer stronger anti-attack capabilities
- **Encoding**: action of changing a message or set of info into another format through the use of code; unlike encryption, encoded info can still be read by anyone with knowledge of the encoding process
- **Encryption**: process and act of converting the message from plaintext to ciphertext (AKA enciphering)
- **Fog computing**: advanced computational architecture often used as an element in IIoT; fog computing relies on sensors, IoT devices, or edge computing devices to collect data, then transfers it back to a central location for processing (centralizing processing and intelligence)
- **Frequency analysis**: form of cryptanalysis that uses frequency of occurrence of letters, words or symbols in the ciphertext as a way of reducing the search space
- **Hybrid encryption system**: a system that uses both symmetric and asymmetric encryption
- **Key**: the input that controls the operation of the cryptographic algorthm, determining the behavior of the algorithm and permits the reliable encyrption and decryption of the message
- **Key pair**: matching set of one public and one private key
- **Key escrow**: process by which keys (asymmetric or symmetric) are placed in a trusted storage agent's custody, for later retrieval
- **Key generation**: the process of creating a new encryption/decryption key
- **Key recovery**: process of reconstructing an encryption key from the cyphertext alone; if there is a workable key recovery system, it means the algorithm is not secure
- **Key space**: represents the total number of possible values of keys in a cryptographic algorithm or password; keyspace = 2 to the power of the number of bits, so 4 bits = 16 keys, 8 bits = 256 keys
- **Meet-in-the-middle**: attack that uses a known plaintext message and both encryption of the plaintext and decryption of the ciphertext simultaneously in a brute-force manner to identify the encryption key; 2DES is vulnerable to this attack
- **Microcontroller**: similar to system on a chip (SoC), consists of a CPU, memory, IO devices, and non-volatile storage (e.g. flash or ROM/PROM/EEPROM); think Raspberry Pi or Arduino
- Mobile device deployment models that cover allowing or providing mobile devices for employees include: BYOD, COPE, CYOD, and COMS/COBO; also consider VDI and VMI options;

- Mobile device deployment policies should address things like data ownership, support ownership, patch and update management, security product management, forensics, privacy, on/offboarding, adherence to corporate policies, user acceptance, legal concerns, acceptable use policies, camera/video, microphone, Wi-Fi Direct, tethering and hotspots, contactless payment methods, and infrastructure considerations
- **Multistate systems**: certified to handle data from different security classifications simultaneously
- **One-time pad**: series of randomly generated symmetric encryption keys, each one to be used only once by the sender and recipient; to be successful, the key must be generated randomly without any known pattern; the key must be at least as long as the message to be encrypted; the pads must be protected against physical disclosure and each pad must be used only one time, then discarded
- **Out-of-band**: transmitting or sharing control information (e.g. encryption keys and crypto variables) by means of a separate and distinct communications path, channel, or system
- **Personal electronic device (PED)** security features can usually be managed using mobile device management (MDM) or unified endpoint management (UEM) solutions, including device authentication, full-device encryption, communication protection, remote wiping, communication protection, device lockout, screen locks, GPS and location services, content management, app control, push notification management, third-party app store control, rooting/jailbreaking, credential management and more
- **Plaintext**: message or data in its readable form, not turned into a secret
- **RTOS**: real-time operating system (RTOS) is an operating system specifically designed to manage hardware resources and run applications with precise timing and high reliability; they are designed to process data with minimum latency; an RTOS is often stored on ROM; they use deterministic timing, meaning tasks are completed within a defined time frame and is designed to operate in a hard (i.e. missing a deadline can cause system failure) or soft (missing a deadline degrades performance but is not catastrophic) real-tme condition
- **Salting vs key stretching**: salting adds randomness and uniqueness to each password before hashing, which reduces the effectiveness of rainbow table attacks; key stretching makes the hashing process deliberately slow, making it much more challenging for attackers to crack passwords using brute-force or precomputed tables; common password hashing algorithms that use key stretching include PBKDF2, bcrypt, and scrypt
- **SDx**: software-defined everything refers to replacing hardware with software using virtualization; includes virtualization, virtualized software, virtual networking, containerization, serverless architecture, IaC, SDN, VSAN, software-defined storage (SDS), VDI, VMI SDV, and software-defined data center (SDDC)
- **Session key**: a symmetric encryption key generated for one-time use; usually requires a key encapsulation approach to eliminate key management issues
- **Static Environments**: apps, OSs, hardware, or networks that are created/configured to meet a particular need or function are set to remain unaltered; static environments, embedded systems, network-enabled devices, edge, fog, and mobile devices need security management that may include network segmentation, security layers, app firewalls, manual updates, firmware version control, wrappers, and control redundancy/diversity
- **Stream mode encryption**: system using a process that treats the input plaintext as a continuous flow of symbols, encrypting one symbol at a time; usually uses a streaming key, using part of the key as a one-time key for each symbol's encryption
- **Stream ciphers**: encrypt the digits (typically bytes), or letters (in substitution ciphers) of a message one at a time

- **Substitution cipher**: encryption/decription process using subsitution
- **Symmetric encryption**: process that uses the same key (or a simple transformation of it) for both encryption/decryption
- **Transposition cypher**: encryption/decription process using transposition
- **Trust and Assurance**: trust is the presence of a security mechanism or capability; assurance is how reliable the security mechanism(s) are at providing security
- **VESDA**: very early smoke detection process (air sensing device brand name)
- **Work factor**: (AKA Work function) is a way to measure the strength of a cryptography system, measuring the effort in terms of cost/time to decrypt messages; amount of effort necessary to break a cryptographic system using a bruteforce attack, measured in elapsed time
- **Zero-knowledge proof**: one person demonstrates to another that they can achieve a reslut that requires sensitive info without actually disclosing the sensitive info

3.1 Research, implement, and manage engineering processes using secure design principles (OSG-9 Chpts 1,8,9,16)

- 3.1.1 Threat Modeling
  - **Threat modeling**: a security process where potential threats are identified, categorized, and analyzed; it can be performed as a proactive measure during design and development or as an reactive measure once a product has been deployed
    - Threat modeling identifies the potential harm, the probability of occurrence, the priority of concern, and the means to eradicate or reduce the threat
    - Threat modeling commonly involves decomposing the app to understand it and how it interacts with other components or users; idnetifying and ranking threats allows potential threats to be propritized; dentifying how to mitigate those threats finishes the process
- 3.1.2 Least Privilege
  - **Least privilege**: states that subjects are granted only the privileges necessary to perform assigned work tasks and no more; this concept extends to data and systems
    - Limiting and controlling privileges based on this concept protects confidentiality and data integrity
- 3.1.3 Defense in Depth
  - **Defense in Depth**: AKA layering, is the use of multiple controls in a series, where a single failed control should not result in exposure of systems or data; layers should be used in a series (one after the other), NOT in parallel
    - When you see the terms like levels, multilevel, layers, classifications, zones, realms, compartments, protection rings etc think about Defense in Depth
- 3.1.4 Secure defaults
  - **Secure defaults**: when you think about defaults, consider how something operates brand new, just turned over to you by the vendor
    - e.g. wireless router default admin password, or firewall configuration requiring changes to meet an organization's needs
- 3.1.5 Fail securely
  - **Fail securely**: if a system, asset, or process fails, it shouldn't reveal sensitive information, or be less secure than during normal operation; failing securely could involve reverting to defaults
- 3.1.6 Separation of duties (SoD)

  - **Separation of duties (SoD)**: separation of duties (SoD) and responsibilities ensures that no single person has total control over a critical function or system; SoD is a process to minimize opportunities for misuse of data or environment damage
    - e.g. one person sells tickets, another collects tickets and restricts access to ticket holders in a movie theater
- 3.1.7 Keep it simple
  - **Keep it simple**: AKA keep it simple, stupid (KISS), this concept is the encouragement to avoid overcomplicating the environment, organization, or product design 3.1.8 Zero Trust
  - **Zero Trust**: "assume breach"; a security concept and alternative of the traditional (castle/moat) approach where nothing is automatically trusted; instead each request for activity or access is assumed to be from an unknown and untrusted location until otherwise verified
    - "Never trust, always verify" replaces "trust but verify" as a security design principle by asserting that all activities by all users/entities must be subject to control, authentication, authorization, and management at the most granular level possible
    - Goal is to have every access request authenticated, authorized, and encrypted prior to access being granted to an asset or resource
    - See my article on an Overview of Zero Trust Basics
- 3.1.9 Privacy by design
  - **Privacy by design (PbD)**: a guideline to integrate privacy protections into products during the earliest design phase rather than tacking it on at the end of development
    - Same overall concept as "security by design" or "integrated security" where security is an element of design and architecture of a product starting at initiation and continuing through the software development lifecycle (SDLC)
    - There are 7 recognized principles to achieve privacy by design:
      - Proactive, preventative: think ahead and design for things that you anticipate might happen
      - Default setting: make private by default, e.g. social media app shouldn't share user data with everybody by default
      - Embedded: build privacy in; don’t add it later
      - Full functionality, positive-sum: achieve both security and privacy, not just one or the other
      - Full lifecycle protection: privacy should be achieved before, during and after a transaction; part of this is securely disposing of data when it is no longer needed
      - Visibility, transparency, open: publish the requirements and goals; audit them and publish the findings
      - Respect, user-centric: involve end users, providing the right amount of information for them to make informed decisions about their data
- 3.1.10 Trust but verify
  - **Trust but verify**: based on a Russian proverb, and no longer sufficient; it's the traditional approach of trusting subjects and devices within a company's security perimeter automatically, leaving an org vulnerable to insider attacks and providing intruders the ability to easily perform lateral movement
- 3.1.11 Shared responsibility
  - **Shared responsibility**: the security design principle that indicates that organizations do not operate in isolation

        - Everyone in an organization has some level of security responsibility
        - The job of the CISO and security team is to establish & maintain security
        - The job of regular employees to perform their tasks within the confines of security
        - The job of the auditor is to monitor the environment for violations
        - Because we participate in shared responsibility we must research, implement, and manage engineering processes using secure design principles
        - When working with third parties, especially with cloud providers, each entity needs to understand their portion of the shared responsibility of performing work operations and maintaining security; this is often referenced as the **cloud shared responsibility model**

3.2 Understand the fundamental concepts of security models (e.g. Biba, Star Model, Bell-LaPadula) (OSG-9 Chpt 8)

- Security models:
    - Intended to provide an explicit set of rules that a computer can follow to implement the fundamental security concepts, processes, and procedures of a security policy
    - Provide a way for a designer to map abstract statements into a security policy prescribing the algorithms and data structures necessary to build hardware and software
    - Enable people to access only the data classified for their clearance level
- **State machine model**: ensures that all instances of subjects accessing objects are secure
- **Information flow model**: designed to prevent unauthorized, insecure, or restricted information flow; the Information Flow model is an extension of the state machine concept and serves as the basis of design for both the Biba and Bell-LaPadula models
- **Noninterference model**: prevents the actions of one subject from affecting the system state or actions of another subject
- **Bell-LaPadula**: Model was established in 1973; the goal is to ensure that information is exposed only to those with the right level of classification
    - Focus is on *confidentiality*
    - **Simple property**: "No read up"
    - **Star (*) property**: "No write down" (AKA confinement property)
    - Discretionary Security Property: uses an access matrix (need to know in order to access)
    - Doesn't address covert channels
- **Biba**: Released in 1977, this model was created to supplement Bell-LaPadula
    - Focus is on *integrity*
    - **Simple Integrity Property**:"No read down" (for example, users with a Top Secret clearance can’t read data classified as Secret)
    - **Star (*) Integrity Property**: "No write up" (for example, a user with a Secret clearance can’t write data to files classified as Top Secret)
    - By combining it with Bell-LaPadula, you get both confidentiality and integrity
    - Biba uses a lattice to control access and is a form of mandatory access control (MAC) model
- **Take-Grant**:

  - Take-grant is a confidentiality-based model that supports four basic operations: take, grant, create, and revoke; it employs a directed graph to dictate how rights can be passed from one subject to another, or from a subject to an object
  - **Take rule**: allows a subject to take rights over an object
  - **Grant rule**: allows a subject to grant rights to an object
  - **Create rule**: allows a subject to create new rights
  - **Remove rule**: allows a subject to remove rights it has

- **Clark-Wilson**:

  - Designed to protect integrity using the access control triplet (subject/program/object)
  - A program interface is used to limit what is done by a subject; if the focus of an intermediary program between subject and object is to protect integrity, then it is an implementation of the Clark-Wilson model
  - Uses security labels to grant access to objects via transformation procedures and a restricted interface model
  - The Clark-Wilson Model enforces the concept of separation of duties
  - Three parts of the Clark-Wilson model are: subject, object, and program (or interface)
  - Clark-Wilson components:
    - **Users**: (AKA active agents) the subjects which will access the objects
    - **Transformation Procedures (TPs)**: operations the subject is trying to perform (read, write, modify)
    - **Constrained Data Items (CDIs)**: objects at a higher-level of protection; CDIs can only be manipulated by a TP
    - **Unconstrained Data Items (UDIs)**: can be accessed directly by the subject, and do not need to go through a intermediary like a TP
    - **Integrity Verification Procedures (IVPs)**: a way to audit the TP, checking for internal and external consistency

- **Brewer and Nash Model**:

  - AKA "ethical wall", and "cone of silence"
  - created to permit access controls to change dynamically based on a user's previous activity

- **Goguen-Meseguer Model**:

  - An integrity model
  - Foundation of noninterference conceptual theories

- **Sutherland Model**:

  - Focuses on preventing interference in support of integrity

- **Graham-Denning Model**

  - Focused on the secure creation and deletion of both subjects and objects
  - 8 primary protection rules or actions
    - 1-4:securely create/delete a subject/object
    - 5-8:securely provide the read/grant/delete/transfer access right

- **Harrison-Ruzzo-Ullman Model**:
  - Focuses on the assignment of object access rights to subjects as well as the resilience of those assigned rights
  - HRU is an extension of Graham-Denning model
- **Star Model**:
  - Not an official model, but name refers to using asterisks (stars) to dictate whether a person at a specific level of confidentiality is allowed to write data to a lower level of confidentiality
  - Also determines whether a person can read or write to a higher or lower level of confidentiality

3.3 Select controls based upon systems security requirements (OSG-9 Chpt 8)

- Be familiar with the **Common Criteria (CC)** for Information Technology Security Evaluation
- Based on ISO/IEC 15408 is a subjective security function evaluation tool that uses protection profiles (PPs) and security targets (Sts) and assigns an Evaluation Assurance level (EAL)
- The CC provides a standard to evaluate systems, defining various levels of testing and confirmation of systems' security capabilities
- The number of the level indicates what kind of testing and confirmation has been performed
- The important concepts:
  - To perform an evaluation, you need to select the **Target of Evaluation (TOE)** (e.g. firewall or an anti-malware app)
  - The evaluation process will look at the **protection profile (PP)**, which is a document that outlines the security needs (customer "I wants"); a vendor might use a specific protection profile for a particular solution
  - The evaluation process will look at the **Security Target (ST)**, specifying the claims of security from the vendor that are built into a TOE (the ST is usually published to customers and partners and available to internal staff)
  - An organization's PP is compared to various STs from the selected vendor's TOEs, and the closest or best match is what the org purchases
  - The evaluation will attempt to gauge the confidence level of a security feature
  - **Security assurance requirements (SARs)**: a description of how the TOE is to be evaluated, based on the development of the solution
  - Key actions during development and testing should be captured
  - An **evaluation assurance level (EAL)**: a numerical rating used to assess the rigor of an evaluation; the scale is EAL 1 (cheap and easy) to EAL7 (expensive and complex):
    - 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, designed, and tested
    - EAL7: formally verified, designed, and tested
- **Authorization to Operate (ATO)**: official auth to use specific IT systems to perform tasks/accept identified risks

3.4 Understand security capabilities of Information Systems (IS) (e.g. memory protection, Trusted Platform Model (TPM), encryption/decryption) (OSG-9 Chpt 8)

- Security capabilities of information systems include memory protection, virtualization, Trusted Platform Module (TPM), encryption/decryption, interfaces, and fault tolerance

- A computing device is likely running multiple apps and services simultaneously, each occupying a segment of memory; the goal of memory protection is to prevent one app or service from impacting another

- There are two primary memory protection methods:

    - **Process isolation**: OS provides separate memory spaces for each processes instructions and data, and prevents one process from impacting another
    - **Hardware segmentation**: forces separation via physical hardware controls rather than logical processes; in this type of segmentation, the operating system maps processes to dedicated memory locations

- **Virtualization**: technology used to host one or more operating systems within the memory of a single host, or to run applications that are not compatible with the host OS; the goal is to protect the hypervisor and ensure that compromising one VM doesn't affect others on that host

- **Virtual Software**: software that is deployed in a way that acts as if it is interacting with a full host OS; virutalized app is isolated from the host OS so it cna't make direct/permanent changes to the host OS

- **Trusted Platform Module (TPM)**: a cryptographic chip that is sometimes included with a client computer or server; a TPM enhances the capabilities of a computer by offering hardware-based cryptographic operations

    - TPM is a tamper-resistant integrated circuit built into some motherboards that can perform cryptographic operations (including key gen) and protect small amoutns of sensitive info, like passwords and cryptographic keys
    - Many security products and encryption solutions require a TPM
    - TPM is both a specification for a cryptoprocessor chip on a motherboard and the general name for implementation of the specification
    - A TPM is an example of a **hardware security module (HSM)**: a cryptoprocessor used to manage and store digital encryption keys, accelerate crypto operations, support faster digital signatures, and improve authentication

- User interface: a constrained UI can be used in an application to restrict what users can do or see based on their privileges

    - e.g. dimming/graying out capabilities for users without the correct privilege
    - An interface is also the method by which two or more systems communicate

- Be aware of the common security capabilities of interfaces:

    - Encryption/decryption: when communications are encrypted, a client and server can communicate without exposing information to the network; when an interface doesn’t provide such a capability, use IPsec or another encrypted transport mechanism

  - Signing: used for non-repudiation; in a high-security environment, both encrypt and sign all communications if possible

- **Fault tolerance**: capability used to enhance availability; in the event of an attack (e.g. DoS), or system failure, fault tolerance helps keep a system up and running

3.5 Assess and mitigate the vulnerabilities of security architectures, designs and solution elements (OSG-9 Chpts 9,16,20)

This objective relates to identifying vulnerabilities and corresponding mitigating contols and solutions; the key is understanding the types of vulnerabilities commonly present in different environments, and their mitigation options

- 3.5.1 Client-based systems
  - **Client-based systems**: client computers are the most attacked entry point
  - Compromised client computers can be used to launch other attacks
  - Productivity software and browsers are constant targets
  - Even patched client computers are at risk due to phishing and social engineering vectors
  - Mitigation: run a full suite of security software, including anti-virus/malware, anti-spyware, and host-based firewall

- 3.5.2 Server-based systems
  - **Data Flow Control**: movement of data between processes, between devices, across a network, or over a communications channel
  - Management of data flow seeks to minimize latency/delays, keep traffic confidential (i.e. using encryption), not overload traffic (i.e. load balancer), and can be provided by network devices/applications and services
  - While attackers may initially target client computers, servers are often the goal
  - **Mitigation**: regular patching, deploying hardened server OS images for builds, and use host-based firewalls

- 3.5.3 Database systems
  - Databases often store a company's most sensitive data (e.g. proprietary, CC info, PHI, and PII)
  - Database general ACID properties (Atomicity, Consistency, Isolation and Durability):
    - Atomicity: transactions are all-or-nothing; a transaction must be an atomic unit of work, i.e., all of its data modifications are performed, or none are performed
    - Consistency: transactions must leave the database in a consistent state
    - Isolation: transactions are processed independently
    - Durability: once a transaction is committed, it is permanently recorded
  - Attackers may use inference or aggregation to obtain confidential information
  - **Aggregation attack**: process where SQL provides a number of functions that combine records from one or more tables to produce potentially useful info
  - **Inference attack**: involves combining several pieces of nonsensitive info to gain access to that which should be classified at a higher level; inference makes use of the human mind’s deductive capacity rather than the raw mathematical ability of database platforms

- 3.5.4 Cryptographic systems

    - Goal of a well-implemented cryptographic system is to make compromise too time-consuming and/or expensive
    - Each component has vulnerabilities:
        - **Kerckhoff's Principle** (AKA Kerckhoff's assumption): a cryptographic system should be secure even if everything about the system, except the key, is public knowledge
        - Software: used to encrypt/decrypt data; can be a standalone app, command-line, built into the OS or called via API; like any software, there are likely bugs/issues, so regular patching is important
        - Keys: dictate how encryption is applied through an algorithm; a key should remain secret, otherwise the security of the encrypted data is at risk
            - **key space**: represents all possible permutations of a key
            - key space best practices:
                - key length is an important consideration; use as long of a key as possible (your goal is to outpace projected increase in cryptanalytic capability during the time the data must be kept safe); longer keys discourage brute-force attacks
                    - a 256-bit key is typically minimum recommendation for symmetric encryption
                    - 2048-bit key typically the minimum for asymmetric
                - always store secret keys securely, and if you must transmit them over a network, do so in a manner that protects them from unauthorized disclosure
                - select the key using an approach that has as much randomness as possible, taking advantage of the entire key space
                - destroy keys securely, when no longer needed
            - always base key length on requirements and sensitivity of the data being handled
        - Algorithms: choose algorithms (or ciphers) with a large key space and a large random **key value** (key value is used by an algorithm for the encryption process)
            - algorithms themselves are not secret; they have extensive public details about history and how they function

- 3.5.5 Industrial Control Systems (ICS)

    - **Industrial control systems (ICS)**: a form of computer-management device that controls industrial processes and machines, also known as operational technology (OT)
    - **Supervisory control and data acquisition (SCADA)**: systems used to control physical devices like those in an electrical power plant or factory; SCADA systems are well suited for distributed environments, such as those spanning continents
        - some SCADA systems still rely on legacy or proprietary communications, putting them at risk, especially as attackers gain knowledge of such systems and their vulnerabilities
        - SCADA risk mitigations:
            - isolate networks
            - limit access physically and logically
            - restrict code to only essential apps
            - log all activity

- 3.5.6 Cloud-based systems (e.g., Software as a Service (SaaS), Infrastructure as a Service (IaaS), Platform as a Service (PaaS))

  - **Cloud-based systems**: on-demand access to computing resources available from almost anywhere
  - Cloud's primary challenge: resources are outside the org's direct control, making it more difficult to manage risk
  - Orgs should formally define requirements to store and process data stored in the cloud
  - Focus your efforts on areas that you can control, such as the network entry and exit points (i.e. firewalls and similar security solutions)
  - All sensitive data should be encrypted, both for network communication and data-at-rest
  - Use centralized identity access and management system, with multifactor authentication
  - Customers shouldn't use encryption controlled by the vendor, eliminating risks to vendor-based insider threats, and supporting destruction using
  - **Cryptographic erase**: methods that permanently remove the cryptographic keys
  - Capture diagnostic and security data from cloud-based systems and store in your SIEM system
  - Ensure cloud configuration matches or exceeds your on-premise security requirements
  - Understand the cloud vendor's security strategy
  - Cloud shared responsibility by model:
    - Software as a Service (SaaS):
      - the vendor is responsible for all maintenance of the SaaS services
    - Platform as a Service (PaaS):
      - customers deploy apps that they've created or acquired, manage their apps, and modify config settings on the host
      - the vendor is responsible for maintenance of the host and the underlying cloud infrastructure
    - Infrastructure as a Service (IaaS):
      - IaaS models provide basic computing resources to customers
      - customers install OSs and apps and perform required maintenance
      - the vendor maintains cloud-based infra, ensuring that customers have access to leased systems

- 3.5.7 Distributed systems

  - **Distributed computing environment (DCE)**: a collection of individual systems that work together to support a resource or provide a service
  - DCEs are designed to support communication and coordination among their members in order to achieve a common function, goal, or operation
  - Most DCEs have duplicate or concurrent components, are asynchronous, and allow for fail-soft or independent failure of components
  - DCE is AKA concurrent computing, parallel computing, and distributed computing
  - DCE solutions are implemented as client-server, three-tier, multi-tier, and peer-to-peer
  - Securing distributed systems:
    - in distributed systems, integrity is sometimes a concern because data and software are spread across various systems, often in different locations
    - Client/server model network is AKA a distributed system or distributed architecture
      - security must be addressed everywhere instead of at a single centralized host
      - processing and storage are distributed on multiple clients and servers, and all must be secured
      - network links must be secured and protected

- 3.5.8 Internet of Things (IoT)
    - **Internet of things (IoT)**: a class of smart devices that are internet-connected in order to provide automation, remote control, or AI processing to appliances or devices
        - An IoT device is almost always separate/distinct hardware used on its own or in conjunction with an existing system
        - IoT security concerns often relate to access and encryption
        - IoT is often not designed with security as a core concept, resulting in security breaches; once an attacker has remote access to the device they may be able to pivot
        - Securing IoT:
            - deploy a distinct network for IoT equipment, kept separate and isolated (known as **three dumb routers**)
            - keep systems patched
            - limit physical and logical access
            - monitor activity
            - implement firewalls and filtering
            - never assume IoT defaults are good enough, evaluate settings and config options, and make changes to optimize security while supporting business function
            - disable remote management and enable secure communication only (such as over HTTPS)
            - review IoT vendor to understand their history with reported vulnerabilities, response time to vulnerabilities and their overall approach to security
            - not all IoT devices are suitable for enterprise networks
- 3.5.9 Microservices
    - **Service-oriented Architecture (SOA)**: constructs new apps or functions out of existing but separate and distinct software services, and the resulting app is often new; therefore its security issues are unknown, untested, and unprotected; a derivative os SOA is microservices
    - **Microservices**: a feature of web-based solutions and derivative of SOA
        - A microservice is simply one element, feature, capability, business logic, or function of a web app that can be called upon or used by other web apps
        - Microservices are usually small and focused on a single operation, engineered with few dependencies, and based on fast, short-term development cycles (similar to Agile)
        - Securing microservices:
            - use HTTPS only
            - encrypt everything possible and use routine scanning
            - closely aligned with microservices is the concept of shifting left, or addressing security earlier in the SDLC; also integrating it into the CI/CD pipeline
            - consider the software supplychain or dependencies of libries used, when addressing updates and patching
- 3.5.10 Containerization
    - **Containerization**: AKA OS virtualization, is based on the concept of eliminating the duplication of OS elements in a virtual machine; instead each app is placed into a container that includes only the actual resources needed to support the app, and the common or shared OS elements are used from the hypervisor

        - Containerization is able to provide 10 to 100 x more application density per physical server compared to traditional virtualization
        - Vendors often have security benchmarks and hardening guidelines to follow to enhance container security
        - Securing containers:
            - container challenges include the lack of isolation compared to a traditional infrastructure of physical servers and VMs
            - scan container images to reveal software with vulns
            - secure your registries: use access controls to limit who can publish images, or even access the registry
            - require images to be signed
            - harden container deployment including the OS of the underlying host, using firewalls, and VPC rules, and use limited access accounts
            - reduce the attack surface by minimizing the number of components in each container, and update and scan them frequently

- 3.5.11 Serverless

    - **Serverless architecture** (AKA **function as a service (FaaS)**): cloud computing where code is managed by the customer and the platform (i.e. supporting hardware and software) or servers are managed by the CSP
        - Applications developed on serverless architecture are similar to microservices, and each function is created to operate independently and automonomously
        - A serverless model, as in other CSP models, is a shared security model,and your org and the CSP share security responsibility

- 3.5.12 Embedded systems

    - **Embedded systems**: any form of computing component added to an existing mechanical or electrical system for the purpose of providing automation, remote control, and/or monitoring; usually including a limited set of specific functions
        - Embedded systems can be a security risk because they are generally static, with admins having no way to update or address security vulns (or vendors are slow to patch)
        - Embedded systems focus on minimizing cost and extraneous features
        - Embedded systems are often in control of/associated with physical systems, and can have real-world impact
        - Securing embedded systems:
            - embedded systems should be isolated from the internet, and from a private production network to minimize exposure to remote exploitation, remote control, and malware
            - use secure boot feature and physically protecting the hardware

- 3.5.13 High-Performance Computing (HPC) systems

    - **High-performance computing (HPC)** systems: platforms designed to perform complex calculations/data manipulation at extremely high speeds (e.g. super computers or MPP (Massively Parallel Processing)); often used by large orgs, universities, or gov agencies
        - An HPC solution is composed of three main elements:

            - compute resources
            - network capabilities
            - storage capacity
        - HPCs often implement real-time OS (RTOS)
        - HPC systems are often rented, leased or shared, which can limit the effectiveness of firewalls and invalidate air gap solutions
        - Securing HPC systems:
            - deploy head nodes and route all outside traffic through them, isolating parts of a system
            - "fingerprint" HPC systems to understand use, and detect anomalous behavior

- 3.5.14 Edge computing systems

    - **Edge computing**: philosophy of network design where data and compute resources are located as close as possible, at or near the network edge, to optimize bandwidth use while minimizing latency; intelligence and processing are contained within each device, and each device can process it's own data locally
        - Securing edge computing:
            - this technology creates additional network edges that result in increased levels of complexity
            - visibility, control, and correlation requires a Zero Trust access-based approach to address security on the LAN edge, WAN edge and cloud edge, as well as network management
            - edge-based computing devices,especially IoT devices, are often produced with limited security forethought
            - devices on your network, no matter where they reside, need to be configured, managed, and patched using a consistent policy and enforcement strategy
            - use intelligence from side-channel signals that can pick up hardware trojans and malicious firmware
            - attend to physical security
            - deploy IDS on the network side to monitor for malicious traffic
            - in many scenarios, you are an edge customer, and likely will need to rely on a vendor for some of the security and vulnerability remediation

- 3.5.15 Virtualized systems

    - **Virtualized systems**: used to host one or more OSs within the memory of a single host computer, or to run apps not compatible with the host OS
        - Securing virtualized systems:
            - the primary component in virtualization is a hypervisor which manages the VMs, virtual data storage, virtual network components
            - the hypervisor represents an additional attack surface
            - in virtualized environments, you need to protect both the VMs and the physical infrastructure/hypervisor
            - hypervisor admin accounts/credentials and service accounts are targets because they often provide access to VMs and their data; these accounts should be protected

            - virtual hosts should be hardened; to protect the host, avoid using it for anything other than hosting virtualized elements
            - virtualized systems should be security tested via vuln assessment and penetration testing
            - virtualization doesn't lessen the security management requirements of an OS, patch management is still required
            - be aware of VM Sprawl and Shadow IT
            - **VM escape**: occurs when software within a guest OS is able to breach the isolation protection provided by the hypervisor
            - VM escape minimizaton:
                - keep highly sensitive systems and data on separate physical machines
                - keep all hypervisor software current with vendor-released patches
                - monitor attack, exposure and abuse indexes for new threats to virtual machines (which might be better protected); often, virtualization administrators have access to all virtuals

3.6 Select and determine cryptographic solutions (OSG-9 Chpts 6,7)

- 3.6.1 Cryptographic lifecycle (e.g., keys, algorithm selection)
    - Keep **Moore’s Law** in mind (processing capabilities of state-of-the-art microprocessors double about every 2 years), and have appropriate governance controls in place to ensure that algorithms, protocols, and key lengths selected are sufficient to preserve the integrity of the cryptosystems for as long as necessary -- to keep secret information safe
    - Specify the cryptographic algorithms (such as AES, 3DES, and RSA) acceptable for use in an organization
    - Identify the acceptable key lengths for use with each algorithm based on the sensitivity of the info transmitted
    - Enumerate the secure transaction protocols (e.g. TLS) that may be used
    - As computing power goes up, the strength of cryptographic algorithms goes down; keep in mind the effective life of a certificate or cert template, and of cryptographic systems
    - TLS uses an ephemeral symmetric session key between server and client, exchanged using asymmetric crtypography; note that all content is protected using symmetric cryptography
    - Beyond brute force, consider things like the discovery of a bug or an issue with an algorithm or system
    - NIST defines the following terms that are commonly used to describe algorithms and key lengths:
        - approved (an algorithm that is specified as a NIST or FIPS recommendation)
        - acceptable (algorithm + key length is safe today)
        - deprecated (algorithm and key length is OK to use, but brings some risk)
        - restricted (use of the algorithm and/or key length is deprecated and should be avoided)
        - legacy (the algorithm and/or key length is outdated and should be avoided when possible)
        - disallowed (algorithm and/or key length is no longer allowed for the indicated use)
- 3.6.2 Cryptographic methods (e.g., symmetric, asymmetric, elliptic curves, quantum)
    - **Symmetric** encryption: uses the same key for encryption and decryption
        - symmetric encryption uses a shared secret key available to all users of the cryptosystem

- symmetric encryption is faster than asymmetric encryption because smaller keys can be used for the same level of protection
- downside is that users or systems must find a way to securely share the key and hope the key is used only for the specified communication
- symmetric-key encryption can use either stream ciphers or block ciphers
- primarily employed to perform bulk encryption and provides only for the security service of confidentiality
- "same" is a synonym for symmetric
- "different" is a synonym for asymmetric
- **total number of keys** required to completely connect n parties using symmetric cryptography is given by this formula:
    - **(n(n - 1)) / 2**
- symmetric cryptosystems operate in several discrete modes:
    - **Electronic Code Book (ECB) mode**: the simplest and weakest of the modes; each block of plaintext is encrypted separately, but they are encrypted in the same way
        - advantages:fast, blocks can be processed simultaneously
        - disadvantages: any plaintext duplication would produce the same ciphertext
    - **Cipher Block Chaining (CBC) mode**: a block cipher mode of operation that encrypts plaintext by using an operation called XOR (exclusive-OR); XORing a block with the previous ciphertext block is known as "chaining"; this means that the decryption of a block of ciphertext depends on all the preceding ciphertext blocks; CBC uses an Initialization Vector or IV, which is a random value or nonce shared between sender and receiver
        - advantages: CBC uses the previous ciphertext block to encrypt the next plaintext block, making it harder to deconstruct; XORing process prevents identical plaintext from producing identical ciphertext; a single bit error in a ciphertext block affects the decryption of that block and the next, making it harder for attackers to exploit errors
        - disadvantages: you have to process the blocks in order, not simultaneously (so it's slower); CBC is also vulnerable to POODLE and GOLDENDOODLE attacks
    - **Cipher Feedback (CFB) mode**: similar to CBC, it uses an IV and the cipher from the previous block; the main difference is that with CFB, the cipher from the previous block is encrypted first, then XORed with the current block
        - advantages: CFB is considered to be faster than CBC even though it’s also sequential
        - disadvantages: if there’s an error in one block, it can carry over into the next block
    - **Output Feedback (OFB) mode**: OFB turns a block cipher into a synchronous stream cipher; based on an IV and the key, it generates keystream blocks which are then simply XORed with the plaintext data; as with CFB, the encryption and decryption processes are identical, and no padding is required
        - advantages: OFB mode doesn't need a unique nonce for each message, which can simplify the management and generation of nonces; it's resistant to replay attacks

        - disadvantages: no data integrity protection, vulnerability to IV management issues, potential for error propagation if a ciphertext block is corrupted, and lack of parallelization capabilities due to the dependence of the keystream on previous blocks
      - **Counter (CTR) mode**: key feature is that you can parallelize encryption and decryption, and it doesn't require chaining; it uses a counter function to generate a nonce value for each block's encryption; the nonce number (aka the counter) gets encrypted and then XORed with the plaintext to generate ciphertext; the resulting ciphertext should also always be unique
        - advantage: CTR mode is fast, and considered to be secure
        - disadvantage: lacks integrity, so we need to use hashing
      - **Galois/Counter (GCM) mode**: combines counter mode (CTR) with Galois authentication; we can not only encrypt data, but we can authenticate where the data came from (providing both data integrity and confidentiality); includes authentication data, and uses hashing as starting values
        - advantages: extremely fast, GCM is recognized by NIST and used in the IEEE 802.1AE standard
        - disadvantages:most stated disadvantages seem to be around implementation burdens
      - **Counter with Cipher Block Chaining Message Authentication Code (CCM) mode**: uses counter mode so there is no error propogation; there is no duplication, but uses chaining, so cannot run in parallel; MAC or message authentication code: provides authentication and integrity
        - advantages: no error propogation, provides authentication and integrity
        - disadvantages: cannot be run in parallel
    - Examples of symmetric algorithms: Twofish, Serpent, AES (Rijndael), Camellia, Salsa20, ChaCha20, Blowfish, CAST5, Kuznyechik, RC4/5/6, DES, 3DES, Skipjack, Safer, and IDEA
  - **Asymmetric** encryption: process that uses different keys for encryption and decryption, and in which the decryption key is computationally not possible to determine given the encryption key itself
    - Asymmetric (AKA public key, since one key of a pair is available to anybody) algorithms provide convenient key exchange mechanisms and are scalable to very large numbers of users (addressing the two most significant challenges for users of symmetric cryptosystems)
    - Asymmetric cryptosystems avoid the challenge of sharing the same secret key between users, by using pairs of public and private keys to allow secure communication without the overhead of complex key distribution
    - **Public key**: one part of the matching key pair, which can be shared or published
    - Besides the public key, there is a private key that should remain private and protected
    - Private key secrecy and integrity of an asymmetric encryption process are entirely dependent upon protecting the value of the private key
    - While asymmetric encryption is slower, it is best suited for sharing between two or more parties
    - Asymmetric encryption provides confidentiality, authentication and non-repudiation
    - Most common asymmetric cryptosystems in use today:

            - Rivest-Shamir-Adleman (RSA): depends on factoring the product of prime numbers
            - Diffie-Hellman: depends on modular arithmetic
            - ElGamal: extension of Diffie-Hellman that depends on modular arithmetic
            - Eliptical Curve Cryptography (EEC): elliptic curve algorithm depends on the elliptic curve discrete logarithm problem and provides more security than other algorithms when both are used with keys of the same length
    - Check out Practical Cryptography for Developers for a deeper dive
- 3.6.3 Public Key Infrastructure (PKI)
    - **Public Key Infrastructure (PKI)**: hierarchy of trust relationships permitting the combination of asymmetric and symmetric cryptography along with hashing and digital certificates (giving us hybrid cryptography)
        - A PKI issues certificates to computing devices and users, enabling them to apply cryptography (e.g., to send encrypted email messages, encrypt websites or use IPsec to encrypt data communications)
        - Many vendors provide PKI services; you can run a PKI privately and solely for your own org, you can acquire certificates from a trusted third-party provider, or you can do both (which is common)
        - A PKI is made up of
            - **certification authorities (CAs)**: servers that provide one or more PKI functions, such as providing policies or issuing certificates
            - certificates: issued to other certification authorities or to devices and users
            - policies and procedures: such as how the PKI is secured, and
            - templates: a predefined configuration for specific uses, such as a web server template
            - CAs generate digital certificates containing the public keys of system users; users then distribute these certificates to people with whom they want to communicate; certificate recipients verify a certificate using the CA's pubic key
        - There are other components and concepts you should know for the exam:
            - A PKI can have multiple tiers:
                - single tier means you have one or more servers that perform all the functions of a PKI
                - two tiers means there is an offline root CA (a server that issues certificates to the issuing CAs but remains offline most of the time) in one tier, and issuing CAs (the servers that issue certificates to computing devices and users) in the other tier
                - servers in the second tier are often referred to as intermediate CAs or subordinate CAs
                - three tier means you can have CAs that are responsible only for issuing policies (and they represent the second tier in a three-tier hierarchy)
                    - in such a scenario, the policy CAs should also remain offline and be brought online only as needed
            - Generally, the more tiers, the more security (but proper configuration is critical)
                - the more tiers you have, the more complex and costly the PKI is to build and maintain
            - A PKI should have a certificate policy and a certificate practice statement (CSP)

        - certificate policy: documents how your org handles items like requestor identities, the uses of certificates and storage of private keys
        - CSP: documents the security configuration of your PKI and is usually available to the public
      - Besides issuing certificates, a PKI has other duties:
        - a PKI needs to be able to provide certificate revocation information to clients
        - if an administrator revokes a certificate that has been issued, clients must be able to get that info from your PKI
        - storage of private keys and info about issued certificates (can be stored in a database or a directory)
      - PKI uses LDAP when integrating digital certs into transmissions
- 3.6.4 Key management practices
  - **Key management practices**: include safeguards surrounding the creation, distribution, storage, destruction, recovery, and escrow of secret keys
    - Cryptography can be used as a security mechanism to provide confidentiality, integrity, and availability only if keys are not compromised
    - Three main methods are used to exchange secret keys:
      - offline distribution
      - public key encryption, and
      - the Diffie-Hellman key exchange algorithm
    - Key management can be difficult with symmetric encryption but is much simpler with asymmetric encryption
    - There are several tasks related to key management:
      - Key creation
      - **Key distribution**: the process of sending a key to a user or system; it must be secure and it must be stored in a secure way on the computing device
        - keys are stored before and after distribution; when distributed to a user, it can't hang out on a user's desktop
      - Keys shouldn't be in cleartext outside the crypography device
      - Key distribution and maintenance should be automated (and hidden from the user)
      - Keys should be backed up!
      - **Key escrow**: process or entity that can recover lost or corrupted cryptographic keys
        - **multiparty key recovery**: when two or more entities are required to reconstruct or recover a key
        - **m of n control**: you designate a group of (n) people as recovery agents, but only need subset (m) of them for key recovery
        - **split custody**: enables two or more people to share access to a key (e.g. for example, two people each hold half the password to the key)
      - Key rotation: rotate keys (retire old keys, implement new) to reduce the risks of a compromised key having access
      - Key states:
        - suspension: temporary hold
        - revocation: permanently revoked
        - expiration

        - destruction
      - See NIST 800-57, Part 1
- 3.6.5 Digital signatures and digital certificates
  - **Digital signatures**: provide proof that a message originated from a particular user of a cryptosystem, and ensures that the message was not modified while in transit between two parties
    - Digital signatures rely on a combination of two major concepts — public key cryptography, and hashing functions
    - Digitally signed messages assure the recipient that the message truly came from the claimed sender, enforcing nonrepudiation
    - Digitally signed messages assure the recipient that the message was not altered while in transit; protecting against both malicious modification (third party altering message meaning), and unintentional modification (faults in the communication process)
    - Digital signature process does not provide confidentiality in and of itself (only ensures integrity, authentication, and nonrepudiation)
    - To digitally sign a message, first use a hashing function to generate a message digest; then encrypt the digest with your private key
    - To verify a digital signature, decrypt the signature with the sender's public key and compare the message digest to the one you generate yourself: if they match, the message is authentic
  - FIPS 186-5 specifies four techniques for the generation and verification of digital signatures that can be used for the protection of data: the Digital Signature Algorithm (DSA), the Rivest-Shamir-Adleman Algorithm (RSA), the Elliptic Curve Digital Signature Algorithm (ECDSA), and the Edwards-curve Digital Signature Algorithm (EdDSA)
- 3.6.6 Digital Non-repudiation
  - Here non-repudiation refers to methods ensuring certainty about data origins; in general, the inability to deny
  - Non-repudiation of origin: sender cannot deny having sent a particular message
  - Non-repudiation of delivery: receiver cannot say they have received a different message (other than what they actually received)
  - Most common method of non-repudiation is digital signatures
  - Digital signatures rely on certificates
  - If a digital signature was verified with the public key of the sender, then we know that it was created using the sender's private key
  - Private key should only be known to the sender, so the verification proves to the recipient that the signature came from the sender, providing origin authentication
  - The recipient (or anyone else) can demonstrate that process to a third party providing nonrepudiation
  - Data encryption provides confidentiality
- 3.6.7 Integrity (e.g., hashing)
  - Hash Functions have a very simple purpose — they take a potentially long message and generate a unique output value derived from the content of the message called a **message digest**
    - hash function implements encryption with a specified algorithm, but without a key
    - used to ensure message sent by the originator is the same one received by recipient
    - input can be of any length

        - output has a fixed length
        - the hash function is relatively easy to compute for any input
        - the hash function is one-way, meaning it is extremely difficult to determine the input given the hash function output
        - the hash function should be collision-resistant, meaning it is extremely hard to find two messages that produce the same hash value output
        - hashes are used for storing passwords, with email, and for file download integrity verification
    - Hashing and integrity: if the hash generated by sender, and separately by the receiver match, then we have integrity
    - Successors to the Secure Hash Algorithm (SHA), SHA-2, and SHA-3, make up the government standawrd message digest funtion
        - SHA-2 supports variable-length message digests, ranging up to 512 bits
        - SHA-3 improves upon the security of SHA-2 and supports the same hash lengths

3.7 Understand methods of cryptanalytic attacks (OSG-9 Chpts 7,14,21)

- 3.7.1 Brute force
    - **Brute force**: an attack that attempts every possible valid combination for a key or password
        - they involve using massive amounts of processing power to methodically guess the key used to secure cryptographic communications
- 3.7.2 Ciphertext only
    - **Ciphertext only**: an attack where you only have the encrypted ciphertext message at your disposal (not the plaintext)
        - if you have enough ciphertext samples, the idea is that you can decrypt the target ciphertext based on the samples
        - frequency analysis is a technique that is helpful against simple ciphers (see below)
- 3.7.3 Known plaintext
    - **Known plaintext**: in this attack, the attacker has a copy of the encrypted message along with the plaintext message used to generate the ciphertext (the copy); this knowledge greatly assists the attacker in breaking weaker codes
    - **Linear cryptanalysis**: a known plaintext attack, in which the attacker studies probabilistic linear relations referred to as linear approximations among parity bits of the plaintext, the Ciphertext and the hidden key
- 3.7.4 Frequency analysis
    - **Frequency analysis**: an attack where the characteristics of a language are used to defeat substitution ciphers
        - for example in English, the letter "E" is the most common, so the most common letter in an encrypted cyphertext could be a substitution for "E"
        - other examples might include letters that appear twice in sequence, as well as the most common words used in a language
- 3.7.5 Chosen ciphertext
    - **Chosen ciphertext**: in a chosen ciphertext attack, the attacker has access to one or more plaintexts of arbitrary ciphertexts; i.e. the attacker has the ability to decrypt chosen portions of the ciphertext message, and use the decrypted portion to discover the key
    - **Chosen-plaintext attack (CPA)**: an attack model for cryptanalysis which presumes that the attacker can obtain the ciphertexts for arbitrary plaintexts, with the goal to gain information

  - that reduces the security of the encryption scheme; a CPA is more powerful than a known plaintext attack; however a chosen-plaintext is less powerful than a chosen ciphertext
  - **Differential cryptanalysis**: a type of chosen plaintext attack, and a general form of cryptanalysis applicable primarily to block ciphers, but also to stream ciphers and cryptographic hash functions; in the broadest sense, it is the study of how differences in information input can affect the resultant difference at the output; advanced methods such as differential cryptanalysis are types of chosen plaintext attacks
    - as an example, an attacker may try to get the receiver to decrypt modified ciphertext, looking for that modification to cause a predictable change to the plaintext
- 3.7.6 Implementation attacks
  - **Implementation attack**: attempts to exploit weaknesses in the implementation of a cryptography system
    - focuses on exploiting the software code, not just errors or flaws but the methodology employed to program the encryption system
    - in this type of attack, attackers look for weaknesses in the implementation, such as a software bug or outdated firmware
- 3.7.7 Side-channel
  - **Side-channel**: these attacks seek to use the way computer systems generate characteristic footprints of activity, such as changes in processor utilization, power consumption, or electromagnetic radiation to monitor system activity and retrieve information that is actively being encrypted
    - similar to an implementation attack, side-channel attacks look for weaknesses outside of the core cryptography functions themselves
    - a side-channel attack could target a computer’s CPU, or attempt to gain key information about the environment during encryption or decryption by looking for electromagnetic emissions or the amount of execution time required during decryption
    - side-channel characteristics information are often combined together to try to break down the cryptography
    - timing attack is an example
- 3.7.8 Side-channel
  - **Fault-Injection**: the attacker attempts to compromise the integrity of a cryptographic device by causing some type of external fault
    - for example, using high-voltage electricity, high or low temperature, or other factors to cause a malfunction that undermines the security of the device
- 3.7.9 Timing
  - **Timing**: timing attacks are an example of a side-channel attack where the attacker measures precisely how long cryptographic operations take to complete, gaining information about the cryptographic process that may be used to undermine its security
- 3.7.10 Man-in-the-middle (MITM)
  - **Man-in-the-middle (MITM) (AKA on-path)**: in this attack a malicious individual sits between two communicating parties and intercepts all communications (including the setup of the cryptographic session)
    - attacker responds to the originator's initialization requests and sets up a secure session with the originator
    - attacker then establishes a second secure session with the intended recipient using a different key and posing as the originator

        - attacker can then "sit in the middle" of the communication and read all traffic as it passes between the two parties
- 3.7.11 Pass the hash
    - **Pass the hash (PtH)**: a technique where an attacker captures a password hash (as opposed to the password characters) and then simply passes it through for authentication and potentially lateral access to other networked systems
        - the threat actor doesn't need to decrypt the hash to obtain a plain text password
        - PtH attacks exploit the authentication protocol, as the passwords hash remains static for every session until the password is rotated
        - attackers commonly obtain hashes by scraping a system's active memory and other techniques
- 3.7.12 Kerberos exploitation
    - **Overpass the Hash**: alternative to the PtH attack, used when NTLM is disabled on the network (AKA pass the key)
    - **Pass the Ticket**: in this attack, attackers attempt to harvest tickets held in the lsass.exe process
    - **Silver Ticket**: a silver ticket uses the captured NTLM hash of a service account to create a ticket-granting service (TGS) ticket (the silver ticket grants the attacker all the privileges granted to the service account)
    - **Golden Ticket**: if an attacker obtains the hash of the Kerberos service account (KRBTGT), they can create tickets at will within Active Directory (this provides so much power it is referred to as having a golden ticket)
    - **Kerberos Brute-Force**: attackers use the Python script kerbrute.py on Linux, and Rubeus on Windows systems; tools can guess usernames and passwords
    - **ASREPRoast**: ASREPRoast identifies users that don't have Kerberos preauthentication enabled
    - **Kerberoasting**: kerberoasting collects encrypted ticket-granting service (TGS) tickets
- 3.7.13 Ransomeware
    - **Ransomware**: a type of malware that weaponizes cryptography
        - using many of the same techniques as other types of malware, ransomware gens an encryption key, and encrypts critical files
        - this encryption renders the data inaccessible to the authorized user or anyone else other than the malware author
        - often threatning to publically release sensitive data if ransome is not paid
        - 2020 study, 56% of orgs suffered a ransomeware attack, 27% of orgs who reported an attack chose to pay, on average ~$1.1m
        - seek legal advice prior to engaging with ransomware authors

3.8 Apply security principles to site and facility design (OSG-9 Chpt 10)

- **Secure facility plan**: outlines the security needs of your org and emphasizes methods or mechanisms to employ to provide security, developed through risk assessment and critical path analysis

    - **critical path analysis (CPA)**: a systematic effort to identify relationships between mission-critical apps, processes, and operations and all the necessary supporting components
    - During CPA, evaluate potential **technology convergence**: the tendency for various technologies, solutions, utilities, and systems to evolve and merge over time, which can result

in a single point of failure and a more valuable target
    - A secure facility plan is based on a layered defense model
    - Site selection should take into account cost, location, and size (but security should always take precedence), that the building can withstand local extreme weather events, vulnerable entry points, and exterior objects that could conceal break-in;
        - Key elements of site selection:
            - visibility
            - composition of the surrounding area
            - area accessibility

- Facility Design:

    - The top priority of security should always be the protection of the life and safety of personnel
    - In the US, follow the guidelines and requirements from Occupational Safety and Health Administration (OSHA), and Environmental Protection Agency (EPA)
    - A key element in designing a facility for construction is understanding the level required by your org and planning for it before beginning construction
    - **Crime Prevention Through Environmental Design (CPTED)**: a well-established school of thought on "secure architecture" - an archiectural approach to building and space design that emphasizes passive features to reduce the likelihood of criminal activity
        - core principle of CPTED is that the design of the physical environment can be managed/manipulated, and crafted with intention in order to create behavioral effects or changes in people present in those areas that result in reduction of crime as well as a reduction of the fear of crime
        - CPTED stresses three main principles:
            - **natural access control**: the subtle guidance of those entering and leaving a building
                - make the entrance point obvious
                - create internal security zones
                - areas of the same access level should be open, but restricted/closed areas should seem more difficult to access
            - **natural surveillance**: any means to make criminals feel uneasy through increased opportunities to be observed
                - walkways/stairways are open, open areas around entrances
                - areas should be well lit
            - **natural territorial reinforcement**: attempt to make the area feel like an inclusive, caring community
    - Overall goal is to deter unauthorized people from gaining access to a location (or a secure portion), prevent unauthorized personnel from hiding inside or around the location, and prevent unauthorized from committing crime
    - There are several smaller activities tied to site and facility design, such as upkeep and maintenance: if property is run down or appears to be in disrepair, it gives attackers the impression that they can act with impunity on the property

3.9 Design site and facility security controls (OSG-9 Chpt 10)

- Note that alghough the topics in this section cover mostly interior spaces, physical security is applicable to both interior and exterior of a facility

- 3.9.1 Wiring closets/intermediate distribution facilities
    - **Wiring closets/intermediate distribution facilities (IDF)**: A wiring closet or IDF is typically the smallest room that holds IT hardware
        - wiring closet is AKA premises wire distribution room, main distribution frame (MDF), intermediate distribution frame (IDF), and telecommunications room, and it is referred to as an IDF in (ISC)^2 CISSP objective 3.9.1
        - where networking cables for the building or a floor are conntected to equipment (e.g. patch panels, switches, routers, LAN extenders etc)
        - usually includes telephony and network devices, alarm systems, circuit breaker panels, punch-down blocks, WAPs, video/security
        - may include a small number of servers
        - access to the wiring closest/IDF should be restricted to authorized personnel responsible for managing the IT hardware
        - use door access control (i.e. electronic badge system or electronic combination lock)
        - from a layout perspective, wiring closets should be accessible only in private areas of the building interiors; people must pass through a visitor center and a controlled doorway prior to be able to enter a wiring closet
- 3.9.2 Server rooms/data centers
    - **Server rooms/data centers**: server rooms, data centers, communication rooms, server vaults, and IT closets are enclosed, restricted, and protected rooms where mission critical servers and networks are housed
        - a server room is a bigger version of a wiring closet, much smaller than a data center
        - a server room typically houses network equipment, backup infrastructure and servers (more archaic versions include telephony equipment)
        - server rooms should be designed to support optimal operation of IT infrastructure and to block unauthorirzed human access or intervention
        - server rooms should be located at the core of the building (avoid ground floor, top floor, or in the basement)
        - server rooms should have a single entrance (and an emergency exit)
        - server room should block unauthorized access, and entries and exits should be logged
        - datacenters are usually more protected than server rooms, and can include guards and mantraps
        - datacenters can be single-tenant or multitenant
- 3.9.3 Media storage facilities
    - **Media storage facilities**:often store backup tapes/disks, blank, resuable and other media, and should be protected just like a server room
        - depending on requirements a cabinet or safe could suffice
        - new blank media, and media that is reused (e.g. thumb drives, flash memory cards, portable hard drives) should be protected against theft and data remnant recovery
        - concerns include theft, corruption, data remnant recovery
        - other recommendations:
            - employ a media librarian or custodian
            - use check-in/check-out process for media tracking
            - run a secure drive sanitization or zeroization when media is returned
        - note: a safe is a movable secured container that's not integrated into a building's construction; a vault is a permanent safe integrated into construction

- 3.9.4 Evidence storage
  - **Evidence storage**: as cybercrime events continue to increase, it is import to retain logs, audit trails, drive images, VM snapshots and other records of digital events; the evidence storage exists to preserve chain of custody
    - a key part of incident response is to gather evidence to perform root cause analysis
    - an evidence storage room should be protected like a server room or media storage facility
    - an evidence storage room can contain physical evidence (such as a smartphone) or digital evidence (such as a database)
    - protections should include dedicated/isolated storage facilities, offline storage, activity tracking, hash management, access restrictions, and encryption
- 3.9.5 Restricted and work area security
  - **Restricted and work area security**: covers the design and configuration of internal security, including work and visitor areas
    - includes areas that contain assets of higher value/importance which should have more restricted access
    - restricted work areas are used for sensitive operations, such as network/security ops
    - protection should be similar to a server room, but video surveillance is typically limited to entry and exit points
- 3.9.6 Utilities and heating, ventilation, and air conditioning (HVAC)
  - Power management in ascending order: surge protectors, power/power-line conditioner, uninterruptible power supply (UPS), generators
  - Types of UPS:
    - double conversion: functions by taking power from the wall outlet, storing it in a battery, pulling power out of the battery and feeding that power to the device/devices
    - line-interactive: has a surge protector, battery charger/inverter and voltage regulator positioned between the grid power source and the equipment (battery is not in line under normal conditions)
  - Commercial power problem types:
    - **fault**: momentary loss of power
    - **blackout**: complete loss of power
    - **sag**: momentary low voltage
    - **brownout**: prolonged low voltage
    - **spike**: momentary high voltage
    - **surge**: prolonged high voltage
    - **inrush**: initial surge of power associated with connecting to a power source
  - Think through types of physical controls for HVAC:
    - restrict duct space continuity to controlled areas
    - use separate and redundant HVAC systems for computer equipment
    - rooms containing primarily computers should have:
      - temps kept at 59 to 89.6 deg Fahrenheit (15 to 32 deg Celsius)
      - humidity should be maintained between 20 and 80 percent
    - note that even on nonstatic carpeting, if the env has low humidity, a 20k-volt static discharge is possible; and even minimal levels of static electricity can destroy electronic equipment
  - Datacenter:

    - should be on different power circuits from occupied areas
    - common to use a backup generator
- 3.9.7 Environmental issues
  - Environmental monitoring is the process of measuring and evaluating the quality of the environment within a given structure (e.g. temperature, humidity, dust, smoke), using things like chemical, biological, radiological, and microbiological detectors
  - Halon starves a fire of oxygen by disrupting the chemical reaction of combustion, but degrades into toxic gases at 900 degrees Fahrenheit, and is not environmentally friendly
  - If water-based sprinklers are used for fire suppression, damage to electronic equipment is likely; automate the shutoff of electricity prior to sprinkler trigger
  - Other environmental issues include earthquakes, power outages, tornados and wind
  - Secondary facilities should be located far enough away from the primary to ensure they won't be damaged by the same event
  - Water leakage and flooding should be addressed in your environmental safety policy and procedures; water and electricity together is sure to cause damange; locate server rooms and critical eqiupment away from any water source or transport pipes
- 3.9.8 Fire prevention, detection, and suppression
  - Protecting personnel from harm should always be the most important goal of any security or protection system!
  - In addition to protecting people, fire detection and suppression is designed to keep asset damage caused by fire, smoke, heat, and suppression materials to a minimum
  - **Fire triangle**: three triangle corners represent fuel, heat, and oxygen; the center of the triangle represents the chemical reaction among these three elements
    - if you can remove any one of the four items from the fire triangle, the fire can be extinguished
  - Fire suppression mediums:
    - water suppresses temperature
    - soda acid and other dry powders suppress the fuel supply
    - carbon dioxide (CO2) suppresses the oxygen supply
    - halon substitutes and other nonflammable gases interfere with the chemistry of combustion and/or suppress the oxygen supply
  - Fire stages:
    - **Stage 1**: incipient stage: at this stage, there is only air ionization and no smoke
    - **Stage 2**: smoke stage: smoke is visible from the point of ignition
    - **Stage 3**: flame stage: this is when a flame can be seen with the naked eye
    - **Stage 4**: heat stage: at stage 4, there is an intense heat buildup and everything in the area burns
  - Fire extinguisher classes:
    - **Class A**: common combustibles
    - **Class B**: liquids
    - **Class C**: electrical
    - **Class D**: metal
    - **Class K**: cooking material (oil/grease)
  - Four main types of suppression:
    - **wet pipe system**: (AKA closed head system): is always filled with water; water discharges immediately when suppression is triggered

    - **dry pipe system**: contains compressed inert gas
    - **preaction system**: a variation of the dry pipe system that uses a two-stage detection and release mechanism
    - **deluge system**: uses larger pipes and delivers larger volume of water
  - Note: Most sprinkler heads feature a glass bulb filled with a glycerin-based liquid; this liquid expands when it comes in contact with air heated to between 135 and 165 degrees; when the liquid expands, it shatters its glass confines and the sprinkler head activates
- 3.9.9 Power (e.g., redundant, backup)
  - Consider designing power to provide for high availability
  - Most power systems have to be tested at regular intervals
  - As part of the design, mandate redundant power systems to accommodate testing, upgrades and other maintenance
  - Additionally, test failover to a redundant power system and ensure it is fully functional
  - The International Electrical Testing Association (NETA) has developed standards around testing power systems
  - Battery backup/fail-over power (including UPS/generators):
    - this is a system that collects power into a battery but can switch over to pulling power from the battery when the power grid fails
    - generally, this type of system was implemented to supply power to an entire building rather than just one or a few devices

Domain 4 **Communication and Network Security**

Networking can be one of the more complex exam topics; if you have a networking background, you likely won’t find this domain difficult-- if not, spend extra time in this section and consider diving deeper into topics that are fuzzy

- **ACK**: an acknowledgement of a signal being received
- **Active-active, active-passive clustering**: a data resiliency architecture in which client workloads are distributed across two or more nodes in a cluster to keep data safe and available in the event of an unexpected component failure; active-active can use the full throughput capability of both devices; active-passive can only handle throughput of a single device allowing the secondary device to remain ready (but not passing traffic) until needed
- **ARP**: Address Resolution Protocol; used to resolve IP addresses into Media Access Control (MAC) addresses; provides for direct communication between two devices within the same LAN segment
- **ARP poisoning**: AKA ARP spoofing, where an attacker sends malicious ARP messages to a local network with the goal of associating the attacker’s MAC address with the IP address of another device (typically the default gateway or another trusted device) in the network; once successful, the attacker can intercept, modify, or block data intended for that IP address, facilitating attacks like man-in-the-middle (MITM), eavesdropping, or denial of service (DoS)
- **APT**: Advanced Persistent Threat is an agent/org that plans, organizes, and carries out highly sophisticated attacks against a target person, org, or industry over a period of time (months or even years); usually with a strategic goal in mind
- **API**: Application Programming Interface; code mechanisms that provide ways for apps to share data, emthods, or functions over a network (usually implemented in XML or JavaScript Object Notation (JSON))

- **Bandwidth**: amount of information transmitted over a period of time; can be applied to moving bits over a medium, or human processes like learning or education
- **Bluebugging**: a type of Bluetooth attack where an attacker exploits vulnerabilities in a device's Bluetooth connection to gain unauthorized access and control over it; the attacker can then use this access to listen to phone calls, read or send messages, steal data, or even manipulate the device's settings, all without the victim's knowledge; to combat this attack, turn off Bluetooth when nont in use, set to "non-discoverable", and use strong authentication
- **Bluejacking**: a relatively harmless Bluetooth-based attack where an attacker sends unsolicited messages to nearby Bluetooth-enabled devices, such as mobile phones, tablets, or laptops; it exploits the Bluetooth feature that allows devices to communicate over short distances (typically up to 10 meters) and works by pushing messages to any device with Bluetooth set to "discoverable" mode
- **Bluesnarfing**: Bluesnarfing is a Bluetooth-based attack where an attacker gains unauthorized access to data on a victim's Bluetooth-enabled device; unlike Bluejacking (which only sends unsolicited messages), bluesnarfing is a serious security threat because it involves stealing or retrieving sensitive data without the user's knowledge or consent
- **Bluesniffing**: Bluesniffing is a type of Bluetooth attack that involves eavesdropping on Bluetooth communications to intercept data being exchanged between devices; it's a form of passive Bluetooth reconnaissance where the attacker attempts to listen in on Bluetooth signals to gather information, much like Wi-Fi sniffing attacks on wireless networks
- **Bound networks**: AKA wired/Ethernet networks, where devices are connected by physical cables
- **Boundary routers**: they advertise routes that external hosts can use to reach internal hosts
- **Bridge**: device that aggregates separate network segments into a single network segment, operating at OSI layer 2 **CAM Table Flooding**: attack where switches don't know where to send traffic; prevented by enabling switch port security
- **Captive portal**: an authentication mechanism that redirects a newly-connected client to a web-based portal access control page
- **CHAP**: Challenge-Handshake Authentication Protocol, used by PPP servers to authenticate remote clients; encrypts both username and password, and performs periodic session reauthentication to prevent replay attacks
- **CSMA/CA**: Carrier Sense Multiple Access with Collission Avoidance is a method of network flow control
- **CSMA/CD**: Carrier Sense Multiple Access with Colliion Detection is a method of network flow control, where if > 1 station accesses the network at the same time, other stations detect and re-try their transmission
- **Circuit-switched network**: network that uses a dedicated circuit between endpoints
- **CDMA**: Code-Division Multiple Access: a method of encoding several sources of data so they can all be transmitted over a single RF carrier by one transmitter, or by using a single RF carrier frequency with multiple transmitters; the data from each call is encoded with a unique key, and calls are transmitted at once
- **Collision Domain**: set of systems that can cause a sollision if they transmitted at the same time; note that broadcast domain is the set of systems that can receive a breadcast from each other
- **Concentrator**: provides communication capability between many low-speed, usually asynchronous channels and one or more high-speed, usually synchronous channels. Usually different speeds, codes, and protocols can be accommodated on the low-speed side; multiplexed into one signal

- **CDN**: Content Distribution Network is a large distributed system of servers deploye in multiple data centers, with a goal of Quality of Service (QoS) and availability requirements
- **Control plane**: part of a network that controls how data packets are forwarded — meaning how data is sent from one place to another; e.g. the process of creating a routing table is considered part of the control plane; control of network functionality and programmability is directly made to devices at this layer
- **Northbound/Southbound interface**: A northbound interface lets a specific component communicate with a higher-level component in the same network; a southbound interface is the opposite — enabling a specific component to communicate with a lower-level component
- **East/West traffic**: network traffic that is within a data, control, or application plane; within a data center or between geo dispersed locations
- **North/South traffic**: in SDN terms, data flowing up (northbound) and down (southbound) the stack of data/control/application planes; data flowing from the organization to external distinations (northbound), or into the org from external sources (southbound)
- **Converged protocol**: combines/converges standard protcols (such as TCP/IP) with proprietary/non-standard ones; they can complicate enterprise-wide security engineering efforts requiring specialist knowledge
- **DHCP**: Dynamic Host Configuration Protocol is an industry standard used to dynamically assign IP addresses to network devices
- **Disassociation attack**: AKA deauthentication attack, is a type of denial-of-service (DoS) attack targeting Wi-Fi networks; it exploits the management frames used in the Wi-Fi protocol to forcibly disconnect (disassociate) a user’s device from a wireless access point (AP); these attacks can cause disruptions in network availability and can be used as part of a larger attack, such as a man-in-the-middle (MITM) attack
- **DNS**: Domain Name Service is three interrelated elements: a service, a physical server, and a network protocol
- **DNS poisoning**: act of falsifying DNS info used by clients to reach a system; can be accomplished via a rogue DNS server, pharming, altering a hosts file, corrupting IP config, DNS query spoofing, and proxy falsification; examples:
    - HOSTS poisoning
    - authorized DNS server attack
    - caching DNS server attack
    - changing a DNS server address
    - DNS query spoofing
- **Domain hijacking**: or domain theft, is the malicious action of changing the registration of a domain name without the authorization of the owner
- **Evil twin**: a type of Wi-Fi attack where a malicious actor sets up a fake wireless access point (AP) that mimics a legitimate one, with the goal of tricking users into connecting to the rogue access point, allowing the attacker to intercept, monitor, and manipulate the victim’s network traffic, potentially capturing sensitive data such as login credentials, financial info, and personal communications
- **FDDI**: Fiber Distributed Data Interface is an ANSI X3T9.5 LAN standard; 100Mbps, token-passing using fiber-optic, up to 2 kilometers
- **FCoE**: Fibre Channel over Ethernet is a lightweight encapulsation protocol without the reliable data transport of TCP

- **Gateway device**: a firewall or other device that sits at the edge of the network to regulate traffic and enforce rules
- **Generic Routing Encapsulation (GRE)**: a protocol for encapsulating data packets that use one routing protocol inside the packets of another protocol; "encapsulating" means wrapping one data packet within another data packet, like putting a box inside another box; encapsulation is the addition of a header, possibly footer, to the data received by each layer from the layer above before handing it off to the layer below; the inverse action is deencapsulation; note that when using IPv4, the GRE header is inserted between the delivery and payload headers
- **Hypervisor**: aka virtual machine monitor/manager (VMM); the virtualization component that creates, manages, and operates VMs
- **Hypervisor Type I**: a native or bare-metal hypervisor; there is no host OS, the hypervisor installs directly onto the hardware instead of the host OS
- **Hypervisor Type II**: a hosted hypervisor; the hypervisor is installed as another app in the OS
- **ICMP**: Internet Control Message Protocol, standardized by IETF via RFC 792 to determine if a particular host is available
- **IGMP**: Internet Group Management Protocol, used to manage multicasting groups
- **Internetworking**: two different sets of servers/communication elements using network protocol stacks to communicate and coordinate activities **IV (Initialization Vector) attack**: occurs when attackers exploit weaknesses in how an IV is generated or used in cryptographic systems, particularly in stream ciphers and certain modes of block ciphers; the Initialization Vector (IV) is a random or pseudo-random value used to ensure that even if the same data is encrypted multiple times, the resulting ciphertext is different each time; if the IV is predictable, reused, or improperly managed, it can lead to security vulnerabilities, enabling attackers to decrypt data, tamper with messages, or perform replay attacks
- **LDAP**: lightweight directory access protocol uses simple (basic) authentication such as SSL/TLS, or SASL (Simple Authentication and Security Layer)
- **MAC filtering**: a list of authorized wireless client interface MAC addresses used by a WAP to block access to all nonauthoried devices
- **Managed Dection and Response (MDR)**: a service that monitors an IT environment in real-time to detect and resolve threats; not limited to endpoints, MDR focuses on threat detection and mediation; often a combination and integration of numerous technologies, such as SIEM, network traffic analysis (NTA), EDR, and IDS
- **Microsegmentation**: part of a zero trust strategy that breaks or divides up an internal network into very small (sometimes as small as a single device or important server/end-point), highly localized zones; each zone is separated from others by internal segmentation firewalls (ISFWs), subnets, or VLANs; note that at the limit, this places a firewall at every connection point
- **Network container names**: network containers are OSI layers 7-5: protocol data unit (PDU); layer 4: segment (TCP) or datagram (UDP); layer 3: packet; layer 2: frame; layer 1: bits
- **NFV**: Network Function Virtualization (AKA Virtual Network Function) that seeks to decouple functions, such as firewall management, intrusion detection, NAT and name service resolution, from specific hardware solutions, and make them virtual/software; the focus is to optimize distinct network services
- **Nonroutable IP addresses**: from RFC 1918; 10.0.0.0/8, 172.16.0.0/12, 192.168.0.0/16
- **PLC**: Packet Loss Concealment used in VoIP communications to mask the effect of dropped packets
- **Packet-Switched Network**: a network that doesn't use a dedicated connection between endpoints

- **Point-to-Point Protocol**: a standard method for transporting multiprotocol datagrams over point-to-point links
- **Ports 1024-4951**: registered ports used with non-system applications associated with vendors and devs
- **Ports 49152-65535**: dynamic ports (AKA private or non-reserved ports) used as temporary ports, often in association when a service is requested via a well-known port
- **Port Address Translation**: an extension of NAT (Network Address Translation) translating all addresses to one routable IP address and translate the source port number in the packet to a unique value
- **Proximity device/reader**: proximity devices can be passive, field-powered, or transponder; when a device passes near a proximity reader, the reader is able to determine the identity and authorization status of the bearer
- **Race condition (RCE)**: AKA race hazard is the condition of an electronics, software, or other system where the system's substantive behavior is dependent on the sequence or timing of other uncontrollable events, leading to unexpected or inconsistent results
- **RPC**: Remote Procedure Call is a protocol that enables one system to execute instructions on other hosts across a network infrastructure
- **Root of Trust**: a source that can always be trusted within a cryptographic system; because cryptographic security is dependent on keys to encrypt and decrypt data and perform functions such as generating digital signatures and verifying signatures, RoT schemes generally include a hardened hardware module; a RoT guarantees the integrity of the hardware prior to loading the OS of a computer
- **SIPS**: secure version of the Session Initialization Protocol for VoIP, adds TLS encryption to keep the session initialization process secure
- **Smartcard**: credit card-sized IDs, badges, or security passes with a magnetic stripe, bar code, or integrated circuit chip, containing info about the authorized bearer; used for identification or auth purposes
- **S/MIME**: provides the following cryptographic security services for electronic messaging applications: - Authentication - Message integrity - Non-repudiation of origin (using digital signatures) - Privacy - Data security (using encryption)
    - S/MIME specifies the MIME type application/pkcs7-mime (smime-type "enveloped-data") for data enveloping (encrypting) where the whole (prepared) MIME entity to be enveloped is encrypted and packed into an object which subsequently is inserted into an application/pkcs7-mime MIME entity
    - S/MIME is the emerging standard for secure email / encrypted messages
- **SNMP**: Simple Network Management Protocol, is a protocol for collecting and organizing info about managed devices on IP networks; it can be used to determine the health of devices such as routers, switches, servers, workstations, etc
- **Smurf attack**: ICMP echo request sent to the network broadcast address of a spoofed victim causing all nodes to respond to the victim with an echo reply
- **SPML**: Service Provisioning Markup Language is XML-based and designed to allow platforms to generate and respond to provisioning requests; uses the concept of requesting authority, a provisioning service point, and a provisioning service target; requesting authorities issue SPML requests to a provisioning service point; provisioning service targets are often user accounts and are required to be allowed unique identification of the data in its implementaion

- **STRP**: Secure Real-time Transport Protocol is an extension of Real-time Transport Protocol (RTP) that features encryption, confidentiality, message authentication, and replay protection to audio and video traffic
- **Multi-tiered firewall**: tiers are not the number of firewalls but the number of zones protected by the firewall; 2-tier protects two zones
- **Teardrop attack**: exploits reassembly of fragmented IP packets in the fragment offset field (indicating the start position or offset of data contained in a fragemented packet)
- **Terminal Emulation Protocol**: designed to provide access between hosts; allows a computer to act like a traditional terminal and send commands/receive output from a remote system via a graphical interface; examples include Telnet, SSH, and Kermit
- **Unbound (Wireless) Network**: network where the physical layer interconnections are done using radio, light, or some other means (not confined to wires, cables, or fibers); may or may not be mobile
- **VLAN hopping**: a method of attacking the network resources of a VLAN by sending packets to a port not usually accessible from an end system; the goal of this form of attack is to gain access to other VLANs on the same network
- **WAF**: Web Application Firewall is a software-based app that monitors and filters exchanges between applications and a host; usually inspect and filter conversations like HTTP/S

4.1 Assess and implement secure design principles in network architectures (OSG-9 Chpts 11,12)

- 4.1.1 Open System Interconnection (OSI) and Transmission Control Protocol/Internet Protocol (TCP/IP) models
  - **TCP/IP**: AKA DARPA or DOD model has four layers: Application (AKA Process), Transport (AKA Host-to-Host), Internet (AKA Internetworking), and Link (AKA Network Interface or Network Access)
  - **OSI**: Open Systems Interconnection (OSI) Reference Model developed by ISO (International Organization for Standardization) to establish a common communication structure or standard for all computer systems; it is an abstract framework
    - Communication between layers via **encapsulation** (at each layer, the previous layer's header and payload become the payload of the current layer) and **deencapsulation** (inverse action occurring as data moves up layers)

Layer	OSI model layer	TCP/IP model	PDU	Devices	Protocols
7	Application	Application	Data	L7 firewall	HTTP/s, DNS, DHCP, FTP, LPD, S-HTTP, TPFT, Telnet, SSH, SMTP, POP3, PEM, IMAP, NTP, SNMP, TLS/SSL, GBP, SIP, S/MIME, X Window, NFS etc.
6	Presentation	Application	Data	L7 firewall	JPEG, ASCII, MIDI etc
5	Session	Application	Data	L7 firewall	All the above

Layer	OSI model layer	TCP/IP model	PDU	Devices	Protocols
4	Transport	Transport (host-to-host)	Segments	L4 firewall	TCP (connection oriented), UDP (connectionless)
3	Network	Internet/IP	Packets	Router, Multilayer Switch, Router	IPv4, IPv6, IPSec, OSPF, EIGRP; ICMP, RIP, NAT
2	Data Link	Network Access	Frames	Switch, Bridge, NIC, Wireless Access Point	MAC, ARP, Ethernet 802.3 (Wired), CDP, LLDP, HDLC, PPP, DSL, L2TP, IEEE 802.11 (Wireless), SONET/SDH, VLANs
1	Physical	Network Access	Bits	All the above	Electrical signal (copper wire), Light signal (optical fibre), Radio signal (air)

OSI layers in detail

- Mnemonics:
  - from top: All People Seem To Need Delicious Pizza
  - from bottom: Please Do Not Throw Sausage Pizza Away
- Application Layer (7)
  - Responsible for:
    - interfacing user applications, network services, or the operating system with the protocol stack
    - identifying and establishing availability of communication partners
    - determining resource availability and
    - synchronizing communication
  - Uses data streams
- Presentation Layer (6)
  - Responsible for transforming data into the format that any system following the OSI model can understand
  - JPEG, ASCII, MIDI etc are used at the presentation lay
  - Associated tasks:
    - data representation
    - character conversion
    - data compression
    - data encryption

    - Uses data streams

  - Session Layer (5)

    - Responsible for establishing, maintaining, and terminating communication sessions between two computers
    - Three communication session phases:
      - connection establishment
        - **simplex**: one-way
        - **half-duplex**: both comm devices can transmit/receive, but not at the same time
        - **full-duplex**: both comm devices can transmit/receive at same time
      - data transfer
      - connection release
    - Uses data streams

  - Transport Layer (4)

    - Responsible for managing the integrity of a connection and controlling the session; providing transparent data transport and end-to-end transmission control
    - Defines session rules like how much data each segment can contain, how to verify message integrity, and how to determine whether data has been lost
    - Protocols that operate at the Transport layer:
      - Transmission Control Protocol (TCP)
        - the major transport protocol in the internet suite of protocols providing reliable, connection-oriented, full-duplex streams
        - emphasizing: full-duplex, connection-oriented protocol
        - uses three-way handshake: involves the following three steps: synchronize (SYN), synchronize-acknowledge (SYN-ACK), and acknowledge (ACK)
        - TCP header flags:
          - URG ACK PSH RST SYN FIN (mnemonic: Unskilled Attackers Pester Real Security Folks)
      - User Datagram Protocol (UDP)
        - connectionless protocol that provides fast, best-effort delivery of **datagrams** (self-container unit of data)
      - Transport Layer Security (TLS)
        - note: in the OSI model, TLS operates on four layers: Application, Presentation, Session, and Transport; in the TCP/IP model, it operates only on the Transport layer
    - Segmentation, sequencing, and error checking occur at the Transport layer

  - Network Layer (3)

    - Responsible for logical addressing, and providing routing or delivery guidance (but not necessarily verifying guaranteed delivery), manages error detection and traffic control
    - **routing protocols**: move routed protocol messages across a network
      - includes RIP, OSPF, IS-IS, IGRP, and BGP

      - routing protocols are defined at the Network Layer and specify how routers communicate
      - routing protocols can be static or dynamic, and categorized as interior or exterior
      - **static routing protocol**: requires an admin to create/update routes on the router
      - **dynamic**: can discover routers and determine best route to a given destination; routing table is periodically updated
      - **distance-vector**: (interior) makes routing decisions based on distance (e.g. hop count), and vector (router egress interface); examples:
        - **Routing Information Protocol (RIP)**: a distance-vector protocol that uses hop count as its routing metric
        - Interior Gateway Routing Protocol (IGRP)
        - Enhanced Interior Gateway Routing Protocol (EIGRP)
      - **link state**: (interior) uses router characteristics (e.g. speed, latency, error rates) to make next hop routing decisions; examples:
        - **Open Shortest Path First (OSPF)**: an interior gateway routing protocol developed for IP networks based on shortest path first or link-state algorithm
        - Intermediate System to Intermediate System (IS-IS)
      - **path vector**: (exterior) a type of routing protocol used to determine the best path for data to travel across networks, particularly in scenarios involving multiple autonomous systems (AS); most commonly associated with **Border Gateway Protocol (BGP)**: the primary exterior routing protocol used on the internet
      - interior vs exterior:
        - interior routing protocols ("myopic") make next hop decisions based only on info related to the next immediate hop
        - exterior routing protocols ("far-sighted") make hop decisions based on the entire remaining path (i.e.) vector
      - dive in further
    - Routed protocols include Internetwork Package Exchange (IPX) and Internet Protocol (IP)

  - Data Link Layer (2)

    - Responsible for formatting a packet for transmission
    - Adds the source and destination hardware addresses to the frame
    - **Media Access Control (MAC)**: a 6-byte (48-bit) binary address written in hex (hexidecimal notation); AKA hardware, physical, or NIC address;
      - first 3b/24-bits: Organizationally Unique Identifier (OUI) which denotes manufacturer
      - last 3b/24-bits: unique to that interface
    - ARP, switches and bridges operates at layer 2
    - Logical Link Control (LLC) is one of two sublayers that make up the Data Link Layer

  - Physical Layer (1)

    - Converts a frame into bits for transmission/receiving over the physical connection medium

        - Network hardware devices that function at layer 1 include NICs, hubs, repeaters, concentrators, amplifiers
        - Know four basic network topologies:
            - **star**: each individual node on the network is directly connect to a switch/hub/concentrator
            - **mesh**: all systems are interconnected; partial mesh can be created by adding multiple NICs or server clustering
            - **ring**: closed loop that connects end devices in a continuous ring (all communication travels in a single direction around the ring);
                - **Multistation Access Unit** (MSAU or MAU) connects individual devices
                - used in token ring and FDDI networks
            - **bus**: all devices are connected to a single cable (backbone) terminated on both ends
        - Know commonly used twisted-pair cable categories
        - Know cable types & characteristics
    - Common TCP Protocols

Port	Protocol
20,21	FTP
22	SSH
23	Telnet
25	SMTP
53	DNS
80	HTTP
110	POP3
143	IMAP
389	LDAP
443	HTTPS
445	AD, SMB
636	Secure LDAP
1433	MS SQL Server
3389	RDP
137-139	NETBIOS

- 4.1.2 Internet Protocol (IP) networking (e.g., Internet Protocol Security (IPSec), Internet Protocol (IP) v4/6)
    - IP is part of the TCP/IP (Transmission Control Protocol/Internet Protocol) suite

- TCP/IP is the name of IETF's four-layer networking model, and its protocol stack; the four layers are link (physical), internet (network-to-network), transport (channels for connection/connectionless data exchange) and application (where apps make use of network services)
- IP provides the foundation for other protocols to be able to communicate; IP itself is a connectionless protocol
- IPv4: dominant protocol that operates at layer 3; IP is responsible for addressing packets, using 32-bit addresses
- IPv6: modernization of IPv4, uses 128-bit addresses, supporting 2^128 total addresses
- TCP or UDP is used to communicate over IP
- **IP Subnetting**: method used to divide a large network into smaller, manageable pieces, called subnets
    - IP addresses: like a street address that identifies a device on a network in two parts:
        - network: identifies the "neighborhood" or network of the device
        - host: specifies the device (or "house") in that neighborhood
    - subnet masK: tool to divide the IP address into its network and host parts; e.g. 192.168.1.15 with subnet mast of 255.255.255.0 tells us that 192.168.1 is the network, and 15 is the host or device part
- **CIDR notation**: a compact way of representing IP addresses and their associated network masks
    - example: 192.168.1.0/24
        - consists of two parts:
            - IP address: 192.168.1.0 - the network or starting address
            - /24 - specifies how many bits of the IP address are used for the network part; here /24 means the first 24 bits (out of 32 for IPv4) are used for the network part, and the remaining bits are used for the host addresses in that network
        - /24 is the same as 255.255.255.0 (where again 24 bits represented by 255.255.255 define the network, and .0 defines the host range)
        - IP address range: 192.168.1.0/24 represents the network 192.168.1.0 and all IPs from 192.168.1.1 to 192.168.1.254; 2^8=256 IP address, but 254 are usable (excludes network and broadcast addresses)
    - Other examples:
        - 10.0.0.0/16: where /16 means the first 16 bits are reserved for the network, leaving 16 bits for hosts; allows 2^16 or 65,536 IP addresses, with 65,534 usable addresses
        - 172.16.0.0/12: /12 means 12 bits are for the network, leaving 20 bits for hosts; providing 2^20 = 1,048,576 IP addresses
- IPSec provides data authentication, integrity and confidentiality

            - specifically, IPsec provides encryption, access control, nonrepudiation, and message authentication using public key cryptography
    - **Logical address**: occurs when an address is assigned and used by software or a protocol rather than being provided/controlled by hardware
    - Network layer's packet header includes the source and destination IP addresses
    - Network Access Layer: defines the protocols and hardware required to deliver data across a physical network
    - Internet Layer: defines the protocols for logically transmitting packets over the network
    - Transport Layer: defines protocols for setting up the level of transmission service for applications; this layer is responsible for the reliable transmission of data and the error-free delivery of packets
    - Application Layer: defines protocols for node-to-node application communication and provides services to the application software running on a computer

- 4.1.3 Secure protocols

    - **Kerberos**: standards-based network authentication protocol, used in many products (most notably Microsoft Active Directory Domain Services or AD DS)

        - Kerberos is mostly used on LANs for organization-wide authentication, single sign-on (SSO) and authorization

    - SSL and TLS: data protection used for protecting website transactions (e.g. banking, ecommerce)

        - SSL and TLS both offer data encryption, integrity and authentication
        - TLS has supplanted SSL (the original protocol, considered legacy/insecure)
        - TLS was initially introduced in 1999 but didn't gain widespread use until years later
        - The original versions of TLS (1.0 and 1.1) are considered deprecated and organizations should be relying on TLS 1.2 or TLS 1.3
        - The defacto standard for secure web traffic is HTTP over TLS, which relies on hybrid cryptography: using asymmetric cryptography to exchange an ephemeral session key, which is then used to carry on symmetric cryptography for the remainder of the session

    - **SFTP**: a version of FTP that includes encryption and is used for transferring files between two devices (often a client / server)

    - **SSH**: remote management protocol, which operates over TCP/IP

        - all communications are encrypted
        - primarily used by IT administrators to manage devices such as servers and network devices

    - **IPSec**: an IETF standard suite of protocols that is used to connect nodes (e.g. computers or office locations) together

        - IPsec protocol standard provides a common framework for encrypting network traffic and is built into a number of common OSs
        - IPsec establishes a secure channel in either transport or tunnel mode

        - IPsec uses two protocols: Authentication Header (AH) and Encapsulating Security Payload (ESP) -- see below
        - widely used in virtual private networks (VPNs)
        - IPSec provides encryption, authentication and data integrity
        - **transport mode**: only packet payload is encrypted for peer-to-peer communication
        - **tunnel mode**: the entire packet (including header) is encrypted for gateway-to-gateway communication
        - **security association (SA)**: represents a simplex communication connection/session, recording any config and status info
        - **authentication header (AH)**: provides assurance of message integrity and nonrepudiation; also provides authentication and access control, preventing replay attacks; does not provide encryption; like an official authentication stamp, but it's not encrypted so anyone can read it
        - **encapsulating security payload (ESP)**: provides encryption of the payload which provides confidentiality and integrity of packet content; works with tunnel or transport mode; provides limited authentication and preventing replay attacks (not to the degree of AH)
    - **Internet Key Exchange (IKE)**: a standard protocol used to set up a secure and authenticated communication channel between two parties via a virtual private network (VPN); the protocol ensures security for VPN negotiation, remote host and network access

- 4.1.4 Implications of multilayer protocols

    - TCP/IP is a multilayer protocol, and derives several associated benefits
        - this means that protocols can be encapsulated within others (e.g. HTTP is encapsulated within TCP, which is in turn encapsulated in IP, which is in Ethernet), and additional security protocols can also be encapsulated in this chain (e.g. TLS between HTTP and TCP, which is HTTPS)
        - note that VPNs use encapsulation to enclose (or tunnel) one protocol inside another
    - Multilayer benefits:
        - many different protocols can be used at higher layers
        - encryption can be incorporated (at various layers)
        - it provides flexibility and resiliiency in complex networks Multilayer disadvantages:
        - nothing stops an added layer from being covert
        - encapsulating can be used to bypass filters
        - logical network segments can be traversed

- 4.1.5 Converged protocols (e.g., Fiber Channel Over Ethernet (FCoE), Internet Small Computer Sysetms Interface (iSCSI), Voice over Internet Protocol (VoIP))

    - **Converged protocols**: merged specialty or proprietary with standard protocols, such as those from the TCP/IP suite
        - converged protocols provide the ability to use existing TCP/IP supporting network infrastructure to host special or proprietary services without the need to deploy different hardware
    - Examples of converged protocols:

        - **Storage Area Network (SAN)**: a secondary network (distinct from the primary network) used to consolidate/manage various storage devices into single network-accessible storage
        - **Fibre Channel over Ethernet (FCoE)**: operating at layer 2, Fibre Channel is a network data-storage solution (SAN or network-attached storage (NAS)) that allows for high-speed file transfers of (up to) 128 Gbps
            - FCoE can be used over existing network infrastructure
            - FCoE used to encapsulate Fibre Channel over Ethernet networks
            - with this technology, Fibre Channel operates as a Network layer (OSI layer 3) protocol, replacing IP as the payload of a standard Ethernet network
        - **Internet Small Computer Sysetms Interface (iSCSI)**: operating at layer 3, iSCSI is a converged protocol, network storage standard based on IP, used to enable location-independent file storage, transmission, and retrieval over LAN, WAN, or public internet connections
        - **Multiprotocol Label Switching (MPLS)**: a WAN protocol that operates at both layer 2 and 3 and does label switching; MPLS is a high-throughput/high-performance network technology that directs data across a network based on short path labels rather than longer network addresses
        - **Voice over Internet Protocol (VoIP)**: a tunneling mechanism that encapsulates audio, video, and other data into IP packets to support voice calls and multimedia collab
            - VoIP is considered a converged protocol because it combines audio and video encapsulation technology (operating as application layer protocols) with the protocol stack of TCP/IP
            - SIPS and SRTP are used to secure VoIP
            - **Secure Real-Time Transport Protocol (SRTP)**: an extension profile of RTP (Real-Time Transport Protocol) which adds further security features, such as message authentication, confidentiality and replay protection mostly intended for VoIP communications
            - SIPS: see definition above Other converged protocols:
            - SDN (see definition above)
            - cloud
            - virtualization (see definition in Domain 3)
            - SOA (see definition in Domain 3)
            - microservices (see definition in Domain 3)
            - IaC (see definition in Domain 8)
            - serverless architecture (see definition in Domain 3)

- 4.1.6 Micro-segmentation (e.g., Software Defined Networks (SDN), Virtual eXtensible Local Area Network (VXLAN),Encapsulation, Software-Defined Wide Area Network (SD-WAN))

    - **Software-defined networks (SDN)**:

        - SDN is a broad range of techniques enabling network management, routing, forwarding, and control functions to be directed by software
        - SDN is effectively network virtualization, and separates the infrastructure layer (aka the data or forwarding plane) - hardware and hardware-based settings, from the control layer - network services of data transmission management

        - NOTE:
            - **control plane**: receives instructions and sends them to the network; uses protocols to decide where to send traffic
            - **data plane**: includes rules that decide whether traffic will be forwarded
            - **application plane**: where applications run that use APIs to communicate with the SDN about needed resources
    - typically ABAC-based
    - an SDN solution provides the option to handle traffic routing using simpler network devices that accept instructions from the SDN controller
    - SDN offers a network design that is directly programmable from a central location, is flexible, vendor neutral, and based on open standards
    - Allows org to mix/match hardware
  - **Virtual extensible local area network (VXLAN)**:
    - an encapsulation protocol that enables VLANs to be stretched across subnets and geographic distances
      - VLANs allow network admins to use switches to create software-based LAN segments that can be defined based on factors other than physical location
    - VLANs are typically restricted to layer 2, but VXLAN tunnels layer 2 connections over a layer 3 network, stretching them across the underlying layer 2 network
    - Allows up to 16 million virtual networks (VLAN limit is 4096)
    - VXLAN can be used as a means to implement microsegmentation without limiting segments to local entities only
    - Defined in RFC 7348
  - Encapsulation:
    - the OSI model represents a protocol stack, or a layered collection of multiple protocols, and communication between protocol layers occurs via encapsulation and deencapsulation (defined above)
  - **Software-defined wide area network (SD-WAN/SDWAN)**: an evolution of SDN that can be used to manage the connectivity and control services between distant data centers, remote locations, and cloud services over WAN links; put another way, SDN-WAN is an extension of SDN practices to connect entities spread across the internet, supporing WAN architecture; espcially related to cloud migration
  - SDWANs are commonly used to manage multiple ISP, and other connectivity options for speed, reliability, and bandwidth design goals
  - **Software-defined Visibility (SDV)**: a framework to automate the processes of network monitoring and response; the goal is to enable the analysis of every packet and make deep intelligence-based decisions on forwarding, dropping, or otherwise responding to threats
- 4.1.7 Wireless networks (e.g. LiFi, Wi-Fi, Zigbee, satellite)
  - **Narrowband**: refers to a communication channel or system that operates with a small bandwidth, meaning it uses a limited range of frequencies to transmit data; in contrast to broadband, which can carry large amounts of data over a wide frequency range, narrowband

systems focus on efficient transmission of smaller amounts of data, often over long distances, by using lower data rates and narrower frequency bands

- **light fidelity (Li-Fi)**: a form of wireless communication technology that relies on light to transmit data, with theorectical speeds up to 224Gbits/sec
- **Radio Frequency Identification (RFID)**: a technology used to identify and track objects or individuals using radio waves, with two main components: an RFID tag (or transponder) and an RFID reader; the tag contains a small microchip and an antenna, and the reader emits a signal that communicates with the tag to retrieve the stored information
    - Passive Tags don't have their own power source, relying instead on the energy from the RFID reader's signal to transmit data
    - Active Tags have a battery and can broadcast signals over longer distances
- **Near Field Communicatio (NFC)**: a wireless communication technology that allows devices to exchange data over short distances, usually within a range of about 4 centimeters (1.5 inches); it operates on the same principle as RFID but is designed for closer proximity communication and is commonly used in mobile devices for tasks like contactless payments and data sharing; unlike RFID, where only the reader actively sends signals, NFC enables two-way communication
    - Active Mode: both devices generate their own radio frequency signals to communicate
    - Passive Mode: one device (like an NFC tag) is passive and only transmits data when powered by the active device's signal, similar to how passive RFID tags work
- **Bluetooth**: wireless personal area network, IEEE 802.15; an open standard for short-range RF communication used primarily with wireless personal area networks (WPANs); secure guidelines:
    - use Bluetooth only for non-confidential activities
    - change default PIN
    - turn off discovery mode
    - turn off Bluetooth when not in active use
- **Wi-Fi**: Wireless LAN IEEE 802.11x; associated with computer networking, Wi-Fi uses 802.11x spec to create a public or private wireless LAN
    - **Wired Equivalent Privacy (WEP)**:
        - WEP is defined by the original IEEE 802.11 standard
        - WEP uses a predefined shared Rivest Cipher 4 (RC4) secret key for both authentication (SKA) and encryption
        - Shared key is static
        - WEP is weak from RC4 flaws
    - Wi-Fi Protected Access II (WPA2):
        - IEEE 802.11i WPA2 replaced WEP and WPA
        - Uses AES-CCMP (Counter Mode with Cipher Block Chaining Message Authentication Code Protocol)

        - WPA2 operates in two modes, personal and enterprise
            - personal mode or the Pre-Shared Key (PSK) relies on a shared passcode or key known to both the access point and the client device; typically used for home network security
            - enterprise mode uses the more advanced Extensible Authentication Protocol (EAP) and an authentication server and individual credentials for each user or device; enterprise mode is best suited to companies and businesses

    - Wi-Fi Protected Access 3 (WPA3):

        - WPA3-ENT uses 192-bit AES CCMP encryption
        - WPA3-PER remains at 128-bit AES CCMP
        - WPA3 **simultaneous authentication of equals (SAE)**: a mode improves on WPA2's PSK mode by allowing for secure authentication between clients and the wireless network without enterprise user accounts; SAE performs a zero-knowledge proof process known as **Dragonfly Key Exchange** (which is a derivative of Diffie-Hellman); SAE uses a preset password and the MAC addresses of the client and AP to perform authentication and session key exchange

    - 802.1X / EAP

        - WPA, WPA2, and WPA3 support the enterprise (ENT) authentication known as 802.1X/EAP (requires user accounts)
        - Extensible Authentication Protocol (EAP) is not a specific mechanism of authentication, rather an authentication framework
        - 802.1X/EAP is a standard port-based network access control that ensures that clients cannot communicate with a resource until proper authentication has taken place
        - Through the use of 802.1X Remote Authentication Dial-In User Service (RADIUS), Terminal Access Control Access Control System (TACACS), certificates, smartcards, token devices and biometrics can be integrated into wireless networks
        - Don’t forget about ports related to common AAA services:
            - UDP 1812 for RADIUS
            - TCP 49 for TACACS+

    - **Service Set Identifier (SSID)**: the name of a wireless network that is broadcast by a Wi-Fi router or access point, and used to uniquely identify a wireless network, so devices can recognize and connect to it; when you search for Wi-Fi networks on your phone or computer, the list of available networks you see consists of their SSIDs

        - **Extended Service Set Identifier (ESSID)**: the name of a wireless network (Wi-Fi network) that users see when they search for available networks, identifying the extended service set, which is essentially a group of one or more access points (APs) that form a wireless network; multiple APs in the same network can share the same ESSID, allowing seamless roaming for users within the network coverage area

    - **Basic Service Set Identifier (BSSID)**: a unique identifier for each AP in a Wi-Fi network; it's the MAC address of the individual wireless access point or router within the network; while multiple APs in a network can share the same ESSID, each AP will have its own unique BSSID to distinguish it from other APs

- **Site survey**: a formal assessment of wireles signal strnegth, quality, and interference using an RF signal detector

- **Wi-Fi Protected Setup (WPS)**: intended to simplify the effort of setting up/adding new clients to a secured wireless network; operates by automatically connecting the first new wireless client to seek the network once WPS is triggered

    - WPS allows users to easily connect devices to a Wi-Fi network by:
        - pressing a physical WPS button on the router
        - entering an 8-digit PIN found on the router
        - using NFC or Push-Button Connect for quick device pairing
        - the 8-digit PIN method is vulnerable to attacks, particularly brute-force, due to the structure of the WPS protocol, since the PIN is validated in two halves; also many routers do not implement rate limiting allowing repeated PIN attempts without lock out
    - Best WPS protection is to turn it off

- Lightweight Extensible Authentication Protocol (LEAP) is a Cisco proprietary alternative to TKIP for WPA

    - Avoid using LEAP, use EAP-TLS as an alternative; if LEAP must be used a complex password is recommended

- Protected Extensible Authentication Protocol (PEAP): a security protocol used to better secure WiFi networks; PEAP is protected EAP, and it comes with enhanced security protections by providing encryption for EAP methods, and can also provide autentication; PEAP encapsulates EAP within an encrypted TLS (Transport Layer Security) tunnel, thus encrypting any EAP traffic that is being sent across a network

- EAP Methods

Method	Type	Auth	Creds	When to Use
EAP-MD5	Non-Tunnel	Challenge/response with hashing	Passwords for client auth	Avoid
EAP-MSCHAP	Non-Tunnel	Challenge/response with hashing	Passwords for client auth	Avoid
EAP-TLS	Non-Tunnel	Challenge/response with public key cryptography	Digital certificates for client/server auth	To support digitial certs as client/server creds
EAP-GTC	Non-Tunnel	Cleartext pass	Passwords/OTP for client auth	Only use inside PEAP or EAP-FAST

Method	Type	Auth	Creds	When to Use
PEAP	Tunnel	Challenge/response with public key cryptography	Digital certificates for server auth	Digital certs as server creds, and TLS secure channel for inner EAP methods
EAP-FAST	Tunnel	Challenge/response with symmetric cryptography (PAC)	DS, PAC for auth, other inside EAP tunnel	To support digital certs as server creds, and TLS secure channel for inner EAP; to support EAP chaining

    - Speed and frequency table:

Amendment	Wi-Fi Alliance	Speed	Frequency
802.11	--	2 Mbps	2.4 GHz
802.11a	Wi-Fi 2	54 Mbps	5 GHz
802.11b	Wi-Fi 1	11 Mbps	2.4 GHz
802.11g	Wi-Fi 3	54 Mbps	2.4 GHz
802.11n	Wi-Fi 4	200+ Mbps	2.4,5 GHz
802.11ac	Wi-Fi 5	1 Gbps	5 GHz
802.11ax	Wi-Fi 6/Wi-Fi 6E	9.5 Gbps	2.4,5,6 GHz
802.11be	Wi-Fi 7	46 Gbps	2.4,5,6 GHz

  - Modes:
    - Ad hoc: directly connects two clients
    - Standalone: connects clients using a WAP, but not to wired resources
    - Infrastructure: connects endpoints to a central network, not to each other
    - Wired extension: uses a wireless access point to link wireless clients to a wired network
  - Wireless antennas: when setting up a wireless network, the type of antenna used on both the wireless client (device trying to connect) and the base station (such as an access point or router) is important for optimizing signal strength and coverage; different antennas are used depending on the needs of the environment, and these antennas vary in terms of their directionality and range

Antenna Type	Directionality	Range	Use Case
Omnidirectional Pole	360-degree (all around)	Short to medium	General-purpose Wi-Fi coverage in homes/offices
Yagi Antenna	Highly directional	Long	Long-distance links between buildings or to distant devices

Antenna Type	Directionality	Range	Use Case
Cantenna	Directional	Medium to long (DIY solutions)	Extending Wi-Fi to a distant access point
Panel Antenna	Semi-directional	Medium	Indoor/outdoor targeted coverage in one direction
Parabolic Antenna	Extremely directional	Very long	Point-to-point communication over miles

  - **Zigbee**: IoT equipment communications concept based on Bluetooth
    - Low power/low throughput
    - Requires close proximity
    - Encrypted using 128-bit symmetric algorithm
    - Zigbee uses AES to protect network traffic, providing integrity and confidentiality controls
  - **Satellite**: primarily uses radio waves between terrestrial locations and an orbiting artificial satellite
    - Supports telephone, tv, radio, internet, military communications
    - 3 primary orbits:
      - LEO: low Earth orbit (160-2k km)
        - have stronger signals
        - multiple devices needed to maintain coverage (e.g. Starlink)
      - MEO: medium Earth orbit (2k-35768 km)
        - above a terrestrial location longer than LEO
        - higher orbit, additional delay/weaker signal
      - GEO: geostationary orbit (35768 km)
        - maintain a fixed position above a terrestrial location, and ground stations can use fixed antennas
        - larger transmission footprint than MEO, but higher latency

- 4.1.8 Cellular networks (e.g. 4G, 5G)
  - A cellular network or a wireless network is the primary communications technology used by many mobile devices
  - Cells are primary transceiver (cell site/tower)
  - Generally encrypted between mobile device and transmission tower; plaintext over wire; use encryption like TLS/VPN
  - 4G
    - 4G allows for mobile devices to achieve 100 Mbps, and stationary devices can reach 1 Gbps
    - LTE and WiMAX are common transmission systems
    - **WiMAX**: Broadband Wireless Access IEEE 802.16 is a well-known example of wireless broadband; WiMAX can potentially deliver data rates of > 30 Mbps
  - 5G

        - 5G uses higher frequencies than previous tech, allowing for higher transmission speeds up to 10 Gbps, but at reduced distances
        - Orgs need to enforce security requirements on 5G
        - 5G advantages over 4G
            - enhanced subscriber identity protection
            - mutual authentication capabilities
    - Security issues with wireless:
        - provider network (voice or data) is not necessarily secure
        - your cell phone can be intercepted
        - provider's towers can be simulated to conduct man-in-the-middle/on-path attack
        - using cell connectivity to access the internet or your office network creates a potential bridge, provider attackers with another avenue

- 4.1.9 Content Distribution Networks (CDN)

    - **Content Distribution Network (CDN)**: a collection of resource services deployed in numerous data centers across the internet in order to provide low latency, high performance, and high availability of the hosted content
        - CDNs provide multimedia performance quality through the concept of distributed data hosts, geographically distributed, closer to groups of customers
        - Provides geographic and logical load balancing; lower-latency and higher-quality throughput
    - Client-based CDN is often referred to as P2P (peer-to-peer)

4.2 Secure network components (OSG-9 Chpt 11)

The components of a network make up the backbone of the logical infrastructure for an organization; these components are often critical to day-to-day operations, and an outage or security issue can be very costly

- 4.2.1 Operation of hardware (e.g. redundant, power, warranty, support)

    - Modems provide modulation/demodulation of binary data into analog signals for transmission; modems are a type of Channel Service Unit/Data Service Unit (CSU/DSU) typically used for converting analog signals into digital; the CSU handles communication to the provider network, the DSU handles communication with the internal digital equipment (in most cases, a router)

        - modems typically operate at Layer 2
        - routers operate at Layer 3, and make the connection from a modem available to multiple devices in a network, including switches, access points and endpoint devices
        - switches are typically connected to a router to enable multiple devices to use the connection
        - switches help provide internal connectivity, as well as create separate broadcast domains when configured with VLANs
        - switches typically operate at Layer 2 of the OSI model, but many switches can operate at both Layer 2 and Layer 3
        - access points can be configured in the network topology to provide wireless access using one of the protocols and encryption algorithms

- Redundant power: most home equipment use a single power supply, if that supply fails, the device loses power
    - redundant power is typically used with components such as servers, routers, and firewalls
    - redundant power is usually paired with other types of redundancies to provide high availability

- 4.2.2 Transmission media
  - Transmission Media: comes in many forms, not just cables
    - includes wireless, LiFi, Bluetooth, Zigbee, satellites
    - most common cause of network failure (i.e. violations of availability) are cable failures or misconfigurations
    - wired transmission media can typically be described in three categories: coaxial, Ethernet, fiber
    - coaxial is typically used with cable modem installations to provide connectivity to an ISP, and requires a modem to convert the analog signals to digital
      - fairly resistent to EMI
      - longer lengths than twisted pair
      - requires segment terminators
      - two main types:
        - **thinnet (10Base2)**: used to connect systems to backbond trunks of thicknet cabling (185m, 10Mbps)
        - **thicknet (10Base5)**: can span 500 meters and provide up to 10Mbps
    - ethernet can be used to describe many mediums, it is typically associated with Category 5/6 unshielded twisted-pair (UTP) or shielded twisted pair (STP), and can be plenum-rated
    - fiber typically comes in two options: single-mode or multi-mode
      - Single-mode is typically used for long-distance communication, over several kilometers or miles
      - Multi-mode fiber is typically used for faster transmission, but with a distance limit depending on the desired speed
      - Fiber is most often used in the datacenter for backend components

Category	Throughput	Notes
Cat 1	1 Mbps	
Cat 2	4 Mbps	
Cat 3	10 Mbps	
Cat 4	16 Mbps	
Cat 5	100 Mbps	
Cat 5e	1 Gbps	
Cat 6	1 Gbps	

Category	Throughput	Notes
Cat 6a	10 Gbps	
Cat 7	10 Gbps	
Cat 8	40 Gbps	

- 4.2.3 Network Access Control (NAC) devices
  - **Network Access Control (NAC)**: the concept of controlling access to an environment through strict adherence to and enforcement of security policy
  - NAC is meant to be an automated detection and response system that can react in real time, ensuring all monitored systems are patched/updated and have current security configurations, as well as keep unauthorized devices out of the network
  - NAC goals:
    - prevent/reduce known attacks directly (and zero-day indirectly)
    - enforce security policy throughout the network
    - use identities to perform access control
  - NAC can be implemented with a preadmission or postadmission philosophy:
    - **preadmission philosohpy**: requires a system to meet all current security requirements (such as patch application and malware scanner updates) before it is allowed to communicate with the network
    - **postadmission philosophy**: allows and denies access based on user activity, which is based on a predefined authorization matrix
  - Agent-based NAC:
    - installed on each management system, checks config files regularly, and can quarantine for non-compliance
    - dissolvable: usually written in a web/mobile language and is executed on each local machine when the specific management web page is accessed (such as captive portal)
    - permanent: installed on the monitored system as a persistent background service
  - Agentless NAC: no software is installed on the endpoint, instead, the NAC system performs security checks using existing network infrastructure, such as switches, routers, firewalls, and network protocols; it gathers information about the device passively or actively through scans, without requiring direct interaction with the endpoint
  - NAC posture assessment capability determines if a system is sufficiently secure and compliant to connect to the network; this is a form of risk-based access control

Feature	Agent-Based NAC	Agentless NAC
Software Requirement	Requires agent installation on devices	No software installation required on devices

Feature	Agent-Based NAC	Agentless NAC
Depth of Security Checks	Provides deep insight into device security posture (antivirus, OS, patches)	Provides basic information (device type, MAC, OS)
Continuous Monitoring	Yes, can perform continuous monitoring after network access	Typically performs one-time or periodic checks
Device Compatibility	May not support unmanaged devices or IoT devices	Works with all devices (IoT, printers, guest devices)
Deployment Complexity	More complex due to agent installation and management	Easier to deploy, no software installation required
Granular Control	Offers granular control over security policies	Limited control, focuses on basic compliance
Remediation Capabilities	Can help remediate non-compliant devices (e.g., installing patches)	Limited or no remediation capabilities

- Just as you need to control physical access to equipment and wiring, you need to use logical controls to protect a network; there are a variety of devices that provide this type of protection, including:
    - stateful and stateless firewalls can perform inspection of the network packets and use rules, signatures and patterns to determine whether the packet should be delivered
        - reasons for dropping a packet could include addresses that don't exist on the network, ports or addresses that are blocked, or the content of the packet (e.g. malicious packets blocked by admin policy)
    - IDP devices, which monitor the network for unusual network traffic and MAC or IP address spoofing, and then either alert on or actively stop this type of traffic
    - proxy server information:
        - **proxy server**: used to mediate between clients and servers, most often in the context of providing cleints on a private network with internet access, while protecting the identify of the client
        - **forward proxy**: usually used by clients to anonymize their traffic, improve privacy, and cache data; a forward proxy is configured on client-side devices to manage access to external resources
        - **reverse proxy**: usually positioned in front of servers to distribute incoming traffic, improve performance through load balancing, and enhance security by hiding the details of backend servers; reverse proxies are often deployed to a perimeter network; they proxy communication from the internet to an internal host, such as a web server
        - **transparent proxy**: operates without client configuration and intercepts traffic transparently, often for monitoring or content filtering purposes without altering the client's perception of the connection

- **nontransparent proxy**: requires explicit configuration on the client side and may modify traffic to enforce policies, such as restricting access or logging user activities

Attribute	Forward Proxy	Reverse Proxy	Transparent Proxy	Nontransparent Proxy
Primary Function	Acts as an intermediary between client and internet	Acts as an intermediary between client and backend servers	Intercepts client requests without modifying them	Requires explicit client configuration
Client Awareness	Client is aware of proxy usage	Client is unaware of proxy usage	Client is unaware of proxy usage	Client is aware of proxy usage
Use Case	Content filtering, privacy, and caching for users	Load balancing, security, and hiding server identity	Caching, content filtering without client configuration	Content filtering, security, and logging
Configuration	Configured on client devices or network settings	Configured on the server side	No configuration needed on the client side	Requires configuration on the client side
Visibility	Proxy IP address is visible to the target website	Proxy hides server IP address from the client	Proxy operation is invisible to both client and server	Proxy server IP address is visible to the client
Modification of Requests	Can modify or filter client requests	Can modify server responses or requests from clients	Does not modify requests or responses	Can modify or filter client requests
Security Benefits	Provides privacy by hiding client IP addresses	Provides security by hiding server details, load balancing	Limited security, primarily used for convenience	High security potential, especially for monitoring

- 4.2.4 Endpoint security

    - **Endpoint security**: each individual device must maintain local security whether or not its network or telecom channels also provide security
        - any weakness in a network, whether border, server, or client-based presents a risk to all elements of the org
        - client/Server model is distributed architecture, meaning that security must be addressed everywhere instead of at a single centralized host
        - processing, storage on clients and servers, network links, communication equipment all must be secured
        - clients must be subjected to policies that impose safeguards on their content and users' activities including:
            - email
            - upload/download policies and screening
            - subject to robust access controls (e.g. MFA)
            - file encryption
            - screen savers
            - isolated processes for user/supervisor modes
            - local files should be backed up
            - protection domains/network segments
            - security awareness training
            - desktop env should be included in org DR
            - EDR/MDR should be considered

4.3 Implement secure communication channels according to design ((OSG-9 Chpt 12))

- Protocols that provide security services for application-specific communication channels are called secure communication protocols
    - examples of secure communication protocols include: IPsec, Kerberos, SSH, Signal protocol, S-RPC, and TLS
- 4.3.1 Voice
    - **Voice over Internet Protocol (VoIP)**: set of technologies that enables voice to be sent over a packet network
    - As more orgs switch to VoIP, protocols like SIP become more common, and introducing additional management, either via dedicated voice VLANs, or by establishing quality of service (QoS) levels to ensure voice traffic priority
    - Web-based voice apps can be more difficult to manage, causing additional unplanned bandwidth consumption
- 4.3.2 Multimedia collaboration
    - There are a variety of new technologies that allow instant organizational collaboration, including smartboards, and products that enhance on-site, hybrid, or virutal meetings
    - Mobile communication apps are a huge market, and will continue to grow, increasing the complexity of mobile security
- 4.3.3 Remote access
    - 4 main types of remote access:
        - **service specific**: gives users the ability to remotely connect to and manipulate or interact with a single service (e.g. email)
        - **remote-control**: grants a remote user the ability to fully control another system that is physically distant

        - **remote node operation**: AKA remote client connecting directly to a LAN
        - **screen scraping**: refers to 1) remote control, remote access, or remote desktop services or 2) technology that allows an automated tool to interact with a human interface
    - VPN (virtual private network) is a traditional remote access technology
        - most common VPN protocols: PPTP, L2F, L2TP, and IPsec
    - WAP (wireless access point) - local env treats as remote access
    - **VDI (virtual desktop infrastructure)**: means of reducing the security risks and performance requirements of end devices by hosting desktop/workstation VMs on servers that are remotely accessible by users
    - **VMI (virtual mobile interface)**: virtual mobile device OS is hosted on a central server
    - **Jumpbox**: a jump server/jumpbox is a remote access system deployed to make accessing a specific system or network easier or more secure; often deployed in extranets, screened subnets, or cloud networks where a standard direct link or private channel is not available
    - RDS (Remote Desktop Service) such as RD, Teamviewer, VNC etc can provide in-office experience while remote
    - Using cloud-based desktop solutions such as Amazon Workspaces, Amazon AppStream, V2 Cloud, and Microsoft Azure
    - Security must be considered to provide protection for your private network against remote access complications:
        - stringent auth before granting access
        - grant permission only for specific need
        - remote comm protected via encryption
    - Create a remote access security policy, addressing:
        - remote connectivity technology
        - transmission protection
        - authentication protection
        - remote user assistance
- 4.3.4 Data communications
    - Whether workers are physically in an office or working remotely, communication between devices should be encrypted to prevent any unauthorized device or person from openly reading the contents of packets as they are sent across a network
    - Corporate networks can be segmented into multiple VLANs to separate different types of resources
    - Communications should be encrypted using TLS or IPSec
- 4.3.5 Virtualized networks
    - **Virtualized network**: is the combination of hardware and software networking components into a single integrated solution; allows for software control over network functions such as management, traffic shaping, address assignment etc
    - Allows for adoptiong of things like software-defined networks (SDNs), VLANs, virtual switches, virtual SANs, guest OSs, port isolation etc
    - Many orgs are moving to the cloud, and not continuing to build out local or on-site server infrastructure
        - however, organizations still use hypervisors to virtualize servers and desktops for increased density and reliability
            - to host multiple servers on a single hypervisor, the Ethernet and storage networks must also be virtualized

        - VMware vSphere and Microsoft Hyper-V both use virtual network and storage switches to allow communication between virtual machines and the physical network; guest OSs running in the VMs use a synthetic network or storage adapter, which is relayed to the physical adapter on the host
        - SDN on the hypervisor can control the VLANs, port isolation, bandwidth and other aspects just as if it was a physical port
- 4.3.6 Third-party connectivity
    - Any time an org's network is connected directly to another entity's network, their local threats and risks affect each other
        - **memorandum of understanding (MOU)** or **memorandum of agreement (MOA)**: (Note: MOU = letter of intent) an expression of agreement or aligned intent, will, or purpose between two entities
        - **interconnection security agreement (ISA)**: a formal declaration of the security stance, risk, and technical requirements of a link between two organizations' IT infrastructures
    - Remote workers are another form of third-party connectivity
    - Vendors (like IT auditing firms) may need to connect to your network, and attackers are routinely looking for creative ways to gain organizational access -- third-party connectivity is one option
    - As organizations evaluate third-party connectivity, they need to look carefully at the principle of least privilege and at methods of monitoring use and misuse

Domain 5 **Identity and Access Management (IAM)**

The identity and Access Management (IAM) domain focuses on issues related to granting and revoking privileges to access data or perform actions on systems

- Assets include information, systems, devices, facilities, and applications
- Organizations use both physical and logical access controls to protect them
- Identification is the process of a subject claiming, or professing, an identity
- Authentication verifies the subject's identity by comparing one or more authentication factors against a database holding authentication info for users
- The three primary authentication factors are something you know, something you have, and something you are
    - 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, includes 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)
- Single sign-on (SSO) technologies allow users to authenticate once and access any resources in a network or the cloud, without authenticating again
- Federated Identity Management (FIM) systems link user identities in one system with other systems to implement SSO
- **Access Control System**: ensuring access to assets is authorized and restricted based on business and security requirements

- **Access Control Token**: based on the parameters like time, date, day etc a token defines access validity to a system
- **ADFS**: identity access solution that provides client computers (internal or external to your network) with seamless SSO access to protected Internet-facing applications or services, even when the user accounts and applications are located in completely different networks or orgs
- **Capability tables**: list privileges assigned to subjects and identify the objects that subjects can access
- **CAS**: Central Authentication Service (an SSO implementation)
- **Content-dependent control**: Content-dependent access control adds additional criteria beyond identification and authentication: the actual content the subject is attempting to access; all employees of an org may have access to the HR database to view their accrued sick time and vacation time, but should an employee attempt to access the content of the CIO's HR record, access is denied
- **Context-dependent access control**: applies additional context before granting access, with time as a commonly used context
- **Crossover Error Rate** (CER): point at which false acceptance (Type 2) error rate equals the false rejection (Type 1) error rate for a given sensor, in a given system and context; it is the optimal point of operation if the potential impacts of both types of errors are equivalent
- **Cross-Site Request Forgery (CSRF)**: (AKA XSRF) an attack that forces authenticated users to submit a request to a Web application against which they are currently authenticated
- **FRR**: False Rejection Rate (Type 1) incorrectly denying authentication to a legit identity and therefore denying access
- **FAR**: False Acceptance Rate (Type 2) incorrectly authenticating a claimed identity as legit, recognizing and granting access on that basis
- **Ethical Wall**: the use of administrative, physical/logical controls to establish/enforce separation of information, assets or job functions for need-to-know boundaries or prevent conflict of interest situations; AKA compartmentalization
- **Granularity of controls**: level of abstraction or detail which in a security function can be configured or tuned for performance and sensitivity
- **IDaaS**: cloud-based service that broker IAM functions to target systems on customers' premise and/or in the cloud
- **Identity proofing**: process of collecting/verifying info about someone who has requested access/credential/special privilege to establish a relationship with that person
- **Self-service identity management**: elements of the identity management lifecycle which the end-user (identity in question) can initiate or perform on their own (e.g. password reset, changes to challenge questions etc)
- **Whaling attack**: phishing attack targeting highly-placed officials/private individuals with sizeable assets authorizing large-fund wire transfers
- **XSS**: Cross-Site Scripting (XSS) essentially uses reflected input to trick a user's browser into executing untrusted code from a trusted site; these attacks are a type of injection, in which malicious scripts are injected into otherwise benign and trusted websites; XSS attacks occur when an attacker uses a web app to send malicious code, generally in the form of a browser side script, to a different end user; flaws that allow these attacks to succeed are quite widespread and occur anywhere a web application uses input from a user within the output it generates without validating or encoding it
- **XST**: Cross-Site Tracing (XST) attack involves the use of Cross-site Scripting (XSS) and the TRACE or TRACK HTTP methods; this could potentially allow the attacker to steal a user's cookies

5.1 Control physical and logical access to assets (OSG-9 Chpt 13)

- Controlling access to assets (tangible: things you can touch, or nontangible: info and data) is a central theme of security
- Understand that there is no security without physical security: admin, technical and logical access controls aren't effective without control over the physical env
- In addition to personnel, assets can be information, systems, devices, facilities, or applications:
    - 5.1.1 Information: an org’s information includes all of its data, stored in simple files (on servers, computers, and small devices), or in databases
    - 5.1.2 Systems: an org’s systems include anything that provide one or more services; a web server with a database is a system; permissions assigned to user and system accounts control system access
    - 5.1.3 Devices: refers to any computing system (e.g. routers & switches, smartphones, laptops, and printers); BYOD has been increasingly adopted, and the data stored on the devices is still an asset to the org
    - 5.1.4 Facilities: any physical location, building, rooms, complexes etc; physical security controls are important to help protect facilities
    - 5.1.5 Applications: apps provide access to data; permissions are an easy way to restrict logical access to apps
- Understand what assets you have, and how to protect them
    - **physical security controls**: such as perimeter security and environmental controls
        - control access and the environment
    - **logical access controls**: automated systems that auth or deny access based on verification that identify presented matches that which was previously approved; technical controls used to protect access to information, systems, devices, and applications
        - includes authentication, authorization, and permissions
        - permissions help ensure only authorized entities can access data
        - logical controls restrict access to config settings on systems/networks to only authed individuals
        - applies to on-prem and cloud

5.2 Manage identification and authentication of people, devices, and services (OSG-9 Chpt 13)

- **Identification**: the process of a subject claiming, or professing an identity
- **Authentication**: verifies the subject’s identity by comparing one or more factors against a database of valid identities, such as user accounts
    - a core principle with authentication is that all subjects must have unique identities
    - identification and authentication occur together as a single two-step process
    - users identify themselves with usernames and authenticate (or prove their identity) with passwords
- 5.2.1 Identiy management (IdM) implementation
    - Identity and access management is a collection of processes and techologies that are used to control access to critical assets; it's purpose is the management of access to information, systems, devices, and facilities
    - Identity Management (IdM) implementation techniques generally fall into two categories:
        - **centralized access control**: implies a single entity within a system performs all authorization verification

            - potentially creates a single point of failure
            - small team can manage initially, and can scale to more users
        - **decentralized access control**: (AKA distributed access control) implies several entities located throughout a system perform auth verification
            - requires more individuals or teams to manage, and admin may be spred across numerous locations
            - difficult to maintain consistency
            - changes made to any individual access control point needs to be repeated at others
    - With ubiquitious mobile computing and anywhere, anytime access (to apps & data), identity is the "new perimeter"
- 5.2.2 Single/Multi-Factor Authentication (MFA)
    - **Single-factor authentication**: any authentication using only one proof of identity
    - **Two-factor authentication (2FA)**: requires two different proofs of identity
    - **Multifactor authentication (MFA)**: any authentication using two or more factors
        - multifactor auth must use multiple types or factors, such as something you know and something you have
        - note: requiring users to enter a password and a PIN is NOT multifactor (both are something you know)
    - Two-factor methods:
        - **Hash Message Authentication Code (HMAC)**: includes a hash function used by the HMAC-based One-Time Password (HOTP) standard to create onetime passwords
        - **Time-based One-Time Password (TOTP)**: similar to HOTP, but uses a timestamp and remains valid for a certain time frame (e.g. 30 or 60 seconds)
            - e.g. phone-based authenticator app, where your phone is mimicking a hardware TOTP token (combined with userid/password is considered two-factor or two-step authentication)
        - **Email challenge**: popular method, used by websites, sending the user an email with a PIN
        - Short Message Service (SMS): to send users a text with a PIN is another 2-factor method; note that NIST SP 800-63B points out vulnerabilities, and deprecates use of SMS as a two-factor method for federal agencies
- 5.2.3 Accountability
    - Two important security elements in an access control system are authorization and accountability
        - **Authorization**: subjects are granted access to objects based on proven identities
        - **Accountability**: users and other subjects can be held accountable for their actions when auditing is implemented
    - **Auditing**: tracks subjects and records when they access objects, creating an audit trail in one or more audit logs
    - Auditing provides accountability
- 5.2.4 Session management
    - Session management is important to use with any type of authentication system to prevent unauthorized access
    - Desktop/laptops: recommendation to use screensavers, although modern OSs have timeout/lock features

  - Secure online sessions should terminate after a timeout period
  - The Open Web Application Security Project (OWASP) publishes “cheat sheets” that provide app developer’s specific recommendations
- 5.2.5 Registration, proofing, and establishment of identity
  - Within an organization, new employees prove their identity with appropriate documentation during the hiring process
    - in-person identity proofing includes things like passport, DL, birth cert etc
  - Online orgs often use knowledge-based authentication (KBA) for identity-proofing of someone new (e.g. a new customer creating a new bank/savings account)
    - example questions include past vehicle purchases, amount of mortgage payment, previous addresses, DL numbers
    - they then query authoritative information (e.g. credit bureaus or gov agencies) for matches
  - Cognitive Passwords: security questions that are gathered during account creation, which are later used as questions for authentication (e.g. name of pet, color of first car etc)
    - one of the flaws associated with cognitive passwords is that the information is often available on social media sites or general internet searches
- 5.2.6 Federated Identity Management (FIM)
  - Federated Identity Management (FIM) systems (a form of SSO) are often used by cloud-based apps
  - A federated identity links a user’s identity in one system with multiple identity management systems
  - FIM allows multiple orgs to join a federation or group, agreeing to share identity information
    - users in each org can log in once in their own org, and their credentials are matched with a federated identity
    - users can then use this federated identity to access resources in any other org within the group
    - where each organization decides what resources to share
  - Methods used to implement federated identity management systems include:
    - Security Assertion Markup Language (SAML)
    - OAuth
    - OpenID Connect (OIDC)
  - Cloud-based federation typically uses a third-party service to share federated identities
  - Federated identity management systems can be hosted on-premises, in the cloud, or in a combination of the two as a hybrid system
- 5.2.7 Credential management systems
  - **Credential management systems**: provide storage space for usernames and password
    - e.g. web browsers that remember usernames and passwords for visited sites
  - The World Wide Web Consortium (W3C) published the Credential Management Level 1 API as a working draft in January 2019, which many browsers have adopted
  - Some federated identity management solutions use the Credential Management API, allowing web apps to implement SSO using a federated identity provider
    - e.g. using your Google or Facebook account to sign into Zoom
- 5.2.8 Singe Sign On (SSO)
  - **Single Sign-On (SSO)**: a centralized access control technique allowing a subject to be authenticated once on a system and access multiple resources without authenticating again

- Advantages of using SSO include:
  - reduces the number of passwords that users need to remember, and they are less likely to write them down
  - eases administration by reducing the number of accounts
- Disadvantages:
  - once an account is compromised, an attacker gains unrestricted access to all of the authorized resources
- Within an organization, a central access control system, such as a directory service, is often used for SSO
  - **directory service**: a centralized database that includes information about subjects and objects, including authentication data
  - many directory services are based on the Lightweight Directory Access Protocol (LDAP)

- 5.2.9 Just-In_time (JIT)
  - Federated identity solutions that support just-in-time (JIT) provisioning automatically create the relationship between two entities so that new users can access resources
  - A JIT solution creates the connection without any administrative intervention
  - JIT systems commonly use SAML to exchange required data

5.3 Federated Identity with a third-party service (OSG-9 Chpt 13)

- 5.3.1 On-premise
  - Federated identity management can be hosted on-premise, and typically provides an organization with the most control
- 5.3.2 Cloud
  - Cloud-based apps used federated identify management (FIM) systems, which are a form of SSO
  - Cloud-based federation typically uses a third-party service to hsare federated identities (e.g. training sites use federated SSO systems)
    - commonly matching the user's internal login ID with a federated identify
- 5.3.3 Hybrid
  - A hybrid federation is a combination of a cloud-based solution and an on-premise solution

5.4 Implement and manage authorization mechanisms (OSG-9 Chpt 14)

- 5.4.1 Role Based Access Control (RBAC)
  - A key characteristic of the Role-Based Access Control (RBAC) model is the use of roles or groups
  - Instead of assigning permissions directly to users, user accounts are placed in roles and administrators assign privileges to the roles (typically defined by job function)
    - if the user account is in a role, the user has all privileges assigned to the role
  - MS Windows OS uses this model with groups
  - RBAC models can group users into roles based on the org's hierarchy, and it is a non-descretionary access control model; central authority access decisions can use the RBAC model
- 5.4.2 Rule Based access control
  - A key characteristic of the Rule-Based access control model is that it applies global rules to all subjects
    - e.g. firewalls uses rules that allow or block traffic to all users equally

    - Rules within the rule-based access control model are sometimes referred to as restrictions or filters
- 5.4.3 Mandatory Access Control (MAC)
    - **Mandatory Access Control (MAC)**: access control that requires the system itself to manage access controls in accordance with the org's security policies
    - A key characteristic of the MAC model is the use of labels applied to both subjects and objects
        - e.g. a label of top secret grants access to top-secret documents
    - When documented in a table, the MAC model sometimes resembles a lattice (i.e. climbing rosebush framework), so it is referred to as a lattice-based model
- 5.4.4 Discretionary Access Control (DAC)
    - **Discretionary Access Control (DAC)**: access control model in which the system owner decides who gets access
    - A key characteristic of the DAC model is that every object has an owner, and the owner can grant or deny access to any other subjects
        - e.g. you create a file and are the owner, and can grant permissions to that file
    - New Technology File System (NTFS) used in Windows, uses the DAC model
- 5.4.5 Attribute Based Access Control (ABAC)
    - **Attribute-Based Access Control (ABAC)**: an access control paradigm where access rights are granted to users with policies that combine attributes together
    - A key characteristic of the ABAC model is its use of rules that can include multiple attributes
        - this allows it to be much more flexible than a rule-based access control model that applies the rules to all subjects equally
        - many software-defined networks (SDNs) use the ABAC model
    - ABAC allows administrators to create rules within a policy using plain language statements such as "Allow Managers to access the WAN using a mobile device"
- 5.4.6 Risk based access control
    - Risk-based access control model grants access after evaluating risk; evaluating the environment and the situation and making risk-based decisions using policies embeded within software
        - Using machine learning, making predictive conclusions about current activity based on past activity

5.5 Manage the identity and access provisioning lifecycle (OSG-9 Chpts 13,14)

- 5.5.1 Account accesss review
    - Administrators need to periodically review user, system and service accounts to ensure they meet security policies and that they don’t have excessive privileges
    - Be careful in using the local system account as an application service account; although it allows the app to run without creating a special service account, it usually grants the app more access than it needs
    - You can use scripts to run periodically and check for unused accounts, and check priveleged group membership, removing unauthorized accounts
    - Guard against two access control issues:
        - excessive privilege: occurs when users have more privileges than assigned work tasks dictate; these privileges should be revoked
        - creeping privileges (AKA privilege creep): user accounts accumulating additional privileges over time as job roles and assigned tasks change

- 5.5.2 Provisioning and deprovisioning
    - Identity and access provisioning lifecycle refers to the creation, management, and deletion of accounts
        - this lifecycle is important because without properly defined and maintained user accounts, a system is unable to establish accurate identity, perform authentication, provide authorization, and track accountability
    - Provisioning/Onboarding
        - proper user account creation, or provisioning, ensures that personnel follow specific procedures when creating accounts
            - new-user account creation is AKA enrollment or registration
        - **automated provisioning**: information is provided to an app, that then creates the accounts via pre-defined rules (assigning to appropriate groups based on roles)
            - automated provisioning systems create accounts consistently
        - provisioning also includes issuing hardware, tokens, smartcards etc to employees
        - it’s important to keep accurate records when issuing hardware to employees
        - after provisioning, an org can follow up with onboarding processes, including:
            - the employee reads and signs the acceptable use policy (AUP)
            - explaining security best practices (like infected emails)
            - reviewing the mobile device policy
            - ensuring the employee’s computer is operational, and they can log in
            - configure a password manager
            - explaining how to access help desk
            - show to access, share and save resources
    - Deprovisioning/Offboarding
        - deprovisioning/offboarding occurs when an employee leaves the organization or is transferred to a different department
        - **account revocation**: deleting an account is the easiest way to deprovision
            - an employee's account is usually first disabled
            - supervisors can then review the user’s data and determine if anything is needed
            - note: if terminated employee retains access to a user account after the exit interview, the risk for sabatage is very high
        - deprovisioning includes collecting any hardware issued to an employee such as laptops, mobile devices and auth tokens
- 5.5.3 Role definition
    - Employee responsibilities can change in the form of transfers to a different role, or into a newly created role
        - for new roles, it’s important to define the role and the privileges needed by the employees in that role
    - Roles and associated groups need to be defined in terms of privileges
- 5.5.4 Privilege escalation (e.g. managed service accounts, use of usdo, minimizing its use)
    - Privilege escalation refers to any situation that gives users more privileges than they should have
    - Attackers use privilege escalation techniques to gain elevated privileges
    - **Horizontal privilege escalation**: gives an attacker similar privileges as the first compromised user, but from other accounts
    - **Vertical privilege escalation**: provides an attacker with significantly greater privileges

    - e.g. after compromising a regular user’s account an attacker can use vertical privilege escalation techniques to gain administrator privileges on the user’s computer
    - the attacker can then use horizontal privilege escalation techniques to access other computers in the network
    - this horizontal privilege escalation throughout the network is AKA **lateral movement**

5.6 Implement authentication systems (OSG-9 Chpt 14)

- 5.6.1 OpenID Connect (OIDC) / Open Authorization (Oauth)
  - OAuth 2.0 authorization framework enables third-party apps to obtain limited access to an HTTP service, either on behalf of a resource owner (by orchestrating an approval interaction), or by allowing third-party applications to obtain access on its own behalf; OAuth provides the ability to access resources from another service
  - OAuth is an open framework used for authentication and authorization protocols
  - OAuth is the most widely used open standard for authorization and delgation of rights for cloud services
  - The most common protocol built on OAuth is OpenID Connect (OIDC); OpenID is used for authentication
  - OAuth 2.0 is often used for delegated access to applications, e.g. a mobile game that automatically finds all of your new friends from a social media app is likely using OAuth 2.0;
  - Conversely, if you sign into a new mobile game using a social media account (instead of creating a user account just for the game), that process might use OIDC
  - **OpenID Connect (OIDC)**: an authentication layer using the OAuth 2.0 authorization framework, maintained by the OpenID Foundation, providing both authentication and authorization
    - OIDC is a RESTful, JSON (JavaScript Object Notation)-based auth protocol that, when pared with OAuth can provide identity verification and basic profile info; uses JSON Web Tokens (JWT) -- AKA ID token
  - OAuth and OIDC are used with many web-based applications to share information without sharing credentials
    - OAuth provides authorization
    - OIDC uses the OAuth framework for authorization and builds on the OpenID technologies for authentication

- 5.6.2 Security Assertion Markup Language (SAML)
  - Security Assertion Markup Language (SAML): an open XML-based standard commonly used to exchange authentication and authorization (AA) information between federated orgs
  - Frequently used to integrate cloud services and provides the ability to make authentication and authorization assertions
  - SAML provides SSO capabilities for browser access
  - SAML is a popular SSO standard on the internet - used to exchange authentication and authorization (AA) information
  - Organization for the Advancement of Structure Information Standards (OASIS) maintains it
  - SAML 2 spec utilizes three entities:
    - Principal or User Agent
    - Service Provider (SP): providing the service a user is interested in using

        - Identity Provider (IdP): a third-party that holds the user authentication and authorization info
    - IdP can send three types of XML messages known as assertions:
        - Authentication Assertion: provides proof that the user agent provided the proper credentials, identifies the identification method, and identifies the time the user agent logged on
        - Authorization Assertion: indicates whether the user agent is authorized to access the requested service; if denied, includes why
        - Attribute Assertion: attributes can be any information about the user agent

- 5.6.3 Kerberos

    - Kerberos is a network authentication protocol widely used in corporate and private networks and found in many LDAP and directory services solutions such as Microsoft Active Directory
    - It provides single sign-on and uses cryptography to strengthen the authentication process and protect logon credentials
    - Ticket authentication is a mechanism that employs a third-party entity to prove identification and provide authentication - Kerberos is a well-known ticket system
    - After users authenticate and prove their identity, Kerberos uses their proven identity to issue tickets, and user accounts present these tickets when accessing resources
    - Kerberos version 5 relies on symmetric-key cryptography (AKA secret-key cryptography) using the Advanced Encryption Standard (AES) symmetric encryption protocol
    - Kerberos provides confidentiality and integrity for authentication traffic using end-to-end security and helps protect against eavesdropping and replay attacks
    - Kerberos uses UDP port 88 by default
    - Kerberos elements:
        - **Key Distribution Center (KDC)**: the trusted third party that provides authentication services
        - **Kerberos Authentication Server**: hosts the functions of the KDC:
            - **ticket-granting service (TGS)**: provides proof that a subject has authenticated through a KDC and is authorized to request tickets to access other objects
                - the ticket for the full ticket-granting service is called a ticket-granting ticket (TGT); when the client asks the KDC for a ticket to a server, it presents credentials in the form of an authenticator message and a ticket (a TGT) and the ticket-granting service opens the TGT with its master key, extracts the logon session key for this client, and uses the logon session key to encrypt the client's copy of a session key for the server
                - a TGT is encrypted and includes a symmetric key, an expiration time, and user’s IP address
                - subjects present the TGT when requesting tickets to access objects
            - **authentication service (AS)**: verifies or rejects the authenticity and timeliness of tickets; often referred to as the KDC
        - **Ticket (AKA service ticket (ST))**: an encrypted message that provides proof that a subject is authorized to access an object
        - **Kerberos Principal**: typically a user but can be any entity that can request a ticket
        - **Kerberos realm**: a logical area (such as a domain or network) ruled by Kerberos
    - Kerberos login process:

1. user provides authentication credentials (types a username/password into the client)
2. client/TGS key generated
   - client encrypts the username with AES for transmission to the KDC
   - the KDC verifies the username against a db of known credentials
   - the KDC generates a symmetric key that will be used by the client and the Kerberos server
   - it encrypts this with a hash of the user's password
3. TGT generated - the KDC generates an encrypted timestamped TGT
4. client/server ticket generated
   - the KDC then transmits the encrypted symmetric key and the encrypted timestamped TGT to the client
   - the client installs the TGT for use until it expires
   - the client also decrypts the symmetric key using a hash of the user's password
   - NOTE: the client's password is never transmitted over the network, but it is verified
     - the server encrypts a symmetric key using a hash of the user's password, and it can only be decrypted with a hash of the user's password
5. user accesses requested service

- When a client wants to access an object (like a hosted resource), it must request a ticket through the Kerberos server, in the following steps:
  - the client sends its TGT back to the KDC with a request for access to the resource
  - the KDC verifies that the TGT is valid, and checks its access control matrix to verify user privileges for the requested resource
  - the KDC generates a service ticket and sends it to the client
  - the client sends the ticket to the server or service hosting the resource
  - the server or service hosting the resource verifies the validity of the ticket with the KDC
  - once identity and authorization are verified, Kerberos activity is complete
    - the server or service host then opens a session with the client and begins communication or data transmission

- 5.6.4 Remote Authentication Dial-in User Service (RADIUS) / Terminal Access Controller Access Control System Plus (TACACS+)

  - Remote Authentication Dial-in User Service (RADIUS): centralizes authentication for remote access connections, such as VPNs or dial-up access
    - a user can connect to any network access server, which then passes on the user's credentials to the RADIUS server to verify authentication and authorization and to track accounting
    - in this context, the network access server is the RADIUS client, and a RADIUS server acts as an authentication server
    - the RADIUS server also provides AAA services for multiple remote access servers
    - RADIUS uses the User Datagram Protocol (UDP) by default and encrypts only the password's exchange
    - RADIUS using Transport Layer Security (TLS) over TCP (port 2083) is defined by RFC 6614
    - RADIUS uses UDP port 1812 for RADIUS messages and UDP port 1813 for RADIUS Accounting messages

        - RADIUS encrypts only the password’s exchange by default
        - it is possible to use RADIUS/TLS to encrypt the entire session
    - Cisco developed Terminal Access Control Access Control System Plus (TACACS+) and released it as an open standard
        - provides improvements over the earlier version and over RADIUS, it separates authentication, authorization, and accounting into separate processes, which can be hosted on three different servers
        - additionally, TACACS+ encrypts all of the authentication information, not just the password, as RADIUS does
        - TACACS+ uses TCP port 49, providing a higher level of reliability for the packet transmissions Domain 6 **Security Assessment and Testing**

- Security assessment and testing programs are an important mechanism for validating the on-going effectiveness of security controls

    - they include a variety of tools, such as vulnerability assessments, penetration tests, software testing, audits, and other control validation

- Every org should have a security assessment and testing program defined and operational

- **Security assessments**: comprehensive reviews of the security of a system, application, or other tested environment

    - during a security assessment, a trained information security professional performs a risk assessment that identifies vulnerabilities in the tested environment that may allow a compromise and makes recommendations for remediation, as needed
    - a security assessment includes the use of security testing tools, but go beyond scanning and manual penetration tests
    - the main work product of a security assessment is normally an assessment report addressed to management that contains the results of the assessment in nontechnical language and concludes with specific recommendations for improving the security of the tested environment

- An organization’s audit strategy will depend on its size, industry, financial status and other factors

    - a small non-profit, a small private company and a small public company will have different requirements and goals for their audit strategies
    - the audit strategy should be assessed and tested regularly to ensure that the organization is not doing a disservice to itself with the current strategy
    - there are three types of audit strategies: internal, external, and third-party

- **Artifact**: piece of evidence such as text, or a reference to a resource which is submitted in response to a question

- **Assessment**: testing or evaluation of controls to understand which are implemented correctly, operating as intended and producing the desired outcome in meeting the security or privacy requirements of a system or org

- **Audit**: process of reviewing a system for compliance against a standard or baseline (e.g. audit of security controls, baselines, financial records) can be formal and independent, or informal/internal

- **Chaos Engineering**: discipline of experiments on a software system in production to build confidence in the system's capabilities to withstand turbulent/unexpected conditions
- **Code testing suite**: usually used to validate function, statement, branch and condition coverage
- **Compliance Calendar**: tracks an org's audits, assessments, required filings, due dates and related
- **Compliance Tests**: an evaluation that determines if an org's controls are being applied according to management policies and procedures
- **Penetration Testing/Ethical Penentration Testing**: security testing and assessment where testers actively attempt to circumvent/defaut a system's security features; typically constrained by contracts to stay within specified Rules of Engagement (RoE)
- **Examination**: process of reviewing/inspecting/observing/studying/analyzing specs/mechanisms/activities to understand, clarify, or obtain evidence
- **Findings**: results created by the application of an assessment procedure
- **Functional order of controls**: deter, deny, detect, delay, deterimine, and decide
- **IAM system**: identity and access management system combines lifecycle management and monitoring tools to ensure that identity and authorization are properly handled throughout an org
- **ITSM**: IT Service Management tools include change management and associated approval tracking
- **Judgement Sampling**: AKA purposive or authoritative sampling, a non-probability sampling technique where members are chosen only on the basis of the researcher's knowledge and judgement
- **Misue Case Testing**: testing strategy from a hostile actor's point of view, attempting to lead to integrity failures, malfunctions, or other security or safety compromises
- **Mutation testing**: mutation testing modifies a program in small ways and then tests that mutant to determine if it behaves as it should or if it fails; technique is used to design and test software through mutation
- **Plan of Action and Milestones (POA&M)**: a document indentifying tasks to be accomplished, including details, resources, milestones, and completion target dates
- **RUM**: real user monitoring is a passive monitoring technique that records user interation with an app or system to ensure performance and proper app behavior; often used as a predeploymment process using the actual user interface
- **RoE**: Rules of Engagement, set of rules/constraints/boundaries that establish limits of participant activity; in ethical pen testing, an RoE defines the scope of testing, and to establish liabilty limits for both testers and the sponsoring org or system owners
- **SCF**: Script Check Engine is designed to make scripts interoperable with security policy definitions
- **Statistical Sampling**: process of selecting subsets of examples from a population with the objective of estimating properties of the total population

- **Substantive Test**: testing technique used by an auditor to obtain the audit evidence in order to support the auditor's opinion
- **Testing**: process of exersizing one or more assessment objects (activities or mechanisms) under specified conditions to compare actual to expected behaior
- **Trust Services Criteria (TSC)**: used by an auditor when evaluating the suitability of the design and operating effectiveness of controls relevant to the security, availabiliity, or processing integrity of information and systems or the confidentiality or privacy of the info processed by the entity

6.1 Design and validate assessment, test, and audit strategies (OSG-9 Chpt 15)

- 6.1.1 Internal
    - An organization’s security staff can perform security tests and assessments, and the results are meant for internal use only, designed to evaluate controls with an eye toward finding potential improvements
    - An internal audit strategy should be aligned to the organization’s business and day-to-day operations
        - e.g. a publicly traded company will have a more rigorous internal auditing strategy than a privately held company
    - Designing the audit strategy should include laying out applicable regulatory requirements and compliance goals
    - Internal audits are performed by an organization’s internal audit staff and are typically intended for internal audiences
- 6.1.2 External
    - An external audit strategy should complement the internal strategy, providing regular checks to ensure that procedures are being followed and the organization is meeting its compliance goals
    - External audits are performed by an outside auditing firm
        - these audits have a high degree of external validity because the auditors performing the assessment theoretically have no conflict of interest with the org itself
        - audits by these firms are generally considered acceptable by most investors and governing bodies
- 6.1.3 Third-party
    - Third-party audits are conducted by, or on behalf of, another org
    - In the case of a third-party audit, the org initiating the audit generally selects the auditors and designs the scope of the audit
    - The statement on **Standards for Attestation Engagements document 18 (SSAE 18)**, titled Reporting on Controls, provides a common standard to be used by auditors performing assessments of service orgs with the intent of allowing the org to conduct external assessments, instead of multiple third-party assessments, and then sharing the resulting report with customers and potential customers
        - outside of the US, similar engagements are conducted under the International Standard for Attestation Engagements (ISAE) 3402, Assurance Reports on Controls at a Service Organization

  - SSAE 18 and ISAE 3402 engagements are commonly referred to as a service organization controls (SOC) audits
  - Three forms of SOC audits:
    - **SOC 1 Engagements**: assess the organization's controls that might impact the accuracy of financial reporting
    - **SOC 2 Engagements**: assess the organization's controls that affect the security (confidentiality, integrity, and availability) and privacy of information stored in a system
      - SOC 2 audit results are confidential and are usually only shared outside an org under an NDA
    - **SOC 3 Engagements**: assess the organization's controls that affect the security (confidentiality, integrity, and availability) and privacy information stored in a system
      - however, SOC3 audit results are intended for public disclosure
  - Two types of SOC reports:
    - **Type I Reports**: provide the auditor's opinion on the description provided by management and the suitability of the design of the controls
      - type I reports also cover only a specific point in time, rather than an extended period
      - think of Type I report as more of a documentation review
    - **Type II Reports**: go further and also provide the auditor's opinion on the operating effectiveness of the controls
      - the auditor actually confirms the controls are functioning properly
      - Type II reports also cover an extended period of time, at least 6 months
      - think of Type II report as similar to a traditional audit; the auditor is checking the paperwork, and verifying the controls are functioning properly
    - Type II reports are considered much more reliable than Type I reports (Type I reports simply take the service orgs word that the controls are implemented as described)

6.2 Conduct security control testing (OSG-9 Chpt 15)

- Security control testing can include testing of the physical facility, logical systems and applications; common testing methods:

- 6.2.1 Vulnerability assessment

  - **Vulnerabilities**: weaknesses in systems and security controls that might be exploited by a threat
  - Vulnerability assessments: examining systems for these weaknesses
  - The goal of a vulnerability assessment is to identify elements in an environment that are not adequately protected -- and not necessarily from a technical perspective; you can also assess the vulnerability of physical security or the external reliance on power, for instance
    - can include personnel testing, physical testing, system and network testing, and other facilities tests
  - Vulnerability assessments are some of the most important testing tools in the information security professional's toolkit
  - **Security Content Automation Protocol (SCAP)**: provides a common framework for discussion and facilitation of automation of interactions between different security systems (sponsored by NIST)
    - SCAP components related to vulnerability assessments:

        - **Common Vulnerabilities and Exposures (CVE)**: provides a naming system for describing security vulnerabilities
        - **Common Vulnerability Scoring Systems (CVSS)**: provides a standardized scoring system for describing the severity of security vulnerabilities; it includes metrics and calc tools for exploitability, impact, how mature exploit code is, and how vulnerabilities can be remediated, and a means to score vulns against users' unqiue requirements
        - **Common Configuration Enumeration (CCE)**: provides a naming system for system config issues
        - **Common Platform Enumeration (CPE)**: provides a naming system for operating systems, applications, and devices
        - **Extensible Configuration Checklist Description Format (XCCDF)**: provides a language for specifying security checklists
        - **Open Vulnerability and Assessment Language (OVAL)**: provides a language for describing security testing procedures; used to describe the security condition of a system
    - Vulnerability scans automatically probe systems, applications, and networks looking for weaknesses that could be exploited by an attacker
    - Four main categories of vulnerability scans:
        - network discovery scans
        - network vulnerability scans
        - web application vulnerability scans
        - database vulnerability scans
    - **Authenticated scans**: (AKA credentialed security scan) involves conducting vulnerability assessments and security checks on a network, system, or application using valid credentials; this approach enables the scanner to simulate the actions of an authenticated user, allowing it to access deeper layers of the target system, gather more information, and provide a more accurate assessment of vulnerabilities; often uses a read-only account to access configuration files

- 6.2.2 Penetration testing

    - Penetration tests goes beyond vulnerability testing techniques because it actually attempts to exploit systems
    - NIST defines the penetration testing process as consisting of four phases:
    - **planning**: includes agreement on the scope of the test and the rules of engagement
        - ensures that both the testing team and management are in agreement about the nature of the test and that it is explicitly authorized
    - **information gathering and discovery**: uses manual and automated tools to collect information about the target environment
        - basic reconnaissance (website mapping)
        - network discovery
        - testers probe for system weaknesses using network, web and db vuln scans
    - **attack**: seeks to use manual and automated exploit tools to attempt to defeat system security
        - step where pen testing goes beyond vuln scanning as vuln scans don’t attempt to actually exploit detected vulns

  - **reporting**: summarizes the results of the pen testing and makes recommendations for improvements to system security
  - tests are normally categorized into three groups:
    - **white-box penetration test**:
      - provides the attackers with **detailed information** about the systems they target
      - this bypasses many of the reconnaissance steps that normally precede attacks, shortening the time of the attack and increasing the likelihood that it will find security flaws
      - these tests are sometimes called "**known environment**" tests
      - in white-box testing, the tester has access to the source code and performss testing from a developer's perspective
    - **gray-box penetration test**:
      - AKA **partial knowledge tests**, these are sometimes chosen to balance the advantages and disadvantages of white- and black-box penetration tests
      - this is particularly common when black-box results are desired but costs or time constraints mean that some knowledge is needed to complete the testing
      - these tests are sometimes called "**partially known environment**" tests
      - in gray-box testing, the tester evaluates software from a user perspective but has access to the source code
    - **black-box penetration test**:
      - does not provide attackers with any information prior to the attack
      - this simulates an external attacker trying to gain access to information about the business and technical environment before engaging in an attack
      - these tests are sometimes called "**unknown environment**" tests

- 6.2.3 Log reviews

  - **Security Information and Event Management (SIEM)**: packages that collect information using the syslog functionality present in many devices, operating systems, and applications
  - Admins may choose to deploy logging policies through Windows Group Policy Objects (GPOs)
  - Logging systems should also make use of the Network Time Protocol (NTP) to ensure that clocks are synchronized on systems sending log entries to the SIEM as well as the SIEM itself, ensuring info from multiple sources have a consistent timeline
  - Information security managers should also periodically conduct log reviews, particularly for sensitive functions, to ensure that privileged users are not abusing their privileges
  - Network flow (NetFlow) logs are particularly useful when investigating security incidents

- 6.2.4 Synthetic transactions

  - **Synthetic transactions**: scripted transactions with known expected results
  - **Synthetic monitoring**: uses emulated or recorded transactions to monitor for performance changes in response time, functionality, or other performance monitors
  - Dynamic testing may include the use of synthetic transactions to verify system performance; synthetic transactions are run against code and compare out to expected state

- 6.2.5 Code review and testing

  - Code review and testing is "one of the most critical components of a software testing program"
  - These procedures provide third-party reviews of the work performed by developers before moving code into a production environment, possibly discovering security, performance, or reliability flaws in apps before they go live and negatively impact business operations
  - In code review, AKA peer review, developers other than the one who wrote the code review it for defects
  - **Fagan inspections**: the most formal code review process follows six steps:
    1. planning
    2. overview
    3. preparation
    4. inspection
    5. rework
    6. follow-up
       - Entry criteria are the criteria or requirements which must be met to enter a specific process
       - Exit criteria are the criteria or requirements which must be met to complete a specific process
  - **Static application security testing (SAST)**: evaluates the security of software without running it by analyzing either the source code or the compiled application
  - **Dynamic application security testing (DAST)**: evaluates the security of software in a runtime environment and is often the only option for organizations deploying applications written by someone else

- 6.2.6 Misuse case testing

  - **Misuse case testing**: AKA abuse case testing - used by software testers to evaluate the vulnerability of their software to known risks;focuses on behaviors that are not what the org desires or that are counter to the proper function of a system/app
  - In misuse case testing, testers first enumerate the known misuse cases, then attempt to exploit those use cases with manual or automated attack techniques

- 6.2.7 Test coverage analysis

  - A test coverage analysis is used to estimate the degree of testing conducted against new software; to provide insight into how well testing covered the use cases that an app is being tested for
  - **Test coverage**: number of use cases tested / total number of use cases
    - requires enumerating possible use cases (which is a difficult task), and anyone using test coverage calcs to understand the process used to develop the input values
  - Five common criteria used for test coverage analysis:
    - **branch coverage**: has every IF statement been executed under all IF and ELSE conditions?
    - **condition coverage**: has every logical test in the code been executed under all sets of inputs?
    - **functional coverage**: has every function in the code been called and returned results?
    - **loop coverage**: has every loop in the code been executed under conditions that cause code execution multiple times, only once, and not at all?

        - **statement coverage**: has every line of code been executed during the test?
    - **Test coverage report**: measures how many of the test cases have been completed; is used to provide test metrics when using test cases

- 6.2.8 Interface testing

    - Interface testing assesses the performance of modules against the interface specs to ensure that they will work together properly when all the development efforts are complete
    - Three types of interfaces should be tested:
        - application programming interfaces (APIs): offer a standardized way for code modules to interact and may be exposed to the outside world through web services
            - should test APIs to ensure they enforce all security requirements
        - user interfaces (UIs): examples include graphical user interfaces (GUIs) and command-line interfaces
            - UIs provide end users with the ability to interact with the software, and tests should include reviews of all UIs
        - physical interfaces: exist in some apps that manipulate machinery, logic controllers, or other objects
            - software testers should pay careful attention to physical interfaces because of the potential consequences if they fail
    - Also see OWASP API security top 10

- 6.2.9 Breach attack simulations

    - **Breach and attack simulation (BAS)**: platforms that seek to automate some aspects of penetration testing
    - The BAS platform is not actually waging attacks, but conducting automated testing of security controls to identify deficencies
    - A BAS system combines red team (attack) and blue team (defense) techniques together with automation to simulate advanced persistent threats (and other advanced threat actors) running against the environment
    - Designed to inject threat indicators onto systems and networks in an effort to trigger other security controls (e.g. place a suspicious file on a server)
        - detection and prevention controls should immediately detect and/or block this traffic as potentially malicious
    - See:
        - OWASP Web Security Testing Guide
        - OSSTMM (Open Source Security Testing Methodology Manual)
        - NIST 800-115
        - FedRAMP Penetration Test Guidance
        - PCI DSS Information Supplemental on Penetration Testing

- 6.2.10 Compliance checks

    - Orgs should create and maintain compliance plans documenting each of their regulatory obligations and map those to the specific security controls designed to satisfy each objective
    - Compliance checks are an important part of security testing and assessment programs for regulated firms: these checks verify that all of the controls listed in a compliance plan are

functioning properly and are effectively meeting regulatory requirements

6.3 Collect security process data (e.g. technical and administrative) (OSG-9 Chpts 15,18)

- 6.3.1 Account management
    - Preferred attacker techniques for obtaining privilege user access include:
        - compromising an existing privileged account: mitigated through use of strong authentication (strong passwords and multifactor), and by admins use of privileged accounts only for specific tasks
        - privelege escalation of a regular account or creation of a new account: these approaches can be mitigated by paying attention to the creation, modification, and use of user accounts
- 6.3.2 Management review and approval
    - Account management reviews ensure that users only retain authorized permissions and that unauthorized modifications do not occur
    - Full review of accounts: time-consuming to review all, and often done only for highly privileged accounts
    - Organizations that don't have time to conduct a full review process may use sampling, but only if sampling is truely random
    - Adding accounts: should be a well-defined process, and users should sign AUP
    - Adding, removing, and modifying accounts and permissions should be carefully controlled and documented
    - Accounts that are no longer needed should be suspended
    - ISO 9000 standards use a Plan-Do-Check-Act loop
        - plan: foundation of everything in the ISMS, determines goals and drives policies
        - do: security operations
        - check: security assessment and testing (this objective)
        - act: formally do the management review
- 6.3.3 Key performance and risk indicators
    - **Key Performance Indicator (KPIs)**: measures that provide significance of showing the performance an ISMS compared to stated goals
    - Choose the factors that can show the state of security
    - Define baselines for some (or better yet all) of the factors
    - Develop a plan for periodically capturing factor values (use automation!)
    - Analyze and interpret the data and report the results
    - Key metrics or KPIs that should be monitored by security managers may vary from org to org, but could include:
        - number of open vulns
        - time to resolve vulns
        - vulnerability/defect recurrence
        - number of compromised accounts
        - number of software flaws detected in pre-production scanning
        - repeat audit findings
        - user attempts to visit known malicious sites

  - Develop a dashboard of metrics and track them

- 6.3.4 Backup verification data

  - Managers should periodically inspect the results of backups to verify that the process functions effectively and meets the organization’s data protection needs
    - this might include reviewing logs, inspecting hash values, or requesting an actual restore of a system or file

- 6.3.5 Training and awareness

  - Training and awareness programs play a crucial role in preparing an organization’s workforce to support information security programs
  - They educate employees about current threats and advise them on best practices for protecting information and systems under their care from attacks
  - Program should begin with initial training designed to provide foundation knowledge to employees who are joining the org or moving to a new role; the initial training should be tailored to an individual’s role
  - Training and awareness should continue to take place throughout the year, reminding employees of their responsibilities and updating them on changes to the organization’s operating environment and threat landscape
  - Use phishing simulations to evaluate the effectiveness of their security awareness programs

- 6.3.6 Disaster Recover (DR) and Business Continuity (BC)

  - **Business Continuity (BC)**: the processes used by an organization to ensure, holistically, that its vital business processes remain unaffected or can be quickly restored following a serious incident
  - **Disaster Recovery (DR)**: is a subset of BC, that focuses on restoring information systems after a disaster
  - These processes need to be periodically accessed, and regular testing of disaster recovery and business continuity controls provide organizations with the assurance they are effectively protected against disruptions to business ops
  - Protection of life is of the utmost importance and should be dealt with first before attempting to save material things

6.4 Analyze test output and generate report (OSG-9 Chpt 15)

- Step 1: review and understand the data

  - The goal of the analysis process is to proceed logically from facts to actionable info
  - A list of vulns and policy exceptions is of little value to business leaders unless it's used in context, so once all results have been analyzed, you're ready to start writing the official report

- Step 2: determine the business impact of those facts

  - Ask "so what?"

- Step 3: determine what is actionable

  - The analysis process leads to valuable results only if they are actionable

- 6.4.1 Remediation
    - Rather than software defects, most vulnerabilities in average orgs come from misconfigured systems, inadequate policies, unsound business processes, or unaware staff
    - Vuln remediation should include all stakeholders, not just IT
- 6.4.2 Exception handling
    - **Exception handling**: the process of handling unexpected activity, since software should never depend on users behaving properly
        - "expect the unexpected", gracefully handle invalid input and improperly sequenced activity etc
    - Sometimes vulns can't be patched in a timely manner (e.g. medical devices needing re-accreditation) and the solution is to implement compensitory controls, document the exception and decision, and revisit
        - **compensitory controls**: measures taken to address any weaknesses of existing controls or to compensate for the inability to meet specific security requirements due to various different constraints
        - e.g. micro-segmentation of device, access restrictions, monitoring etc
    - Exception handling may be required due to system crash as the result of patching (requiring roll-back)
- 6.4.3 Ethical disclosure
    - While conducting security testing, cybersecurity pros may discover previously undiscovered vulns (perhaps implementing compensating controls to correct) that they may be unable to correct
    - **Ethical disclosure**: the idea that security pros who detect a vuln have a responsibility to report it to the vendor, providing them with enough time to patch or remediate
        - the disclosure should be made privately to the vendor providing reasonable amount of time to correct
        - if the vuln is not corrected, then public disclosure of the vuln is warrented, such that other professionals can make informed decisions about future use of the product(s)

6.5 Conduct or facilitate security audits (OSG-9 Chpt 15)

- 6.5.1 Internal
    - Having an internal team conduct security audits has several advantages:
        - understanding the internal environment reduces time
        - an internal team can delve into all parts of systems, because they have insider knowledge
        - internal auditors can be more agile in adapting to changing needs, rescheduling failed assessment components quickly
    - Disadvantages of using an internal team to conduct security audits:
        - the team may have limited exposure to new/other methodologies (e.g. the team may have depth but not breadth of experience and knowledge)
        - potential conflicts of interest (e.g. reluctance to throw other teams under the bus and accurately report their findings)

        - audit team members may start with an agenda (say to secure funding) and overstate faults, or have interpersonal motives

- 6.5.2 External
    - An external audit (sometimes called a second-party audit) is one conducted by (or on behalf of) a business partner
    - External audits are tied to contracts; by definition, an external audit should be scoped to include only the contractual obligations of an organization

- 6.5.3 Third-party
    - Third-party audits are often needed to demonstrate compliance with some government regulation or industry standard
    - Advantages of having a third-party audit an organization:
        - they likely have breadth of experience auditing many types of systems, across many types of organizations
        - they are not affected by internal dynamics or org politics
    - Disadvantage of using a third-party auditor:
        - cost: third-party auditors are going to be much more costly than internal teams; this means that the organization is likely to conduct audits as frequently
        - internal resources are still required to assist or accompany auditors, to answer questions and guide Domain 7 **Security Operations**

- **Allowed/Blocked listing**: allowed or blocked entities, register of entities that are being provided (or blocked) for a particular privilege, service, mobility, access or recognition including web, IP, geo, hardware address, files/programs; entities on the allowed list will be accepted, approved and/or recognized; deprecated AKA whitelist/blacklist; systems also alert IT security personnel an access attempt involves a resource not on a pre-approved list; can also incorporate anti-malware

- **Alternate site**: contingency or Continuity of Operations (COOP) site used to assume system or org operations, if the primary site is not available

- **Backup**: copies of files or programs to facilitate recovery

- **Baseline**: total inventory of all of a system's components (e.g. hardware, software, data, admin controls, documentation, user instruction); types of baselines include enumerated (which are inventory lists, generated by system cateloging, discovery or enumeration), build security (minimal set of securtiy controls for each CI, see below), modification/update/patch baselines (subsets of total system baseline), or configuration baseline (which should include a revision/version identifier associated with each CI)

- **Bastion host**: a special-purpose computer on a network specifically designed and configured to withstand attacks; it is typically placed in a demilitarized zone (DMZ) or exposed network segment, and its primary function is to act as a gateway between an internal network and external, potentially untrusted networks (like the internet); key characteristics of a Bastion host include:
    - Hardened security: minimizing the number of running services and apps which reduces potential attack surfaces

  - Publically accessible: exposed to the internet or untrusted network, acting as the first point of contact for external users
  - Logging and monitoring: include extensive logging and monitoring to detect suspicious activity
  - Limited network access: typically has limited access to the internal network

- **Clipping**: one of two main methods of choosing records from a large pool for ruther analysis, clipping uses threshold values to select those records exceeding a predefined threashold (also see sampling)

- **Configuration Item (CI)**: aggregation of information system components designated for configuration management and treated as a single entity in the config management process

- **Cyber forensics**: gathering, retaining, analyzing data for investigative purposes, while maintaining the integrity of that data

- **Disruption**: unplanned event that causes a system to be inoperable for a length of time

- **DPI (Deep Packet Inspection)**: a method used by firewalls and other network security devices to examine the data portion (or payload) of packets as they pass through the firewall; DPI goes beyond traditional packet filtering by not only inspecting the header information (such as source/destination IP addresses and port numbers) but also analyzing the content within the packet to identify and respond to security threats

- **Egress monitoring**: monitoring the flow of info out of an org's boundaries

- **Entitlement**:refers to the privelege granted to users when an account is first provisioned

- **Entity**: any form of a user including hardware device, software daemon, task, processing thread or human, which is attempting to use or access system resources; e.g. endpoint devices are entities that human (or non-human) users use to access a system; should be subject to access control and accounting

- **Event**: observable occurance in a network or system

- **Hackback**: actions taken by a victim of hacking to compromise the systems of the alleged attacker

- **Heuristics**: method of machine learning which identifies patterns of acceptable activity, so that deviations from the patterns will be identified

- **Incident**: an event which potentially or actually jeopardizes the CIA of an information system or the info the system processes, stores, transmits

- **Indicator**: technical artifact or observable occurrence suggesting that an attack is imminent, currently underway, or already occured

- **Indicators of Compromise (IoC)**: a signal that an intrusion, malware, or other predefined hostile or hazardous set of events has or is occurring

- **Information Security Continuous Monitoring (ICSM)**: maintaining ongoing awareness of information security, vulnerabilities and threats to support organizational risk management decisions; ongoing monitoring sufficient to ensure and assure effectiveness of security controls

- **Information Sharing and Analysis Center (ISAC)**: entity or collab created for the purposes of analyzing critical cyber and related info to better understand security problems and interdependencies to ensure CIA

- **Log**: record of actions/events that have taken place on a system

- **Motion detector types**: wave pattern motion detectors transmit ultrasonic or microwave signals into the montored area watching for changes in the returned signals bouncing off objects; infrared heat-based detectors watch for unusual heat patters; capacitance detectors work based on electromagnetic fields

- **MTBF**: mean time between failure is an estimation of time between the first and any subsequent failures

- **MTTF**: mean time to failure is the expected typical functional lifetime of the device given a specific operating enviornment

- **MTTR**: mean time to repair is the average length of time required to perform a repair on the device

- **Netflow**: data that contains info on the source, destination, and size of all network communications and is routinely saved as a matter of normal activity

- **Precursor**: signal from events suggesting a possible change of conditions, that may alter the current threat landscape

- **Regression testing**: testing of a system to ascertain whether recently approved modifications have changed its performance, or if other approved functions have introduced unauthorized behaviors

- **Request For Change (RFC)**: documentation of a proposed change in support of change management activities

- **Root Cause Analysis**: principle-based systems approach for the identification of underlying causes associated with a particular risk set or incidents

- **RTBH (Remote Triggered Black Hole)**: a network security technique used in conjunction with firewalls and routers to mitigate Distributed Denial of Service (DDoS) attacks or unwanted traffic by dropping malicious or unwanted traffic before it reaches the target network; RTBH works by creating a "black hole route", where packets destined for a specific IP address are discarded or "dropped" by the network equipment, effectively isolating malicious traffic

- **Sampling**: one of two main methods of choosing records from a large pool for ruther analysis, sampling uses statistical techniques to choose a sample that is representative of the entire pool (also see clipping)

- **SCCM**: System Center Configuration Manager is a Microsoft systems management software product that provides the capability to manage large groups of computers providing remote control, patch management, software distribution, operating system deployment, and hardware and software inventory

- **Security Incident**: Any attempt to undermine the security of an org or violation of a security policy is a security incident

- **SWG (Secure Web Gateway)**: a security solution that filters and monitors internet traffic for orgs, ensuring that users can securely access the web while blocking malicious sites, preventing data leaks, and enforcing web browsing policies; while it is not a traditional firewall, it complements firewall functionality by focusing specifically on web traffic security

- **TCP Wrappers**: a host-based network access control system used in Unix-like operating systems to filter incoming connections to network services; allows administrators to define which IP addresses or hostnames are allowed or denied access to certain network services, such as SSH, FTP, or SMTP, by controlling access based on incoming TCP connections; TCP Wrappers relies on two config files: /etc/hosts.allow, and /etc/hosts.deny

- **Trusted Computing Base (TCB)**: the collection of all hardware, software, and firmware components within an architecture that is specifically responsible for security and the isolation of objects that forms a trusted base
  - TCB is a term that is usually associated with security kernels and the reference monitor
  - a trusted base enforces the security policy
  - a security perimeter is the imaginary boundary that separates the TCB from the rest of the system; TCB comonents communicate with non-TCB components using trusted paths
  - the reference monitor is the logical part of the TCB that confirms whether a subject has the right to use a resource prior to granting access;
  - the security kernel is the collection of the TCB components that implement the functionality of the reference monitor

- **Tuple**: tuple usually refers to a collection of values that represent specific attributes of a network connection or packet; these values are used to uniquely identify and manage network flows, as part of a state table or rule set in a firewall; as an example, a 5-tuple is as a bundle of five values that identify a specific connection or network session, which might include the sourced IP address, source port numbers, destination IP address, destination port number, and the specific protocol in use (e.g. TCP UDP)

- **View-Based access controls**: access control that allows the database to be logically divided into components like records, fields, or groups allowing sensitive data to be hidden from non-authorized users; admins can set up views by user type, allowing only access to assigned views

7.1 Understand and comply with investigations (OSG-9 Chpt 19)

- **Investigation**: a formal inquiry and systematic process that involves gathering information to determine the cause of a security incident or violation

- Investigators must be able to conduct reliable investigations that will hold up in court; securing the scene is an essential and critical part of every investigation
  - securing the scene might include any/all of the following:
    - sealing off access to the area or crime scene
    - taking images of the scene
    - documenting evidence
    - ensuring evidence (e.g. computers, mobile devices, portable drives etc) is not contacted, tampered with, or destroyed
  - general principles:

        - identify and secure the scene
        - protect evidence -- proper collection of evidence preserves its integrity and the chain of custody
        - identification and examination of the evidence
        - further analysis of the most compelling evidence
        - final reporting of findings

- **Locard exchange principle**: whenever a crime is committed something is taken, and something is left behind

- The purpose of an investigation is to:

    - identify the root cause of the incident
    - prevent future occurrences
    - mitigate the impact of the incident on the organization

- Types of investigations:

    - **administrative**: an investigation that is focused on policy violations
    - **criminal**: conducted by law enforcement, this type of investigation tries to determine if there is cause to believe (beyond a reasonable doubt) that someone committed a crime
        - the goal is to gather evidence that can be used to convict in court
        - the job of a security professional is to preserve evidence, ensure law enforcement has been contacted, and assist as necessary
    - **civil**: non-criminal investigation for matters such as contract disputes
        - the goal of a civil investigation is to gather evidence that can be used to support a legal claim in court, and is typically triggered from an imminent or on-going lawsuit
        - the level of proof is much lower for a civil compared to a criminal investigation
    - **regulatory**: investigation initiated by a government regulator when there is reason to believe an organization is not in compliance
        - this type of investigation varies significantly in scope and could look like any of the other three types of investigation depending on the severity of the allegations
        - as with criminal investigations, it is key to preserve evidence, and assist the regulator’s investigators

- 7.1.1 Evidence collection and handling

    - Evidence collection is complex, should be done by professionals, and can be thrown out of court if incorrectly handled
    - It’s important to preserve original evidence
    - International Organization on Computer Evidence (IOCE) six principles for media, network and software analysis:
        - all general forensic and procedural principles must be applied to digital evidence collection
        - seizing digital evidence shouldn't change the evidence
        - accessing original digital evidence should only be done by trained professionals
        - all activity relating to seizure, access, storage, or transfer of digital evidence must be fully documented, preserved, and available for review

    - a person in possession of digital evidence is responsible for all actions taken with respect to that evidence
    - any agency that is responsible for seizing, accessing, storing, or transferring digital evidence is responsible for compliance with these principles
  - Scientific Working Group on Digital Evidence (SWGDE) developed principles for standardized recovery of computer-based evidence:
    - legal system consistency
    - use of a common language
    - durability
    - ability to cross international and state boundaries
    - instill confidence in evidence integrity
    - forensic evidence applicability at the individual, agency, and country levels
  - **ISO/IEC 27037: Guidelines for Identification, Collection, Acquisition, and Preservation of Digital Evidence**: the international standard on digital evidence handling, with four phases:
    - identification
    - collection
    - acquisition
    - preservation
  - Types of evidence:
    - **primary evidence**:
      - most reliable and used at trial
      - original documents (e.g. legal contracts), no copies or duplicates
    - **secondary evidence**:
      - less powerful and reliable than primary evidence (e.g. copies of originals, witness oral evidence etc)
      - if primary evidence is available secondary of the same content is not valid
    - **real evidence**: this type of evidence includes physical objects, such as computers, hard drives, and other storage devices, that can be brought into a court of law
    - **direct evidence**: this type of evidence is based on the observations of a witness or expert opinion and can be used to prove a fact at hand (with backup evidence support)
    - **circumstantial evidence**: this type of evidence is based on inference and can be used to support a conclusion, but not prove it
    - **corroborative evidence**: this type of evidence is used to support other evidence and can be used to strengthen a case
    - **hearsay evidence**: type of evidence that is based on statements made by someone outside of court and is generally not admissible; rule says that a witness cannot testify about what someone else told them; courts have applied it such that attorneys may not introduce system logs into evidence unless they are authenticated by a system admin
    - **best evidence rule**: states that the original evidence should be presented in court, rather than a copy or other secondary evidence
    - **parol evidence rule**: determines whether extra/additional evidence can be used to alter or explain a written contract, stating that a written contract takes precedence over any oral negotiations or stipulations that relate to it; the rule generally prohibits the introduction of parol (extra) evidence that contradicts or varies the contract's terms
  - It is important to note that evidence should be collected and handled in a forensically sound manner to ensure that it is admissible in court and to avoid any legal issues

  - The chain of custody: focuses on having control of the evidence -- who collected and handled what evidence, when, and where
    - think about establishing the chain of custody as:
      - tag,
      - bag, and
      - carry the evidence
  - Five rules of evidence: five evidence characteristics providing the best chance of surviving legal and other scrutiny:
    - **authentic**: evidence is not fabricated or planted, and can be proven through crime scene photos, or bit-for-bit copies of storage
    - **accurate**: evidence that has integrity (not been modified)
    - **complete**: evidence must be complete, and all parts available and shared, whether they support the case or not
    - **convincing**: evidence must be easy to understand, and convey integrity
    - **admissible**: evidence must be accepted as part of a case

- 7.1.2 Reporting and documentation

  - Each investigation should result in a final report that documents the goals of the investigation, the procedures followed, the evidence collected, and the final results
  - Preparing formal documentation prepares for potential legal action, and even internal investigations can become part of employment disputes
  - Identify in advance a single point of contact who will act as your liasion with law enforcement, providing a go-to person with a single perspective, potentially improving the working relationship
  - Participate in the FBI's InfraGard program

- 7.1.3 Investigative techniques

  - Whether in response to a crime or incident, an organizational policy breach, troubleshooting a system or network issue etc, digital forensic methodologies can assist in finding answers, solving problems, and in some cases, help in successfully prosecuting crimes
  - The forensic investigation process should include the following:
    - identification and securing of a crime scene
    - proper collection of evidence that preserves its integrity and the chain of custody
    - examination of all evidence
    - further analysis of the most compelling evidence
    - final reporting
  - Sources of information and evidence:
    - oral/written statements: given to police, investigators, or as testimony in court by people who witness a crime or who may have pertient information
    - written documents: checks, printed contracts, handrwitten letters/notes
    - computer systems: components, local/portable storage, memory etc
    - visual/audio: visual and audio evidence pertient to a security investigation could include photographs, video, taped recordings, and surveillance footage from security cameras
  - Several investigative techniques can be used when conducting analysis:
    - media analysis: examining the bits on a hard drive that are intact dispite not having an index

        - software analysis: focuses on an applications and malware, determining how it works and what it's trying to do, with a goal of attribution

- 7.1.4 Digital forensics tools, tactics, and procedures
    - Digital forensics: the scientific examination and analysis of data from storage media so that the information can be used as part of an investigation to identify the culprit or the root cause of an incident
    - **Live evidence**: data stored in a running system e.g. random access memory (RAM), cache, and buffers
    - Examining a live system can change the state of the evidence
        - small changes like interacting with the keyboard, mouse, loading/unloading programs, or of course powering off the system, can change or eliminate live evidence
    - Whenever a forensic investigation of a storage drive is conducted, two identical bit-for-bit copies of the original drive should be created first
    - **eDiscovery**: the process of identifying, collecting, and producing electronic evidence in legal proceedings

- 7.1.5 Artifacts (e.g. computer, network, mobile device)
    - Forensic artifacts: remnants of a system or network breach/attempted breach, which and may or may not be relevant to an investigation or response
    - Artifacts can be found in numerous places, including:
        - computer systems
        - web browsers
        - mobile devices
        - hard drives, flash drives

7.2 Conduct logging and monitoring activities (OSG-9 Chpts 17,21)

- 7.2.1 Intrusion detection and prevention
    - **Intrusion**: a security event, or a combination of multiple security events that constitutes an incident; occurs when an attacker attempts to bypass or can bypass or thwart security mechanisms and access an organization’s resources without the authority to do so
    - **Intrusion detection**: a specific form of monitoring events, usually in real time, to detect abnormal activity indicating a potential incident or intrusion
    - **Intrusion Detection System (IDS)**: (AKA burglar alarms) is a security service that monitors and analyzes network or system events for the purpose of finding/providing realtime/neartime warnings of unauthorized attempts to access system resources; automates the inspection of logs and real-time system events to detect intrusion attempts and system failures
        - an IDS is intended as part of a defense-in-depth security plan
    - **Intrusion Prevention Systems (IPS)**: a security service that uses available info to determine if an attack is underway, alerting and also blocking attacks from reaching intended target; includes detection capabilities, you’ll also see them referred to as intrusion detection and prevention systems (IDPSs)
    - NIST SP 800-94 Guide to Intrusion Detection and Prevention Systems provides comprehensive coverage of both IDS and IPS

- 7.2.2 Security Information and Event Management (SIEM)

  - Security Information and Event Management (SIEM): systems that ingest logs from multiple sources, compile and analyze log entries, and report relevant information
    - SIEM systems are complex and require expertise to install and tune
    - require a properly trained team that understands how to read and interpret info, and escalation procedures to follow when a legitimate alert is raised
    - SIEM systems represent technology, process, and people, and each is important to overall effectiveness
    - a SIEM includes significant intelligence functionality, allowing large amounts of logged events and analysis and correlation of the same to occur very quickly
  - SIEM capabilities include:
    - Aggregation
    - Normalization
    - Correlation
    - Secure storage
    - Analysis
    - Reporting

- 7.2.3 Continuous monitoring

  - After a SIEM is set up, configured, tuned, and running, it must be routinely updated and continuously monitored to function effectively
  - Effective continuous monitoring encompasses technology, processes, and people
  - Continuous monitoring steps are:
    - Define
    - Establish
    - Implement
    - Analyze/report
    - Respond
    - Review/update
  - **monitoring**: the process of reviewing information logs, looking for something specific
    - necessary to detect malicious actions by subjects as well as attempted intrusions and system failures
    - can help reconstruct events, provide evidence for prosecution, and create reports for analysis
    - continuous monitoring ensures that all events are recorded and can be investigated later if necessary
  - **log analysis**: a detailed and systematic form of monitoring where logged info is analyzed for trends and patterns as well as abnormal, unauthorized, illegal, and policy-violating activities
    - log analysis isn’t necessarily in response to an incident, it’s a periodic task

- 7.2.4 Egress monitoring

  - It’s important to monitor traffic exiting as well as entering a network, and **Egress monitoring** refers to monitoring outgoing traffic to detect unauthorized data transfer outside the org (AKA data exfiltration)

    - Common methods used to detect or prevent data exfiltration are data loss prevention (DLP) techniques and monitoring for steganography

- 7.2.5 Log management

    - **Log management**: refers to all the methods used to collect, process, and protect log entries (see SIEM definition above)
    - **rollover logging**: allows admins to set a maximum log size, when the log reaches that max, the system begins overwriting the oldest events in the log

- 7.2.6 Threat intelligence (e.g. threat feeds, threat hunting)

    - **Threat intelligence**: an umbrella term encompassing threat research and analysis and emerging threat trends; gathering data on potential threats, including various sources to get timely info on current threats; information that is aggregated, transformed, analyzed, interpreted, or enriched to provide the necessary context for the decision-making process
    - **Kill chain**: military model (used for both offense and defense):
        - find/identify a target through reconnaissance
        - get the target's location
        - track the target's movement
        - select a weapon to use on the target
        - engage the target with the selected weapon
        - evaluate the effectiveness of the attack
    - Orgs have adapted this model for cybersecurity: Lockheed Martin created the **Cyber Kill Chain** framework including seven ordered stages of an attack:
        - **reconnaissance**: attackers gather info on the target
        - **weaponize**: attackers identify an exploit that the target is vulnerable to, along with methods to send the exploit
        - **delivery**: attackers send the weapon to the target via phishing attacks, malicious email attachments, compromised websites, or other common social engineering methods
        - **exploitation**: the weapon exploits a vulnerability on the target system
        - **installation**: code that exploits the vulnerability then installs malware with a backdoor allowing attacker remote access
        - **command and control**: attackers maintain a command and control system, which controls the target and other compromised systems
        - actions on objectives: attackers execute their original goals such as theft of money, or data, destruction of assets, or installing additional malicious code (eg. ransomware)

- 7.2.7 User and Entity Behavior Analytics (UEBA)

    - **UEBA (aka UBA)**: focuses on the analysis of user and entity behavior as a way of detecting inappropriate or unauthorized activity (e.g. fraud, malware, insider attacks etc); analysis engines are typically included with SIEM solutions or may be added via subscription
    - **Behavior-based detection**: AKA statistical intrusion, anomaly, and heuristics-based detection, starts by creating a baseline of normal activities and events; once enough baseline data has been accumulated to determine normal activity, it can detect abnormal activity (that may indicate a malicious intrusion or event)

  - Behavior-based IDSs use the baseline, activity statistics, and heuristic evaluation techniques to compare current activity against previous activity to detect potentially malicious events
  - **Static code scanning techniques**: the scanner scans code in files, similar to white box testing
  - **Dynamic techniques**: the scanner runs executable files in a sandbox to observe their behavior.

7.3 Perform Configuration Management (CM) (e.g. provisioning, baselining, automation) (OSG-9 Chpt 16)

- **Configuration Management (CM)**: collection of activities focused on establishing and maintaining the integrity of IT products and info systems, via the control of processes for initializing, changing, and monitoring the configurations of those products/systems through their lifecycle; the process of identifying, controlling, and verifying the configuration of systems and components throughout their lifecycle
  - CM is an integral part of secure provisioning and relates to the proper configuration of a device at the time of deployment
  - CM helps ensure that systems are deployed in a secure, consistent state and that they stay in a secure, consistent state throughout their lifecycle
- **Provisioning**: taking a particular config baseline, making additional or modified copies, and placing those copies into the environment in which they belong; refers to installing and configuring the operating system and needed apps on new systems
  - new systems should be configured to reduce vulnerabilities introduced via default configurations; the key is to harden a system based on intended useage
- **Hardening a system**: process of applying security configurations, and locking down various hardware, communications systems, software (e.g. OS, web/app server, apps etc); normally performed based on industry guidelines and benchmarks like the Center for Internet Securit (CIS);
  - makes it more secure than the default configuration and includes the following:
    - disable all unused services
    - close all unused logical ports
    - remove all unused apps
    - change default passwords
- **Baseline**: in the context of configuration management, it is the starting point or starting config for a system
  - an easy way to think of a baseline is as a list of services; an OS baseline identifies all the settings to harden specific systems
  - many organizations use images to deploy baselines; baseline images improve the security of systems by ensuring that desired security settings are always configured correctly
  - baseline images improve the security of systems by ensuring that desired security settings are always configured correctly; they also reduce the amount of time required to deploy and maintain systems, reducing overall maintenance costs
- Automation: it's typical to create a baseline, and then use automated methods to add additional apps, features, or settings for specific groups of computers
  - note that admins can use create/modify group policy settings to create domain-level standardization or to make security-related Windows registry changes

7.4 Apply foundational security operations concepts (OSG-9 Chpt 16)

- Security operations encompasses the day-to-day tasks, practices, and processes involved in securing and maintaining the operational integrity of an organization's information systems and assets; it includes security monitoring, incident response, and security awareness and training

- The primary purpose of security operations practices is to safeguard assets such as information, systems, devices, facilities, and apps, and helping organizations to detect, prevent, and respond to security threats

- Implementing common security operations concepts, along with performing periodic security audits and reviews, demonstrates a level of due care and due diligence

- 7.4.1 Need-to-know/least privilege
  - **Need-to-know principle**: imposes the requirement to grant users access only to data or resources they need to perform assigned work tasks
  - **Least privilege principle**: states that subjects are granted only the privileges necessary to perform assigned work tasks and no more
    - privilege in this context includes both permissions to data and rights to perform systems tasks
    - limiting and controlling privileges based on this concept protects confidentiality and data integrity
    - principle relies on the assumption that all users have a well-defined job description that personnel understand
    - least privilege is typically focused on ensuring that user privileges are restricted, but it also applies to apps or processes (e.g. if an app or service is compromised, the attacker can assume the service account's privileges)

- 7.4.2 Separation of Duties (SoD) and responsibilities
  - **Separation of Duties (SoD)**: ensures that no single person has total control over a critical function or system
    - SoD policies help reduce fraud by requiring collusion between two or more people to perform unauthorized activity
    - example of how SoD can be enforced, is by dividing the security or admin capabilities and functions among multiple trusted individuals
  - **Two-person control**: (AKA two-man rule) requires the approval of two individuals for critical tasks
    - using two-person controls within an org ensures peer review and reduces the likelihood of collusion and fraud
    - ex: privilege access management (PAM) solutions that create special admin accounts for emergency use only; perhaps a password is split in half so that two people need to enter the password to log on
  - **Split knowledge**: combines the concepts of separation of duties and two-person control into a single solution; the info or privilege required to perform an operation is divided among two or more users, ensuring that no single person has sufficient privileges to compromise the security of the environment; M of N control is an example of split knowledge
  - Principles such as least privilege and separation of duties help prevent security policy violations, and monitoring helps to deter and detect any violations that occur despite the use of preventive controls

- 7.4.3 Privilege account management

  - **Privileged Account Management (PAM)**: solutions that restrict access to privileged accounts or detect when accounts use any elevated privileges (e.g. admin accounts)
    - Microsoft domains, this includes local admin accounts, Domain and Enterprise Admins groups
    - Linux includes root or sudo accounts
  - PAM solutions should monitor actions taken by privileged accounts, new user accounts, new routes to a router table, altering config of a firewall, accessing system log and audit files

- 7.4.4 Job rotation

  - **Job rotation**: (AKA rotation of duties) means that employees rotate through jobs or rotate job responsibilities with other employees
    - using job rotation as a security control provides peer review, reduces fraud, and enables cross-training
    - job rotation policy can act as both a deterrent and a detection mechanism

- 7.4.5 Service Level Agreements (SLA)

  - **Service Level Agreement (SLA)**: an agreement between an organization and an outside entity, such as a vendor, where the SLA stipulates performance expectations and often includes penalties if the vendor doesn’t meet these expectations
  - **Memoradum of Understanding (MOU)**: documents the intention of two entities to work together toward a common goal

7.5 Apply resource protection (OSG-9 Chpt 16)

- Media management should consider all types of media as well as short- and long-term needs and evaluate:
  - Confidentiality
  - Access speeds
  - Portability
  - Durability
  - Media format
  - Data format
- For the test, data storage media should include any of the following:
  - Paper
  - Microforms (microfilm and microfiche)
  - Magnetic (HD, disks, and tapes)
  - Flash memory (SSD and memory cards)
  - Optical (CD and DVD)
- Mean Time Between Failure (MTBF) is an important criterion when evaluating storage media, especially where valuable or sensitive information is concerned
- Media management includes the protection of the media itself, which typically involves policies and procedures, access control mechanisms, labeling and marking, storage, transport, sanitization, use, and end-of-life
- 7.5.1 Media management
  - **Media management**: refers to the steps taken to protect media (i.e. anything that can hold data) and the data stored on that media; includes mostt portable devices (e.g. smart phones,

memory/flash cards etc)
  - As above, OSG-9 also refers to tape media, as well as “hard-copy data”
- 7.5.2 Media protection techniques
  - If media includes sensitive info, it should be stored in a secure location with strict access controls to prevent loss due to unauthorized access
    - any location used to store media should have temperature and humidity controls to prevent losses due to corruption
  - Media management can also include technical controls to restrict device access from computer systems
  - When media is marked, handled, and stored properly, it helps prevent unauthorized disclosure (loss of confidentiality), unauthorized modification (loss of integrity), and unauthorized destruction (loss of availability)

7.6 Conduct incident management (OSG-9 Chpt 17)

- **Incident response**: the mitigation of violations of security policies and recommended practices; the process to detect and respond to incidents and to reduce the impact when incidents occur; it attempts to keep a business operating or restore operations as quickly as possible in the wake of an incident

- Incident management is usually conducted by an Incident Response Team (IRT), which comprises individuals with the required expertise and experience to manage security incidents; the IRT is accountable for implementing the incident response plan, which is a written record that defines the processes to be followed during each stage of the incident response cycle

- The main goals of incident response:

  - Provide an effective and efficient response to reduce impact to the organization
  - Maintain or restore business continuity
  - Defend against future attacks

- An important distinction needs to be made to know when an incident response process should be initiated: events take place continually, and the vast majority are insignificant; however, events that lead to some type of adversity can be deemed incidents, and those incidents should trigger an org's incident response process steps:

  - **Preparation**: includes developing the IR process, assigning IR team members, and everything related to what happens when an incident is identified; preparation is critical, and will anticipate the steps to follow
  - **Analysis:** Gathering and analyzing information about the incident to determine its scope, impact, and root cause (e.g., by interviewing witnesses, collecting and analyzing evidence, and reviewing system logs)
  - **Containment:** Limiting the impact of the incident and preventing further damage (e.g., by isolating affected systems, changing passwords, and implementing security controls)
  - **Eradication:** Removing the cause of the incident from the environment (e.g., by removing malware, patching vulnerabilities, and disabling compromised accounts)
  - **Recovery:** Restoring systems and data to their normal state (e.g., by restoring from backups, rebuilding systems, and re-enabling compromised accounts)

  - **Lessons Learned:** Documenting the incident and learning from it to improve future responses (e.g., by identifying areas where the incident response process can be improved and by sharing lessons learned with other organizations)

- The following steps (Detection, Response, Migtation, Reporting, Recovery, Remediation, and Lessons Learned) are on the exam

- 7.6.1 Detection

  - **Detection:** the identification of potential security incidents via monitoring and analyzing security logs, threat intelligence, or incident reports; as above, understanding the distinction between an event and an incident, the goal of detection is to identify an adverse event (an incident) and begin dealing with it
  - Common methods to detect incidents:
    - intrusion detection and prevention systems
    - antimalware
    - automated tools that scan audit logs looking for predefined events
    - end users sometimes detect irregular activity and contact support
  - Note: receiving an alert or complaint doesn’t always mean an incident has occurred

- 7.6.2 Response

  - After detecting and verifying an incident, the next step is activate an Incident Response (IR) or CSIRT team
  - An IR team is AKA computer incident response team (CIRT) or computer security incident response team (CSIRT)
  - Among the first steps taken by the IR Team will be an impact assessment to determine the scale of the incident, how long the impact might be experienced, who else might need to be involved etc.
  - The IR team typicall investigate the incident, assess the damage, collect evidence, report the incident, perform recovery procedures, and participate in the remediation and lessons learned stages, helping with root cause analysis
  - its important to protect all data as evidence during an investigation, and computers should not be turned off

- 7.6.3 Mitigation

  - **Migitation**: attempt to contain an incident; in addition to conducting an impact assessment, the IR Team will attempt to minimize or contain the damage or impact from the incident
  - The IR Team's job at this point is not to fix the problem; it's simply to try and prevent further damage
  - Note this may involve disconnecting a computer from the network; sometimes responders take steps to mitigate the incident, but without letting the attacker know that the attack has been detected

- 7.6.4 Reporting

  - Reporting occurs throughout the incident response process
  - Once an incident is mitigated, formal reporting occurs because numerous stakeholders often need to understand what has happened

  - Jurisdictions may have specific laws governing the protection of personally identifiable information (PII), and must report if it's been exposed
  - Additionally, some third-party standards, such as the Payment Card Industry Data Security Standard (PCI DSS), require orgs to report certain security incidents to law enforcement

- 7.6.5 Recovery

  - At this point, the goal is to start returning to normal
  - Recovery is the next step, returning a system to a fully functioning state
  - The most secure method of restoring a system after an incident is completely rebuilding the system from scratch, including restoring all data from the most recent backup
    - effective configuration and change management will provide the necessary documentation to ensure the rebuilt systems are configured properly
  - According to the OGS, you should check these areas as part of recovery:
    - access control lists (ACLs), including firewall or router rules
    - services and protocols, ensuring the unneeded services and protocols are disabled or removed
    - patches
    - user accounts, ensuring they have changed from default configs
    - known compromises have been reversed

- 7.6.6 Remediation

  - **Remdiation**: changes to a system's config to immediately limit or reduce the change of reoccurance of an incident;
  - Remediation stage: personnel look at the incident, identify what allowed it to occur, and then implement methods to prevent it from happening again
  - Remediation includes performing a root cause analysis (which examines the incident to determine what allowed it to happen), and if the root cause analysis identifies a vulnerability that can be mitigated, this stage will recommend a change

- 7.6.7 Lessons Learned

  - Lessons learned stage: an all-encompassing view of the situation related to an incident, where personnel, including the IR team and other key stakeholders, examine the incident and the response to see if there are any lessons to be learned
    - the output of this stage can be fed back to the detection stage of incident management
  - It's common for the IR team to create a report when they complete a lessons learned review
    - based on the findings, the team may recommend changes to procedures, the addition of security controls, or even changes to policies
    - management will decide what recommendations to implement and is responsible for the remaining risk for any recommendations they reject

- NOTE: Incident management DOES NOT include a counterattack against the attacker

- Incident Response Summary:

Step	Stage	Action/Goal
Preparation		

Step	Stage	Action/Goal
Detection	Triage	
Response	Triage	activate IR team
Mitigation	Investigate	containment
Reporting	Investigate	
Recovery	Recovery	return to normal
Remediation	Recovery	prevention
Lessons Learned	Recovery	improve process

7.7 Operate and maintain detective and preventative measures (OSG-9 Chpts 11,17)

- As noted in Domain 1, a preventive or preventative control is deployed to thwart or stop unwanted or unauthorized activity from occurring
    - Examples:
        - fences
        - locks
        - biometrics
        - separation of duties policies
        - job rotation policies
        - data classification
        - access control methods
        - encryption
        - smart cards
        - callback procedures
        - security policies
        - security awareness training
        - antivirus software
        - firewalls
        - intrusion prevention systems
- A detective control is deployed to discover or detect unwanted or unauthorized activity; detective controls operate after the fact
    - Examples:
        - security guards, guard dogs
        - motion detectors
        - recording and reviewing of events captured by security cameras
        - job rotation policies
        - mandatory vacation policies
        - audit trails
        - honeypots or honeynets
        - intrusion detection systems
        - violation reports

        - supervision and reviews of users
        - incident investigations

- Some preventative measures:

    - Keep systems and applications up to date
    - Remove or disable unneeded services and protocols
    - Use intrusion detection and prevention systems
    - Use up-to-date antimalware software
    - Use firewalls
    - Implement configuration and system management processes

- 7.7.1 Firewalls (e.g. next generation, web application, network)

    - Firewalls are preventive and technical controls

    - Types of firewalls:

        - **Static Packet Filtering**: inspects individual packets based on predefined rules (such as IP address, port number, and protocol) without considering the connection state or the content of the data; simple and fast, but lacks context awareness
        - **Application-Level**: functions at the application layer (OSI:Layer 7), acts as an intermediary or proxy, inspecting traffic between the user and the service; can perform deep packet inspection, meaning it can analyze the contents of data packets to identify malicious content or enforce rules for specific applications (e.g., web, email); example: a web application firewall (WAF) inspects traffic going to a web server and can block malicious traffic such as SQL injection attacks and cross-site scripting (XSS) attacks
        - **Circuit-Level Gateway Firewall**: works at the session layer (OSI:Layer 5), and monitors TCP handshakes (i.e., the connection establishment process) to ensure the validity of the session; once the session is validated, it allows the traffic to pass without further inspection of the content; circuit-level gateway firewalls have lower processing overhead, but lacks deep packet inspection
        - **Stateful Inspection Firewall**: operates at the network and transport layers (Layers 3 and 4) but maintains a record of active connections (i.e., it tracks the state of traffic streams across the network); checks whether a packet belongs to an active, legitimate connection before allowing it through; offers better security than static packet filtering; lacks the ability to inspect data at the application layer
        - **Next-Generation Firewall (NGFW)**: functions as a unified threat management (UTM) device and combines the features of traditional firewalls (like stateful inspection) with additional features such as deep packet inspection, intrusion prevention systems (IPS), and the ability to detect and block threats at the application layer; often incorporates advanced threat detection using techniques such as sandboxing and behavioral analysis; an NFGW inspects traffic at both the application and network layers, providing comprehensive security, including the ability to identify and block sophisticated threats, but is more expensive and resource-intensive
        - **Internal Segmentation Firewall (ISFW)**: used within a network to segment internal traffic and control access between different parts of an org; an ISFW monitors and filters traffic between network segments (such as between the finance department and HR),

preventing lateral movement of threats within the network; provides internal protection by monitoring east-west traffic, reduces the risk of an insider threat or lateral movement, can enforce micro-segementation, but can be complex to configure and management

Firewall Type	OSI Layers	Key Features	Strengths	Weaknesses
Static Packet Filtering	Layer 3 (Network)	Basic filtering on source/destination IPs and ports	Fast, low overhead	No context awareness, can't inspect data payload
Application-Level	Layer 7 (Application)	Inspects application-level data	Deep inspection, blocks specific applications	High processing overhead, slower performance
Circuit-Level	Layer 5 (Session)	Validates session establishment	Low overhead, monitors session validity	No payload inspection, can't detect deeper threats
Stateful Inspection	Layers 3-4 (Network, Transport)	Tracks connection states across sessions	Better security than static filtering	Doesn't inspect data at the application layer
NGFW	Layers 3-7	Combines stateful inspection with deep packet inspection, IPS, and app control	Comprehensive threat detection, application-aware	Expensive, high resource usage
ISFW	Internal Segmentation	Filters traffic between internal network segments	Prevents lateral movement, enforces micro-segmentation	Complex configuration, typically for internal use

- 7.7.2 Intrusion Detection Systems (IDS) and Intrusion Prevention Systems (IPS)
  - Intrusion Detection Systems (IDSs) and Intrusion Prevention Systems (IPSs) are two methods organizations typically implement to detect and prevent attacks
  - **Intrusion detection**: a specific form of monitoring events, usually in real time, to detect abnormal activity indicating a potential incident or intrusion
    - Intrusion Detection System (IDS) automates the inspection of logs and real-time system events to detect intrusion attempts and system failures
    - IDSs are an effective method of detecting many DoS and DDoS attacks

        - an IDS actively watches for suspicious activity by monitoring network traffic and inspecting logs
        - an IDS is intended as part of a defense-in-depth security plan
        - **knowledge-based detection**: AKA signature-based or pattern-matching detection, the most common method used by an IDS
        - **behavior-based detection**: AKA statistical intrusion, anomaly, and heuristics-based detection; behavior-based IDSs use baseline, activity stats, and heuristic eval techniques to compare current activity against previous activity to detect potentially malicious events
    - An IPS includes detection capabilities, you'll see them referred to as intrusion detection and prevention systems (IDPSs)
        - an IPS includes all the capabilities of an IDS but can also take additional steps to stop or prevent intrusions
    - IDS/IPS should be deployed at strategic network locations to monitor traffic, such as at the perimeters, or between network segments, and should be configured to alert for specific types of scans and traffic patterns
    - See NIST SP 800-94

- 7.7.3 Whitelisting/blacklisting

    - Method used to control which applications run and which applications can't is allow list and deny list (AKA whitelists and blacklists)
    - **Allow list**:: identifies a list of apps authorized to run on a system and blocks all other apps
    - **Deny list**: identifies a list of apps that are not authorized to run on a system
    - Allow and deny lists use for applications help to prevent malware infections
    - Important to note: a system would only use one list, either allow or deny
    - Apple iOS running on iPhones/iPads is an example of an extreme version of an allow list; users are only able to install apps from the App Store

- 7.7.4 Third-party provided security services

    - Some orgs outsource security services such as auditing and penetration testing to third party security services
    - Some outside compliance entities (e.g. PCI DSS) require orgs to ensure that service providers comply
    - OSG also mentions that some SaaS vendors provide security services via the cloud (e.g. next-gen firewalls, UTM devices, and email gateways for spam and malware filtering)

- 7.7.5 Sandboxing

    - **Sandboxing**: refers to a security technique where a separate, secure environment is created to run and analyze untested or untrusted programs or code without risking harm to the host device or network; this isolated environment, known as a **sandbox**, effectively contains the execution of the code, allowing it to run and behave as if it were in a normal computing environment, but without the ability to affect the host system or access critical resources and data
    - **Confinement**: restriction of a process to certain resources, or reading from and writing to certain memory locations; bounds are the limits of memory a process cannot exceed when

reading or writing;isolation is using bounds to create/enforce confinement
    - Sandboxing provides a security boundary for applications and prevents the app from interacting with other apps; can be used as part of development, integration, or acceptance testing, as part of malware screening, or as part of a honeynet

- 7.7.6 Honeypots/honeynets

    - **Honeypots**: individual computers created as a trap or a decoy for intruders or insider threats
    - **Honeynet**: two or more networked honeypots used together to simulate a network
    - They look and act like legit systems, but they do not host data of any real value for an attacker; admins often configure honeypots with vulnerabilities to tempt intruders into attacking them
    - In addition to keeping the attacker away from a production environment, the honeypot allows administrators to observe an attacker's activity without compromising the live environment

- 7.7.7 Anti-malware

    - **Malware**: program inserted into a system with the intent of compromising the CIA of the victim's data, applications, or OS; malicious software that negatively impacts a system

    - The most important protection against malicious code is the use of antimalware software with up-to-date signature files and heuristic capabilities

        - multi-pronged approach with antimalware software on each system in addition to filtering internet content helps protect systems from infections
        - following the principle of least privilege, ensuring users do not have admin permissions on systems won't be able to install apps that may be malicious

    - These are the characteristics of each malware type:

        - **virus**: software written with the intent/capability to copy and disperse itself without direct owner knowledge/cooperation; the defining characteristic is that it's a piece of malware that has to be triggered in some way by the user; program that modifies other programs to contain a possibly altered version of itself
        - **worm**: software written with the intent/capability to copy and disperse without owner knowledge/cooperation, but without needing to modify other programs to contain copies of itself; malware that can self-propagate and spread through a network or a series of systems on its own by exploiting a vulnerability in those systems
        - **companion**: helper software that is not malicious on its own; it could be something like a wrapper that accompanies the actual malware
        - **macro**: associated with Microsoft Office products, and is created using a straightforward programming language to automate tasks; macros can be programmed to be malicious and harmful
        - **multipartite**: means the malware spreads in different ways (e.g. Stuxnet)
        - **polymorphic**: malware that can change aspects of itself as it replicates to evade detection (e.g. file name, file size, code structure etc)
        - **trojan**: a Trojan horse is malware that looks harmless or desirable but contains malicious code; trojans are often found in easily downloadable software; a trojan inserts backdoors or trapdoors into other programs or systems

        - **bot**: an emerging class of mobile code; employing limited machine learning capabilities to assist with user requests for help or assistance, automation of or assistance with workflows, data input quality validation etc.
        - **botnet**: many infected systems that have been harnessed together and act in unison
        - **boot sector infectors**: pieces of malware that can install themselves in the boot sector of a drive
        - **hoaxes/pranks**: not actually software, they're usually part of social engineering—via email or other means—that intends harm (hoaxes) or a joke (pranks)
        - **logic bomb**: malware inserted into a program which will activate and perform functions suiting the attacker when some later date/conditions are met; code that will execute based on some triggering event
        - **stealth**: malware that uses various active techniques to avoid detection
        - **ransome attack**: any form of attack which threatens the destruction, denial or unauthorized public release/remarketing of private infomation assets; usually involves encrypting assets and withholding the decryption key until a ransom is paid
        - **ransomware**: type of malware that typically encrypts a system or a network of systems, effectively locking users out, and then demands a ransom payment (usually in the form of a digital currency) to gain access to the decryption key
        - **rootkit**: Similar to stealth malware, a rootkit attempts to mask its presence on a system; malware that embeds itself deeply in an OS; term is derived from the concept of rooting and a utility kit of hacking tools; rooting is gaining total or full control over a system; typically includes a collection of malware tools that an attacker can utilize according to specific goals
        - **zero-day**: is any type of malware that's never been seen in the wild before, and the vendor of the impacted product is unaware (or hasn't issued a patch), as are security companies that create anti-malware software intended to protect systems; previously unreported vuln which can be potentially exploited without risk of detection or prevention until system owner/developer detects and corrects vuln; gets name from the "zero time" being the time at which the exploit or vuln is first identified by the systems' owners or builders; AKA zero-hour exploit, zero-day attack

- 7.7.8 Machine learning and Artificial Intelligence (AI) based tools
    - **AI**: gives machines the ability to do things that a human can do better or allows a machine to perform tasks that we previously thought required human intelligence
    - **Machine Learning**: a subset of AI and refers to a system that can improve automatically through experience
        - a ML system starts with a set of rules or guidelines
        - an AI system starts with nothing and progressively learns the rules, creating its own algorithms as it learns the rules and applies ML techniques based on these rules
    - Behavior-based detection is one way ML and AI can apply to cybersecurity
        - an admin relates a baseline of normal activities and traffic on a network; the baseline in this case is similar to a set of rules given to a ML system
        - during normal operations, it detects anomalies and reports them; if the detection is a **false positive** (incorrectly classifying a benign activity, system state, or configuration as malicious or vulnerable), the ML system learns

  - An AI system starts without a baseline, monitors traffic and slowly creates its own baseline based on the traffic it observes
    - as it creates the baseline it also looks for anomalies
    - an AI system also relies on feedback from admins to learn if alarms are valid or false positives

7.8 Implement and support patch and vulnerability management (OSG-9 Chpt 16)

- **Vulnerability Management**: activities necessary to identify, assess, prioritize, and remediate information systems weaknesses
- Patch and vulnerability management processes work together to help protect an org against emerging threats; patch management ensures that appropriate patches are applied, and vuln management helps verify that systems are not vulnerable to known threats
- **Patch**: (AKA updates, quick or hot fixes) a blanket term for any type of code written to correct bug or vulnerability or to improve existing software performance; when installed, a patch directly modifies files or device settings without changing the version number or release details of the related software comonent
  - in the context of security, admins are primarily concerned with security patches, which are patches that affect a system's vulns
- **Patch Management**: systematic notification, identification, deployment, installation and verification of OS and app code revisions known as patches, hot fixes, and service packs
  - an effective patch management program ensures that systems are kept up to date with current patches
- **Patch Tuesday**: several big-tech orgs (e.g. Microsoft, Adobe, Oracle etc) regularly release patches on the second Tuesday of every month
- There are three methods for determining patch levels:
  - agent: update software (agent) installed on devices
  - agentless: remotely connect to each device
  - passive: monitor traffic to infer patch levels
- Deploying patches can be done manually or automatically
- Common steps within an effective program:
  - evaluate patches: determine if they apply to your systems
  - test patches: test patches on an isolated, non-production system to determine if the patch causes any unwanted side effects
  - approve the patches: after successful testing, patches are approved for deployment; it's common to use Change Management as part of the approval process
  - deploy the patches: after testing and approval, deploy the patches; many orgs use automated methods to deploy patches, via third-party or the software vendor
  - verify that patches are deployed: regularly test and audit systems to ensure they remain patched
- **Vulnerability Management**: regularly identifying vulns, evaluating them, and taking steps to mitigate risks associated with them
  - it isn't possible to eliminate risks, and it isn't possible to eliminate all vulnerabilities
  - a vuln managment program helps ensure that an org is regularly evaluating vulns and mitigating those that represent the greatest risk
  - one of the most common vulnerabilities within an org is an unpatched system, and so a vuln management program will often work in conjunction with a patch management program

7.9 Understand and participate in change management processes (OSG-9 Chpt 16)

- **Change management**: formal process an org uses to transition from the current state to a future state; typically includes mechanisms to request, evaluate, approve, implement, verify, and learn the change; ensures that the costs and benefits of changes are analyzed and changes are made in a controlled manner to reduce risks
  - Change management processes allow various IT experts to review proposed changes for unintended consequences before implementing
  - Change management controls provide a process to control, document, track, and audit all system changes
- The change management process includes multiple steps that build upon each other:
  - Change request: a change request can come from any part of an org and pertain to almost any topic; companies typically use some type of change management software
  - Assess impact: after a change request is made, however small the request might be, the impact of the potential change must be assessed
  - Approval/reject: based on the requested change and related impact assessment, common sense plays a big part in the approval process
  - Build and test: after approval, any change should be developed and tested, ideally in a test environment
  - Schedule/notification: prior to implementing any change, key stakeholders should be notified
  - Implement: after testing and notification of stakeholders, the change should be implemented; it's important to have a roll-back plan, allowing personnel to undo the change
  - Validation: once implemented, senior management and stakeholders should again be notified to validate the change
  - Document the change: documentation should take place at each step; it's critical to ensure all documentation is complete and to identify the version and baseline related to a given change
- When a change management process is enforced, it creates documentation for all changes to a system, providing a trail of info if personnel need to reverse the change, or make the same change on other systems
- Change management control is a mandatory element for some security assurance requirements (SARs) in the ISO Common Criteria

7.10 Implement recovery strategies (OSG-9 Chpt 18)

- **Recovery strategy**: a plan for restoring critical business components, systems, and operations following a disruption

- **Disaster recovery (DR)**: set of practices that enables an organization to minimize loss of, and restore, mission-critical technology infrastructure after a catastrophic incident

- **Business continuity (BC)**: set of practices that enables an organization to continue performing its critical functions through and after any disruptive event

- 7.10.1 Backup storage strategies

  - Backup strategies are driven by org goals and objectives and usually focus on backup and restore time as well as storage needs

- **Archive bit**: technical detail (metadata) that indicates the status of a backup relative to a given backup strategy
    - 0 = no changes to the file or no backup required
    - 1 = file has been modified or backup required
- Different backup strategies deal with the archive bit differently; Incremental and differential backup strategies don't treat the archive bit in the same manner
    - once a full backup is complete, the archive bit on every file is reset, turned off, or set to 0
- Three types of backups:
    - **Full backup**: store a complete copy of the data contained on the protected device or backup media; full backups duplicate every file on the system regardless of the setting of the archive bit
    - **Incremental backup**: changes since the last incremental backup
        - only files that have the archive bit turned on, enabled, or set to 1 are duplicated
        - once an incremental backup is complete, the archive bit on all duplicated files is reset, turned off, or set to 0
    - **Differential backup**: changes since the last full backup
        - only files that have the archive bit turned on, enabled, or set to 1 are duplicated
        - unlike full and incremental backups, the differential backup process does not change the archive bit
    - the most important difference between incremental and differential backups is the time needed to restore data in the event of an emergency
        - a combination of full and differential backups will require only two backups to be restored: the most recent full backup and the most recent differential backup
        - a combination of full backups with incremental backups will require restoration of the most recent full backups as well as all incremental backups performed since that full backup
        - differential backups don’t take as long to restore, but they take longer to create than incremental
    - Note: Grandfather/Father/Son, Tower of Hanoi, and Six Cartridge Weekly are all different approaches to rotating backup media, balancing media reuse with data retention concerns
        - Grandfather/Father/Son (GFS): three or more backup cycles, such as daily, weekly and monthly; the daily backups are rotated on a 3-months basis using a FIFO system, the weekly backups are similarly rotated on a bi-yearly basis, and the monthly backup on a yearly basis
        - Tower of Hanoi: based on the puzzle of the same name, where the first backup is overwritten every other day, the second backup is overwritten every fourth day, and the third backup is overwritten every other day at a different increment than the first backup
        - Six Cartridge Weekly: a method involves six different media (cartridge, tape, drives etc) used for each day of the week; many small businesses that do not need to backup high volumes of data use this type of tape rotation schedule, and

usually consists of using four media for incremental and differential backups between Monday and Thursday

- Backup storage best practices include keeping copies of the media in at least one offsite location to provide redundancy should the primary location be unavailable, incapacitated, or destroyed; common strategy is to store backups in a cloud service that is itself geographically redundant
- Two commmon backup strategies:
  1. full backup on Monday night, then run differential backups every other night of the week
     - if a failure occurs Saturday morning, restore Monday's full backup and then restore only Friday's differential backup
  2. full backup on Monday night, then run incremental backups every other night of the week
     - if a failure occurs Saturday morning, restore Monday's full backup and then restore each incremental backup in the original chronological order

Feature	Full Backup	Incremental Backup	Differential Backup
Description	A complete copy of all selected data	Only backs up data that has changed since the last backup (regardless of type)	Backs up all changes made since the last full backup
Storage Space	Requires the most storage space	Requires the least storage space	Requires more space than incremental but less than full
Backup Speed	Slowest, as it copies all data	Fastest, as it only copies changed data since the last backup	Faster than full but slower than incremental, as it copies all changes since the last full backup
Recovery Speed	Fastest, as all data is in one place	Slowest, as it may require multiple incremental backups to restore to a specific point	Faster than incremental since it requires the last full backup and the last differential backup
Complexity	Simplest, with no dependency on previous backups	Complex, as it depends on a chain of backups from the last full backup to the most recent incremental backup	Less complex than incremental, requires the last full backup and the last differential backup for restoration
Best Use Case	When backup time and storage space are not issues Ideal for less frequent backups	Suitable for environments where daily changes are minimal and quick backups are necessary	Ideal for environments where storage space is a concern but restoration time needs to be relatively quick

- Three main techniques used to create offsite copies of DB content: electronic vaulting, remote journaling, and remote mirroring

        - **electronic vaulting**: where database backups are moved to a remote site using bulk transfers
        - **remote journaling**: data transfers are performed in a more expeditious manner; remote journaling is similar to electronic vaulting in that transaction logs transferred to the remote site are not applied to a live database server but are maintained in a backup device
        - **remote mirroring**: the most advanced db backup solution, and the most expensive, with remote mirroring, a live db server is maintained at the backup site; the remote server receives copies of the db modifications at the same time they are applied to the production server at the primary site

- 7.10.2 Recovery site strategies

    - Non-disaster: service disruption with significant but limited impact
    - Disaster: event that causes an entire site to be unusable for a day or longer (usually requires alternate processing facility)
    - Catastrophe: major disruption that destroys the facility altogether
    - For disasters and catastrophes, an org has 3 basic options:
        - use a dedicated site that the org owns/operates
        - lease a commercial facility (hot, warm, cold site)
        - enter into a formal agreement with another facility/org
    - When a disaster interrupts a business, a disaster recovery plan should kick in nearly automatically and begin providing support for recovery operations -in addition to improving your response capabilities, purchasing insurance can reduce the impact of financial losses
    - Recovery site strategies consider multiple elements of an organization, such as people, data, infrastructure, and cost, as well as factors like availability and location
    - When designing a disaster recovery plan, it’s important to keep your goal in mind — the restoration of workgroups to the point that they can resume their activities in their usual work locations
        - sometimes it's best to develop separate recovery facilities for different work groups
    - To recover your business operations with the greatest possible efficiency, you should engineer the disaster recovery plan so that those business units with the highest priority are recovered first

- 7.10.3 Multiple processing sites

    - One of the most important elements of the disaster recovery plan is the selection of alternate processing sites to be used when the primary sites are unavailable
        - **cold sites**: standby facilities large enough to handle the processing load of an organization and equipped with appropriate electrical and environmental support systems
            - a cold site has NO COMPUTING FACILITIES (hardware or software) preinstalled
            - a cold site has no active broadband comm links
            - advantages:
                - a cold site is the LEAST EXPENSIVE OPTION and perhaps the most practical
            - disadvantages:
                - tremendous lag to activate the site, often measured in weeks, which can yield a false sense of security

      - difficult to test
    - **warm sites**: a warm site is better than a cold site because, in addition to the shell of a building, basic equipment is installed
      - a warm site contains the data links and preconfigured equipment necessary to begin restoring operations, but no usable data for information
      - unlike hot sites, however, warm sites do not typically contain copies of the client's data
      - activation of a warm site typically takes at least 12 hours from the time a disaster is declared
    - **hot sites**: a fully operational offsite data processing facility equipped with hardware and software; a backup facility that is maintained in constant working order, with a full complement of servers, workstations, and comm links
      - a hot site is usually a subscription service
      - the data on the primary site servers is periodically or continuously replicated to corresponding servers at the hot site, ensuring that the hot site has up-to-date data
      - advantages:
        - unsurpassed level of disaster recovery protection
      - disadvanages:
        - extremely costly, likely doubling an org's budget for hardware, software and services, and requires the use of additional employees to maintain the site
        - has (by definition) copies of all production data, and therefore increases your attack surface
    - **Mobile sites**: non-mainstream alternatives to traditional recovery sites; usually configured as cold or warm sites, if your DR plan depends on a workgroup recovery strategy, mobile sites are an excellent way to implement that approach
    - Cloud computing: many orgs now turn to cloud computing as their preferred disaster recovery option
      - some companies that maintain their own datacenters may choose to use these IaaS options as backup service providers
  - Note: A hot site is a subscription service, while a redundant site, in contrast, is a site owned and maintained by the org (and a redudant site may be "hot" in terms of capabilities)
    - the exam differentiates between a hot site (a subscription service) and a redundant site (owned by the organization)

- 7.10.4 System resilience, High Availability (HA), Quality of Service (QoS), and fault tolerance

  - **System resilience**: the ability of a system to maintain an acceptable level of service during an adverse event
  - **High Availability (HA)**: the use of redundant technology components to allow a system to quickly recover from a failure after experiencing a brief disruption
    - **Clustering**: refers to a group of systems working together to handle workloads; often seen in the context of web servers that use a load balancer to manage incoming traffic, and distributes requests to multiple web servers (the cluster)
    - **Redundancy**: unlike a cluster, where all members work together, redundancy typically involves a primary and secondary system; the primary system does all the work, and the

secondary system is in standby mode unless the primary system fails, at which time activity can fail over to the secondary
    - Both clustering and redundancy include high availability as a by-product of their configuration
- **Quality of Service (QoS)**: controls protect the availability of data networks under load
    - many factors contribute to the quality of the end-user experience and QoS attempts to manage all of these factors to create an experience that meets business requirements
    - factors contributing to QoS:
        - bandwidth: the network capacity available to carry communications
        - latency: the time it takes a packet to travel from source to destination
        - packet loss: some packets may be lost between source and destination, requiring re-transmission
        - interference: electrical noise, faulty equipment, and other factors may corrupt the contents of packets
- **Fault tolerance**: the ability of a system to suffer a fault but continue to operate
- **Redundant array of independent disks (RAID)**: refers to multiple drives being used in unison in a system to achieve greater speed or availability; the most well-known RAID levels are:
    - RAID 0—Striping: provides significant speed, writing and reading advantages
    - RAID 1—Mirroring: uses redundancy to provide reliable availability of data
    - RAID 10—Mirroring and Striping: requires a minimum of four drives and provides the benefits of striping (speed) and mirroring (availability) in one solution; this type of RAID is typically one of the most expensive
    - RAID 5—Parity Protection: requires a minimum of three drives and provides a cost-effective balance between RAID 0 and RAID 1; RAID 5 utilizes a parity bit, computed from an XOR operation, for purposes of storing and restoring data

Backup Method	Cost Implications	Time Implications for RPO
Incremental	Lower cost due to reduced storage requirements as only changes are backed up	Longer recovery time as it requires the last full backup plus all subsequent incremental backups until the RPO
Differential	Moderate cost; more storage is needed than incremental, but less than full, as it stores all changes since the last full backup	Faster recovery than incremental as it requires the last full backup and the last differential backup up to the RPO
Replication	Higher cost due to the need for a duplicate environment ready to take over at any time; continuous data replication can also increase bandwidth costs	Minimal recovery time as the data is continuously updated, allowing for near-instant recovery up to the latest point before failure
Clustering	Highest cost because it involves multiple servers (cluster) working together to provide high availability and redundancy	Minimal recovery time as the system is designed for immediate failover without data loss, ensuring the RPO can be met instantaneously

Site Recovery Method	Cost Implications	Time Implications for RTO
Cold Site	Lowest cost option; facilities and infrastructure are available, but equipment and data need to be set up post-disaster	Longest recovery time as systems and data must be configured and restored from backups Suitable for non-critical applications with more flexible RTOs
Warm Site	Moderate cost; a compromise between cold and hot sites, includes some pre-installed hardware and connectivity that can be quickly activated	Faster recovery than a cold site as the infrastructure is partially ready, but data and systems might still need updates to be fully operational
Hot Site	High cost; a duplicate of the original site with full computer systems and near-real-time replication of data and ready to take over operations immediately	Minimal recovery time, designed for seamless takeover with data and systems up-to-date, allowing for critical operations to continue with little to no downtime
Redundant Site	Highest cost; essentially operates as an active-active configuration where both sites are running simultaneously, fully mirroring each other	Instantaneous recovery, as the redundant site is already running in parallel with the primary site, ensuring no interruption in service

7.11 Implement Disaster Recovery (DR) processes (OSG-9 Chpt 18)

- **Business Continuity Management (BCM)**: the process and function by which an organization is responsible for creating, maintaining, and testing BCP and DRP plans
- **Business Continuity Planning (BCP)**: focuses on the survival of the business processes when something unexpected impacts it
- **Disaster Recovery Planning (DRP)**: focuses on the recovery of vital technology infrastructure and systems
  - BCM, BCP, and DRP are ultimately used to achieve the same goal: the continuity of the business and its critical and essential functions, processes, and services
- The key BCP/DRP steps are:
  - Develop contingency planning policy
  - Conduct BIA
  - Identify controls
  - Create contingency strategies
  - Develop contingency plan
  - Ensure testing, training, and exercises
  - Maintenance
- Four key measurements for BCP and DRP procedures:

  - RPO (recovery point objective): max tolerable data loss measured in time
  - RTO (recovery time objective): max tolerable time to recover systems to a defined service level
  - WRT (work recovery time): max time available to verify system and data integrity as part of the resumption of normal ops
  - MTD (max tollerable downtime): max time-critical system, function, or process can be disrupted before unacceptable/irrecoverable consequences to the business

- 7.11.1 Response

  - A disaster recovery plan should contain simple yet comprehensive instructions for essential personnel to follow immediately upon recognizing that a disaster is in progress or imminent
  - Emergency-response plans are often put together in a form of checklists provided to responders; arrange the checklist tasks in order of priority, with the most important task first!
  - The response plan should include clear criteria for activation of the disaster recovery plan, define who has the authority to declare a disaster, and then discuss notification procedures

- 7.11.2 Personnel

  - A disaster recovery plan should contain a list of personnel to contact in the event of a disaster
    - usually includes key members of the DRP team as well as critical personnel
  - Businesses need to make sure employees are trained on DR procedures and that they have the necessary resources to implement the DR plan
  - Key activities involved in preparing people and procedures for DR include:
    - develop DR training programs
    - conduct regular DR drills
    - provid eemployees with necessary resources and tools to implement the DR plan
    - communicate the DR plan to all employees

- 7.11.3 Communications

  - Ensure that response checklists provide first responders with a clear plan to protect life and property and ensure the continuity of operations
    - the notification checklist should be supplied to all personnel who might respond to a disaster

- 7.11.4 Assessment

  - When the DR team arrives on site, one of their first tasks is to assess the situation
    - this normally occurs in a rolling fashion, with the first responders performing a simple assessment to triage the situation and get the disaster response under way
    - as the incident progresses more detailed assessments will take place to gauge effectiveness, and prioritize the assignment of resources

- 7.11.5 Restoration

  - Note that recovery and restoration are separate concepts
  - **Restoration**: bringing a business facility and environment back to a workable state
  - **Recovery**: bringing business operations and processes back to a working state
  - System recovery includes the restoration of all affected files and services actively in use on the system at the time of the failure or crash

    - When designing a disaster recovery plan, it’s important to keep your goal in mind — the restoration of workgroups to the point that they can resume their activities in their usual work locations

- 7.11.6 Training and awareness
    - As with a business continuity plan, it is essential that you provide training to all personnel who will be involved in the disaster recovery effort
    - When designing a training plan consider the following:
        - orientation training for all new employees
        - initial training for employees taking on a new DR role for the first time
        - detailed refresher training for DR team members
        - brief awareness refreshers for all other employees

- 7.11.7 Lessons learned
    - A lessons learned session should be conducted at the conclusion of any disaster recovery operation or other security incident
    - The lessons learned process is designed to provide everyone involved with the incident response effort an opportunity to reflect on their individual roles and the teams overall response
    - Time is of the essence in conducting a lesson learned, before memories fade
    - Usually a lessons learned session is led by trained facilitators
    - NIST SP 800-61 offers a series of questions to use in the lessons learned process:
        - exactly what happened and at what times?
        - how well did staff and management perform in dealing with the incident?
        - were documented procedures followed?
        - were the procedures adequate?
        - were any steps or actions taken that might have inhibited the recovery?
        - what would the staff and management do differently the next time a similar incident occurs?
        - how could information sharing with other organizations have been improved?
        - what corrective actions can prevent similar incidents in the future?
        - what precursors or indicators should be watched for in the future to detect similar incidents?
        - what additional tools or resources are needed to detect, analyze, and mitigate future incidents?
    - The team leader to document the lessons learned in a report that includes suggested process improvement actions

7.12 Test Disaster Recovery Plans (DRP) (OSG-9 Chpt 18)

- Every DR plan must be tested on a periodic basis to ensure that the plan’s provisions are viable and that it meets an org’s changing needs

- Five main test types:
    - checklist tests
    - structured walk-throughs

    - simulation tests
    - parallel tests
    - full-interruption tests

- 7.12.1 Read-through/tabletop
    - **Read-through test**: one of the simplest to conduct, but also one of the most critical; copies of a DR plan are distributed to the members of the DR team for review, accomplishing three goals:
        - ensure that key personnel are aware of their responsibilities and have that knowledge refreshed periodically
        - provide individuals with an opportunity to review and update plans, remvoving obsolete info
        - helps identify situations in which key personnel have left the company and the DR responsibility needs to be re-assigned (note that DR responsibilities should be included in job descriptions)
- 7.12.2 Walkthrough
    - **Structured walk-through**: AKA tabletop exercise, takes testing one step further, where members of the DR team gather in a large conference room and role-play a disaster scenario
        - the team refers to their copies of the DR plan and discuss the appropriate responses to that particular type of disaster
- 7.12.3 Simulation
    - **Simulation tests**: similar to the structured walk-throughs, where team members are presented with a scenario and asked to develop an appropriate response
        - unlike read-throughs and walk-throughs, some of these response measures are then tested
        - this may involve the interruption of noncritical business activities and the use of some operational personnel
- 7.12.4 Parallel
    - **Parallel tests**: represent the next level, and involve relocating personnel to the alternate recovery site and implementing site activation procedures
        - the relocated employees perform their DR responsibilities just as they would for an actual disaster
        - operations at the main facility are not interrupted
- 7.12.5 Full interruption
    - **Full-interruption tests**: operate like parallel tests, but involve actually shutting down operations at the primary site and shifting them to the recovery site
        - these tests involve a significant risk (shutting down the primary site, transfer recovery ops, followed by the reverse) and therefore are extremely difficult to arrange (management resistance to these tests are likely)

7.13 Particpate in Business Continuity (BC) planning and exercises (OSG-9 Chpt 3)

- Business continuity planning addresses how to keep an org in business after a major disruption takes place
    - It's important to note that the scope is much broader than that of DR
    - A security leader will likely be involved, but not necessarily lead the BCP effort
- The BCP life cycle includes:
    - Developing the BC concept
    - Assessing the current environment
    - Implementing continuity strategies, plans, and solutions
    - Training the staff
    - Testing, exercising, and maintaining the plans and solutions

7.14 Implement and manage physical security (OSG-9 Chpt 10)

- Physical access control mechanisms deployed to control, monitor and manage access to a facility
    - Sections, divisions, or areas within a site should be clearly designated as public, private, or restricted with appropriate sinage
- 7.14.1 Perimeter security controls
    - A fence is a perimeter-defining device and can consist of:
        - stripes painted on the ground
        - chain link fences
        - barbed wire
        - concrete walls
        - invisible perimeters using laser, motion, or heat detection
    - **Perimeter intrusion detection and assessment system (PIDAS)**: an advanced form of fencing that has two or three fences used in concert to optimize security
    - **Gate**: controlled exit and entry point in a fence or wall
    - **turnstile**: form of gate that prevents more than one person at a time from gaining entry and often restricts movement in one direction
    - **access control vestibule**: (AKA mantrap) a double set of doors that is often protected by a guard or other physical layout preventing piggybacking and can trap individuals at the discretion of security personnel
    - **Security bollards**: a key element of physical security, which prevent vehicles from ramming access points and entrances
    - **Barricades**: in addition to fencing, are used to control both foot traffic and vehicles
    - Lighting is the most commonly used form of perimeter security control providing the security benefit of deterrence (primary purpose is to discourage casual intruders, trespassers etc)
    - Security guards are able to adapt and react to considtions and situations; guard dogs can be an alternative for perimiter control, functioning as detection and deterrent
    - All physical security controls ultimately rely on personnel to intervene and stop actual intrusions and attacks
    - KPIs (kep performance indicators) of physical security are metrics or measurements of the operation of or failure of key security aspects; they should be monitored, recorded, and evaluated
- 7.14.2 Internal security controls

  - In all circumstances and under all conditions, the most important aspect of security is protecting people
  - Internal security controls include locks, badges, protective distribution systems (PDSs), motion detectors, intrusion alarms, and secondary verification systems
  - If a facility is designed with restricted areas to control physical security, a mechanism to handle visitors is required
  - **Visitor logs**: manual (or automated) list of nonemployee entries or access to a facility/location
    - physical access logs can establish context for interpretation of logical logs
  - Locks: designed to prevent access without proper authorization; a lock is a crude form of an identification and authorization mechanism 7.15 Address personnel safety and security concerns (OSG-9 Chpt 16)

- 7.15.1 Travel

  - Training personnel on safe practices while traveling can increase their safety and prevent security incidents:
    - sensitive data: devices traveling with the employee shouldn't contain sensitive data
    - malware and monitoring devices: possibilities include physical devices being installed in a hotel room of a foreign country
    - free wi-fi: sounds appealing but can be a used to capture a user's traffic
    - VPNs: employers should have access to VPNs that they can use to create secure connections

- 7.15.2 Security training and awareness

  - Orgs should add personnel saftey and security topics to their training and awareness program and help ensure that personnel are aware of duress systems, travel best practices, emergency management plans, and general safety and security best practices
  - Training programs should stress the importance of protecting people

- 7.15.3 Emergency management

  - Emergency management plans and practices help an organization address personnel safety and security after a disaster
  - Safety of personnel should be a primary consideration during any disaster

- 7.15.4 Duress

  - An example of a duress system is a button that sends a distress call
  - Duress systems are useful when personnel are working alone
  - If a duress system is activated accidentally code word(s) can be used to assure responding personnel it was an accident, or omit the word(s) keying an actual response

Domain 8 **Software Development Security**

8.1 Understand and integrate security in the Software Develoment Life Cycle (SDLC) (OSG-9 Chpt 20)

- Domain 8 is focused on helping security pros understand and apply software or application security

  - Applications can present significant risks, and security pros must understand and balance these risks with business requirements and implement appropriate risk mitigation; if a company

    develops custom software, the custom solution can present additional, unique risks and vulns
        - orgs with custom solutions should be on the lookout for logic weaknesses (e.g. buffer overflow vulns), and guard against malicious changes (e.g. backdoors) that can leave the system vulnerable to attacks
    - As software development environments have become increasingly complex, it's important to review this area -- one of the biggest threats to an organization's security

- In this domain you'll learn the basic principles behind securely designing, building, testing, operating and even decomissioning enterprise apps

- Security should be part of the design, and incorporated into the architecture, with the level of protection based on requirements and operating environment

- The Software Development LifeCycle consists of the following steps:

    - Requirements gathering: why create the software, what it will do, and for whom it will be created
    - Design: encapsulating how the software will meet requirements
    - Development: creating/coding the software to meet spec, and integrating with other systems as required
    - Testing: verifying/validating software meets requirements
    - Operations and Maintenance: deploying, and ensuring it's appropriately configured, patched, and monitored

- **Acceptance**: formal, structured hand-off of the completed software system to the customer org; usually involves test, analysis and assessment activites

- **Accreditation**: AKA Security Accreditation a formal declaration by a designated accrediting authority (DAA) that an information system is approved to operate at an acceptable level of risk, absed onthe implementation an approved set of technical, managerial, and procedural safeguards

- **ACID Test**: data integrity provided by means of enforcing atomicity, consistency, isolation, and durability policies

- **Aggregation**: ability to combine non-sensitive data from separate sources to create sensitive info; note that aggregation is a "security issue", where as inference is an attack (where an attacker can pull together peces of less sensitive info to derive info of greater sensitivity)

- **Arbitrary code**: alternate set of instructions and data that an attacker attempts to trick a processor into executing

- **Buffer overflow**: source code vulnerability allowing access to data locations outside of the storage space allocated to the buffer; can be triggered by attempting to input data larger than the size of the buffer

- **Bypass attack**: attempt to bypass front-end controls of a database to access information

- **Certification**: comprehensive technical security analysis of a system to ensure it meets all applicable security requirements

- **CAB**: Change Advisory Board purpose is to review and approve/reject proposed code changes

- **Citizen programmers**: organizational members who codify work-related knowledge, insights, and ideas into (varying degress of) usable software; the process and result is ad hoc, difficult to manage, and usually bereft of security considerations

- **Code protection/logic hiding**: prevents one software unit from reading/altering the source/intermediate/executable code of another software unit

- **Code resuse**: reuse of code, rather than re-invented code means units of software (procedures/objects) means higher productivity toward development requirements using correct, complete, safe code

- **Object/Memory reuse**: systems allocate/release and reuse memory/resources as objects to requesting processes; data remaining in the object when it is reused is a potential security violation (i.e. data remanence)

- **CORBA**: Common Object Request Broker Architecture is a set of standards addressing interoperability between software and hardware products, residing on different machines across a network; providing object location and use across a network

- **Configuration Control**: process of controlling modifications to hardware, firmware, software, and documentation to protect the information system against improper modifications prior to, during, and after system implementation

- **Configuration Management (CM)**: collection of activities focused on establishing and maintaining integrity of IT products and information systems, through the control of processes for initialization, changing and monitoring the configurations of those products and systems throughout the system development lifecycle

- **Covert Channels/Paths**: a method used to pass information over a path that is not normally used for communiction; communication pathways that violate security policy or requirement (deliberately or unwittingly); basic types are timing and storage

- **Data Contamination**: attackers attempt to use malformed inputs, at the field, record, transaction, or file level, in an attempt to disrupt the proper functioning of the system

- **Data Lake**: a data warehouse incorporating multiple types of streams of unstructured or semi-structured data

- **Data Mining**: analysis and decision-making technique that relies on extracting deeper meanings from many different instances and types of data; often applied to data warehouse content

- **Data Modeling**: design process that identifies all data elements that the system will need to input, create, store, modify, output, and destroy during operational use; should be one of the first steps in analysis and design

- **Data Protection and Data Hiding**: restricts or prevents one software unit from reading or altering the private data of another software unit or in preventing data from being discovered or accessed by a subject

- **Data Type Enforcement**: how a language protects a developer from trying to perform operations on dissimilar types of data, or in ways that would lead to erroneous results

- **Data Warehouse**: collection of data sources such as separate internal databases to provide a broader base of info for analysis, trending and reference; may also involve databases from outside the org
- **Data-centric Threat Modeling**: methodology and framework focusing on the authorized movements and data input/output into and from a system; corresponds with protecting data in transit, at rest, and in use when classifying organizational data
- **Defensive Programming**: design/coding allowing acceptable but sanitized data inputs to a system; lack of defensive programming measures can result in arbitrary code execution, misdirection of the program to other resoruces/locations, or reveal info useful to an attacker
- **Design Reviews**: should take place after the development of functional and control specifications but before the creation of code
- **Dirty read**: occurs when one transaction reads a value from a database that was written by another transaction that didn't commit
- **Emerging Properties**: an alternate/more powerful way of looking at systems-level behavior characteristics such as safety and security; helps provide a more testable, measurable answer to questions such as "how secure is our system?"
- **Encapsulation**: note see network Encapsulation (disambiguation); enforcement of data/code hiding during all phases of software development and operational use; bundling together data and methods is the process of encapulation (opposite of unpacking/revealing)
- **Executable/Object Code**: binary representation of the machine language instruction set that the CPU and other hardware of the target computer can directly execute
- **XML**: Extensible Markup Language is a set of HTML extensions providing for data storage and transport in networked environments; frequently used to integrate web pages with databases; XML is often embedded in the HTML files making up elements of a web page
- **Functional requirements**: describes a finite task or process the system must perform; often directly traceable to specific elements in the final system's design and construction
- **Hierarchical database model**: data elements and records are arranged in tree-like parent-child structures
- **Integrated Product and Process Development (IPPD)**: management technique that simultaneously integrates essential acquisition activities through the use of multidisciplinary teams to optimize the design, manufacturing, and supportability processes
- **Integrated Product Team**: team of stakeholders and individuals that possess different skills and who work together to acheive a defined process or product
- **Infrastructure as Code (IaC)**: instead of viewing hardware config as a manual, direct hands-on, one-on-one admin hassel, it is viewed as just another collecti on of elements to be managed in the same way that software and code are managed under DevSecOps

- **Interactive Application Security Testing (IAST)**: testing that combines or integrates SAST and DAST to improve testing and provide behavioral analysis capabilities to pinpoint the source of vulnerabilities
- **Knowledge Discovery in Database (KDD)**: mathematical, statistical, and visualization method of identifying valid and useful patterns in data
- **Knowledge Management**: efficent/effective management of info and associated resources in an enterprise to drive busienss intelligence and decision-making; may include workflow management, business process modeling, doc management, db and info systems and knowledge-based systems
- **Level of abstraction**: how closely a source-code/design doc represents the details of the underlying object/system/component; lower-level abstractions generally have more detail than high-level ones
- **Living off the land** (non-malware based ransom attack): system attack where the system/resources compromised are used in pursuit of additional attacks (i.e. the attacker's agenda); anti-malware defence doesn't detect/prevent the attack given the attacker's methodology
- **Malformed input attack**: not currently handling input data is a common source of code errors that can result in arbitrary code exec, or misdirection of the program to other resources/locations
- **Markup Language**: non-programming language used to express formatting or arrangement of data on a page/screen; usually extensible, allowing users to define additional/other operations to be performed; they etend the language into a programming language (e.g. in the same way JavaScript extends HTML)
- **Metadata**: info that describes the format or meaning of other data, which can be used to provide a systematic method for describing resources and improving info retrieval
- **Mobile code (executable content)**: file(s) sent by a system to others, that will either control the execution of systems/applications on that client or be directly executed
- **Modified prototype model**: approach to system design/build that starts with a simplified version of the application; feedback from stakeholders is used to improve design of a second version; this is repeated until owners/stakeholders are satisfied with the final product
- **Network database model**: database model in which data elements and records are arranged in arbitrary linked fashion (.e.g lists, clusters, or other network forms)
- **Nonfunctional requirements**: broad characteristics that do not clearly align with system elements; many safety, security, privacy, and resiliency can be deemed nonfunctional
- **Object**: encapsulation of a set of data and methods that can be used to manipulate that data
- **Object-oriented database model**: database model that uses object-oriented programming concepts like classes, instances, and objects to organize, structure, and store data and methods; schemas define the structure of the data, views specify table, rows, and columns that meet user/security requirements
- **Object-oriented security**: systems security designs that make sue of object-oriented programming characteristics such as encapsulation, inheritance, polymorphism, and polyinstantiation

- **Open-source software**: source code and design info is made public, and often using licenses that allow modification and refactoring
- **Pair programming**: requires two devs to work together, one writing code, and the other reviewing and tracking progress
- **Pass-around reviews**: often done via email or code review system, allows devs to review code asynchronously
- **PERT**: chart that uses nodes to represent milestones or deliverables, showing the estimated to to move between milestones
- **Polyinstantiation**: creates a new instance (copy) of a data item, with the same identifier or key, allowing each process to have its own version of that data; useful for enforcing and protecting different security levels for a shared resource; polyinstantiation also allows the storage of multiple different pieces of info in a database at different classification levels to prevent attackers from inferring anything about the absence of info
- **Procedural programming**: emphasizes the logical sequence of steps to be peformed, where a procedure is a set of software that performs a particular function, requiring specific input data, producing a specific set of outputs, and procedures can invoke other procedures
- **Query attack**: use of query tools to access data not normally allowed by the trusted front end, including the views controlled by the query application; could also result from malformed queries using SQL to bypass security controls; improper/incomplese checks on queries can be used in a similar way to bypass access controls
- **Ransom attack**: form of attack that threatens destruction, denial, or unauthorized public release/remarketing of private information assets; usually involves encrypting assets and withhold the decryption key until a ransom is paid by the victim
- **Refactoring**: partial or complete rewrite of a set of software to perform the same functions, but in a more straightforward, more efficient, or more maintainable form
- **Regression testing**: test a system to ascertain whether recently approved modifications have changed performance of other approved functions or introduced other unauthorized behavior;testing that runs a set of known inputs against an app and compares to results previously produced (by an earlier version of the software)
- **Relational database model**: data elements and records arragned in tables which are related or linked to each other to implement business logic, where data records of different structures or types are needed together in the same activity
- **Representational State Transfer (REST)**: software architectural style for synchronizing the activities of two or more apps running on different systems on a network; REST facilitiates these processes exchanging state information, usually via HTTP/S
- **Reputation monitoring**: defensive tactic that uses the trust reputation of a website or IP address as a means of blocking an org's users, processes or systems from connecting to a possible source of malware or exploitations; possibly the only real defense against zero-day exploits; involves monitoring URLs, domains, IP addresses or other similar info to separate untrustworthy traffic

- **Runtime Application Security Protection (RASP)**: security agents comprised of small code units built into an app which can detect set of security violations; upon detection, the RASP agent can cause the app to terminate, or take other protective actions
- **Security Assessment**: testing, inspection, and analysis to determine the degree to which a system meets or exceeds the required security posture; may assess whether an as-built system meets the requirements in its specs, or whether an in-use system meets the current perception of the real-world security threats
- **Software Quality Assurance**: variety of formal and informal processes that attempt to determine whether a software app or system meets all of its intended functions, doesn't perform unwanted functions, is free from known security vulns, and is free from insertion or other errors in design and function
- **SDLC**: Software Development LifeCycle is a framework and systematic associated with tasks that are performed in a series of steps for building, deploying, and supporting software apps; begins with planning and requirements gathering, and ends with decommissioning and sunsetting; there are many different SDLCs, such as agile, DevSecOps, rapid prototyping, offering different approaches to defining and managing the software lifecycle
- **Source code**: program statements in human-readable form using a formal programming language's rules for syntax and semantics
- **Spyware/Adware**: software that performas a variety of monitoring and data gathering functions; AKA potentailly unwanted programs/applications (PUP/PUA), may be used in monitoring employee activities/use of resources (spyware), or advertising efforts (adware); both may be legit/authorized by system owners or unwanted intruders
- **Strong data typing**: feature of a programming language preventing data type mismatch errors; strongly typed languages will generate errors at compile time
- **Threat surface**: total set of penetrations of a boundary or perimeter that surrounds or contains system elements
- **TOCTOU attack**: time of check vs time of use (TOCTOU) attack takes advantage of the time delay between a security check (such as authentication or authorization) being performed and actual use of the asset
- **Trapdoor/backdoor**: AKA maintenance hook; hidden mechanism that bypasses access control measures; an entry point into an architecture or system that is inserted in software by devs during development to provide a method of gaining access for modification/support; can also be inserted by an attacker, bypassing access control measures designed to prevent unauthorized software changes
- **UAT**: User Acceptance Testing typically the last phase of the testing process; verifies that the solution developed meets user requirements, and validates against use cases
- 8.1.1 Development methodologies (e.g. Agile, Waterfall, DevOps, DevSecOps)
  - **Agile methodology**: a project management approach to development that involves breaking the project into phases and emphasizes continuous collaboration and improvement; teams follow a cycle of planning, executing, and evaluating

- Agile development emphasizes:
  - the delivery of working software in short iterations, helping to get the software to market faster
  - reduced risk by frequently testing and providing feedback, helping to identify and resolve issues earlier in the development process
- Agile was started by 17 pioneers in 2001, producing the "Manifesto for Agile Software Development" (agilemanifesto.org) that lays out the core philosophy of the Agile approach:
  - **individuals and interactions** over processes and tools
  - **working software** over comprehensive documentation
  - **customer collaboration** over contract negotiation
  - **responding to change** over following a plan
- Agile Manifesto also defines 12 principles:
  - the highest priority is to satisfy the customer through early and continuous delivery of valuable software
  - welcome changing requirements, even late in development; Agile processes harness change for the customer’s competitive advantage
  - deliver working software frequently, from a couple of weeks to a couple of months, with a preference for the shorter timescale
  - business people and developers must work together daily throughout the project
  - build projects around motivated individuals; give them the environment, support, and tools and trust them to build
  - emphasizing face-to-face conversation
  - working software is the primary measure of progress
  - agile processes promote sustainable development; the them should be able to maintain a constant pace indefinitely
  - continuous attention to technical excellence and good design enhances agility
  - simplicity, or the art of maximizing the amount of work not done, is essential
  - the best architectures, requirements, and designs emerge from self-organizing teams
  - at regular intervals, the team reviews their effective and adjusts for improvement
- Several methodologies have emerged that take these Agile principles and define specific processes around them:
  - **Scrum**: a management framework that teams use to self-organize and work towards a common goal; it describes a set of meetings, tools, and roles for efficient project delivery, allowing teams to self-manage, learn from experience, and adapt to change; named from the daily team meetings, called scrums
  - **Kanban**: a visual system used to manage and keep track of work as it moves through a process; the word kanban is Japanese for "card you can see"; Kanban teams focus on reducing the time a project (or user story) takes from start to finish, using a kanban board and continuously improving their flow of work
  - **Rapid Application Development (RAD)**: an agile software development approach that focuses more on ongoing software projects and user feedback and less on following a strict plan, emphasizing rapid prototyping over planning; RAD uses four phases: requirements planning, user design, construction, and cutover

        - **Rational Unified Process (RUP)**: an agile software development methodology that splits the project life cycle into four phases:
            - inception: which defines the scope of the project and develop business case
            - elaboration: Plan project, specify features, and baseline the architecture
            - construction: Building the product
            - transition: providing the product to its users
            - during each of the phases, all six core development disciplines take place: business modeling, requirements, analysis and design, implementation, testing, and deployment
        - **Agile Unified Process (AUP)**: a simplified version of the rational unified process, it describes a simple, easy to understand approach to developing business application software using agile techniques and concepts yet still remaining true to the RUP
        - **Dynamic Systems Development Model (DSDM)**: an agile project delivery framework, initially used as a software development method; key principles:
            - focus on the business need: DSDM teams establish a valid business case and ensure organizational support throughout the project
            - deliver on time: work should be time-boxed and predictable, to build confidence in the development team
        - **Extreme Programming (XP)**: an Agile project management methodology that targets speed and simplicity with short development cycles, using five guiding values, five rules, and twelve practices for programming; the goal of the rigid structure, focused sprints and continuous integrations is higher quality product
        - **Scaled Agile Framework® (SAFe)**: a set of org and workflow patterns for implementing agile practices at an enterprise scale; the framework is a body of knowledge that includes structured guidance on roles and responsibilities, how to plan and manage the work, and values to uphold
- **Waterfall**:
    - Developed by Winston Royce in 1970, the waterfall model uses a linear sequential life-cycle approach where each phase must be completed before the next can begin; all project requirements are gathered up front, and there is no formal way to integrate changes as more information becomes available
    - Traditional model has 7 stages, as each stage is completed, the project moves into the next phase; the iterative waterfall model does allow development to return to the previous phase to correct defects
        - System requirements
        - Software requirements
        - Preliminary design
        - Detailed design
        - Code and debug
        - Testing
        - Operations and maintenance
    - A major criticism of this model is that it's very rigid, and not ideal for most complex projects which often contain many variables that affect the scope throughout the project's lifecycle

    - **Spiral model**: improved waterfall dev process providing for a cycle of Plan, Do, Check, Act (PDCA) sub-stages at each phase of the SDLC; a risk-driven development process that follows an iterative model while also including waterfall elements
        - following defined phases to completion and then repeats the process, resembling a spiral
        - the spiral model provides a solution to the major criticism of the waterfall model in that it allows devs to return to planning stages as technical demands and customer requirements iterate
    - **DevOps (Development and Operations)**: an approach to software development, quality assurance, and technology operations that seeks to unite siloed staff, and bring the three functions together in a single operational model
        - closely aligned with lean and the Agile development approach, DevOps aims to dramatically decrease the time required to develop, test, and deploy software changes
        - using the DevOps model, and continuous integration/continuous delivery (CI/CD), orgs strive to roll out code dozens or even hundreds of times per day
        - this requires a high degree of automation, including integrating code repositories, the software configuration management process, and the movement of code between development, testing and production environments
        - the tight integration of development and operations also calls for the simultaneous integration of security controls
        - security must be tightly integrated and move with the same agility
    - **DevSecOps**: refers to the integration of development, security, and operations
        - provides for a merger of phased review (as in the waterfall SDLC) with the DevOps method, to incorporate the needs for security, safety, resilience or other emerging properties in the final system, at each turn of the cycle of development
        - DevSecOps supports the concept of software-defined security, where security controls are actively managed into the CI/CD pipeline

- 8.1.2 Maturity models (e.g., Capability Maturity Model (CMM), Software Assurance Maturity Model (SAMM))

    - Software Engineering Institute (SEI) (Carnegie Mellon University) created the Capability Maturity Model for Software (AKA Software Capability Maturity Model, abbreviated SW-CMM, CMM, or SCMM)

        - **SW-CMM**: a management process to foster the ongoing and continuous improvement of an org's processes and workflows for developing, maintaining and using software
        - all software development moves through a set of maturity phases in sequential fashion, and CMM describes the principles and practices underlying software process maturity, intended to help improve the maturity and quality of software processes
        - note that CMM doesn't explicitly address security
        - stages of the CMM:
            - Level 1: Initial: process is disorganized; usually little or no defined software development process
            - Level 2: Repeatable: in this phase, basic lifecycle management processes are introduced

        - Level 3: Defined: in this phase, software devs operate according to a set of formal, documented software development processes; marked by the presence of basic liefcycle management processes and reuse of code; includes the use of requirements management, software project planning, quality assurance, and configuration management
        - Level 4: Managed: in this phase, there is better management of the software process; characterized by the use of quantitative software development measures
        - Level 5: Optimizing: in this phase continuous improvement occurs

- **Software Assurance Maturity Model (SAMM)**: an open source project maintained by the Open Web Application Security Project (OWASP)

    - provides a framework for integrating security into the software development and maintenance processes and provides orgs with the ability to assess their maturity
    - SAMM associates software development with 5 business functions:
        - Governance: the activities needed to manage software development processes
            - this function includes practices for:
                - strategy
                - metrics
                - policy
                - compliance
                - education
                - guidance
        - Design: process used to define software requirements and develop software
            - this function includes practices for:
                - threat modeling
                - threat assessment
                - security requirements
                - security architecture
        - Implementation: process of building and deploying software components and managing flaws
            - this function includes:
                - secure build
                - secure deployment
                - defect management practices
        - Verification: activities undertaken to confirm code meets business and security requirements
            - this function includes:
                - architecture assessment
                - requirements-driven testing
                - security testing
        - Operations: actions taken to maintain security throughout the software lifecycle after code is released
            - function includes:
                - incident management
                - environment management
                - operational management

  - **IDEAL Model**: developed by SEI, a model for software development that uses many of the SW-CMM attributes, using 5 phases:
    - Initiating: business reasons for the change are outlined, support is built, and applicable infrastructure is allocated
    - Diagnosing: in this phase, engineers analyze the current state of the org and make general recommendations for change
    - Establishing: development of a specific plan of action based on the diagnosing phase recommendations
    - Acting: in this phase, the org develops solutions and then tests, refines, and implements them
    - Learning: continuously analyze efforts to achieve these goals, and propose new actions as required
  - IDEAL vs SW-CMM:

IDEAL	SW-CMM
Initiating	Initial
Diagnosing	Repeatable
Establishing	Defined
Acting	Managed
Learning	Optimizing

- 8.1.3 Operations and maintenance
  - Once delivered to the production environment, software devs must make any additional changes to accomodate unexpected bugs, vulnerabilities, or interoperability issues
  - They must also keep pace with changing business processes, and work closely with the operations team (typically IT), to ensure reliable operations
    - together, ops and development transition a new system to production and management of the system's config
  - The dev team must continually provide hotfixes, patches, and new releases to address discovered security issues and identified coding errors
- 8.1.4 Change management
  - Change management (AKA control management) plays an important role when monitoring systems in a controlled environment, and has 3 basic components:
    - **Request Control**: process that provides an organized framework within which users can request modifications, managers can conduct cost/benefit analysis, and developers can prioritize tasks
    - **Change Control**: the process of controlling specific changes that need to take place during the life cycle of a system, serving to document the necessary change-related activities; or the process of providing an organized framework within which multiple devs can create and test a solution prior to rolling it out in a production environment

- where change management is the project manager’s responsibility for the overarching process, change control is what devs do to ensure the software or environment doesn’t break when changed
- change control is basically the process used by devs to re-create a situation encountered by a user and analyze the appropriate changes; it provides a framework where multiple devs can create and test a solution prior to rolling it out into a prod environment
- **Release Control**: once changes are finalized, they must be approved for release through the release control procedure
  - one of the responsibilities of release control is ensuring that the process includes acceptance testing, confirming that any alterations to end-user work tasks are understood and functional prior to code release

- 8.1.5 Integrated Product Team (IPT)
  - **Integrated Product Team (IPT)**:Introduced by the DoD as an approach to bring together multifunctional teams with a single goal of delivering a product or developing a process or policy, and fostering parallel, rather than sequential, decisions
  - Essentially, IPT is used to ensure that all aspects of a product, process, or policy are considered during the development process

8.2 Identify and apply security controls in software development ecosystems (OSG-9 Chpts 15,17,20,21)

- Applications, including custom systems, can present significant risks and vulnerabilities, and to protect against these it's important to introduce security controls into the entire system’s development lifecycle

- 8.2.1 Programming languages
  - Computers understand 1s and 0s (binary), and each CPU has its own (machine) language
  - **Assembly language**: a way of using mnemonics to represent the basic instruction set of a CPU
  - **Assemblers**: tools that convert assembly language source code into machine code
  - Third-generation programming languages, such as C/C++, Java, and Python, are known as high-level languages
    - high-level languages allow developers to write instructions that better approximate human communication
  - **Compiled language**: converts source code into machine-executable format
    - compiled code is generally less prone to manipulation by a third party, however easier to embed backdoors or other security flaws without detection
  - **Decompilers**: convert binary executable back into source code
  - **Disassemblers**: convert back into machine-readable assembly language (an intermediate step during the compilation process)
  - **Interpreted language**: uses an interpreter to execute;sourcecode is viewable; e.g. Python, R, JavaScript, VBScript
  - **Object-oriented programming (OOP)**: defines an object to be set of a software that offers one or more methods, internal to the object, that software external to that object can request

to access; each method may require specific inputs and resources and may produce a specified set of outputs; focuses on the objects involved in an interaction

- OOP languages include C++, Java, and .NET
- think of OOP as a group of objects that can be requested to perform certain operations or exhibit certain behaviors, working together to provide a system's functionality or capabilities
- OOP has the potential to be more reliable and to reduce the propagation of program change errors, and is better suited to modeling or mimicking the real world
- each object in the OOP model has methods that correspond to specific actions that can be taken on the object
- objects can also be subclasses of other objects and inherit methods from their parent class; the subclasses can use all the methods of the parent class and have additional class-specific methods
- from a security standpoint, object-oriented programming provides a black-box approach to abstraction
- OOP terms:
  - **message**: a communication to or input of an object
  - **method**: internal code that defines the actions of an object
  - **behavior**: results or output exhibited by an object
    - behaviors are the results of a message being processed through a method
  - **class**: a collection of the common methods, from a set of objects that defines the behavior of those objects
  - **instance**: objects are instances of or examples of classes that contain their methods
  - **inheritance**: occurs when the methods from a class (parent or superclass) are inherited by another subclass (child) or object
  - **delegation**: the forwarding of a request by an object to another object or delegate
  - **polymorphism**: the characteristic of an object that allows it to respond with different behaviors to the same message or method because of changes in external conditions
  - **cohesion**: describes the strength of the relationship between the purposes of the methods within the same class
    - if all methods have similar purposes, there is high cohesion, and a sign of good design
  - **coupling**: the level of interaction between objects
    - lower coupling: means less interaction
    - lower coupling provides better software design because objects are more independent, and code is easier to troubleshoot and update

- 8.2.2 Libraries

  - **Software library**: a pre-written collection of components (classes, procedures, scripts etc) that do specific tasks, useful to other components (e.g. software libraries for encryption algorithms, managing network connections, or displaying graphics)
  - Shared software libraries contain reusable code, improving developer's efficiency, and reducing the need to write well-known algorithms from scratch; often available as open source

        - shared libraries can also include many security issues (e.g. Heartbleed), and devs should be aware of the origins of the shared code that they use, and keep informed about any security vulns that might be discovered in these libraries

- 8.2.3 Tool sets

    - Forcing all devs to use the same toolset can reduce productivity and job satisfaction; however letting every dev choose their own tools and environment widens an organization's attack surface
        - a better approach is to use a change advisory board to validate developer tool requirements, assess associated risks; if approved, the sec team monitors controls
    - Developers use a variety of tools, and one of the most important is the IDE (defined below)

- 8.2.4 Integrated Development Environment (IDE)

    - **Integrated Development Environment (IDE)**: software applications, their control procedures, supporting databases, libraries and toolsets that provide a programmer or team what they need to specify, code, compile, test, and integrate code; IDEs provide developers with a single environment where they can write their code, test and debug, and compile it

- 8.2.5 Runtime

    - **RunTime Environments (RTE)**: allows the portable execution of code across different operating systems or platforms without recompiling (e.g. Java Virtual Manager (JVM))
        - this is known as portable code, which needs translation between each environment, the role of the RTE

- 8.2.6 Continuous Integration and Continuous Delivery (CI/CD)

    - **Continuous Integration and Continuous Delivery**: workflow automation processes and tools that attempt to reduce, if not eliminate, the need for manual communication and coordination between the steps of a software development process
    - **Continuous integration (CI)**: all new code is integrated into the rest of the system as soon as the developer writes it, merging it into a shared repo
        - this merge triggers a batch of unit tests
        - if it merges without error, it's subjected to integration tests
        - CI improves software development efficiency by identifying errors early and often
        - CI also allows the practice of continuous delivery (CD)
    - **Continuous Delivery (CD)**: incrementally building a software product that can be released at any time; because all processes and tests are automated, code can be released to production daily or more often
    - CI/CD relies on automation and often third-party tools which can have vulnerabilities or be compromised
    - Secure practices such as threat modeling, least privilege, defense in depth, and zero trust can help reduce possible threats to these tools and systems

- 8.2.7 Security Orchestration, Automation, and Response (SOAR)

    - **Security Orchestration, Automation, and Response (SOAR)**: refers to a group of technologies that allow orgs to respond to some incidents automatically

    - **Playbook**: a document or checklist that defines how to verify an incident
    - **Runbook**: implements the playbook data into an automated tool
    - SOAR allows security admins to define these incidents and the response, typically using playbooks and runbooks
    - Both SOAR and SIEM platforms can help detect and, in the case of SOAR, respond to threats against your software development efforts
        - devs can be resistent to anything that slows down the development process, and this is where DevSecOps can help build the right culture, and balance the needs of developers and security

- 8.2.8 Software Configuration Management (SCM)

    - **Software Configuration Management (SCM)**: a product that identifies the attributes of software at various points in time and performs methodical change control for the purpose of maintaining software integrity and traceability throughout the SDLC
        - SCM tracks config changes, and verifies that the delivered software includes all approved changes
        - SCM systems manage and track revisions made by multiple people against a single master software repository, providing concurrency management, versioning, and synchronization

- 8.2.9 Code repositories

    - Software development is a collaborative effort, and larger projects require teams of devs working simultaneously on different parts
    - Code repositories support collaborations, acting as a central storage point for source code
        - github, bitbucket, and sourceforge are examples of systems that provide version control, bug tracking, web hosting, release management, and communications functionality

- 8.2.10 Application security testing (e.g., Static Application Security Testing (SAST), Dynamic Application Security Testing (DAST))

    - **Static Application Security Testing (SAST)**: AKA static analysis, tools and technique to help identify software defects (e.g. data type errors, loop/structure bounds violations, unreachable code) or security policy violations and is carried out by examining the code without executing the program (or before the program is compiled)
        - the term SAST is generally reserved for automated tools that assist analysts and developers, whereas manual inspection by humans is generally referred to as code review
        - SAST allows devs to scan source code for flaws and vulns; it also provides a scalable method of security code review and ensuring that devs are following secure coding policies
    - **Dynamic Application Security Testing (DAST)**: AKA dynamic analysis, is the evaluation of a program while running in real time
        - tools that execute the software unit, application or system under test, in ways that attempt to drive it to reveal a potentially exploitable vulnerability
        - DAST is usually performed once a program has cleared SAST and basic code flaws have been fixed

        - DAST enables devs to trace subtle logical errors that are likely to cause security problems, without the need to create artificial error-inducing scenarios
        - dynamic analysis is also effective for compatibility testing, detecting memory leakages, identifying dependencies, and analyzing software without accessing the software's actual source code

8.3 Assess the effectiveness of software security (OSG-9 Chpts 20,21)

- 8.3.1 Auditing and logging of changes
    - Applications should be configured to log details of errors and other security events to a centralized log repository
    - The Open Web Application Security Project (OWASP) Secure Coding Practices suggest logging the following events:
        - input validation failures
        - authentication attempts, especially failures
        - access control failures
        - tampering attempts
        - use of invalid or expired session tokens
        - exceptions raised by the OS or applications
        - use of admin privileges
        - Transport Layer Security (TLS) failures
        - cryptographic errors
- 8.3.2 Risk analysis and mitigation
    - Risk management is at the center of secure software development, in particular regarding the mapping of identified risks and implemented controls
        - this is a difficult part of secure software dev, especially related to auditing
    - Threat modeling is important to dev teams, and particularly in DevSecOps
    - Assessors are also interested in the linkages between the software dev and risk management programs
        - software projects should be tracked in the org's risk matrix, to ensure the dev team is connected to the broader risk management efforts, and not working in isolation

8.4 Assess security impact of acquired software (OSG-9 Chpts 16,20)

- 8.4.1 Commercial-off-the-shelf (COTS)

    - **Commercial Off-the-Shelf (COTS)**: software elements, usually apps, that are provided as finished products (not intended for alteration by or for the end-user)
    - Most widely used commercial-off-the-shelf (COTS) software products have been security researcher (both benign and malicious) tested
        - researching discovered vulnerabilities and exploits can help us understand how seriously the vendor takes security
        - for niche products, you should research vendor certifications, such as ISO/IEC 27034 Application Security
        - other than secure coding certification, you can look for overall information security management system (ISMS) certifications such as ISO/IEC 27001 and FedRAMP (which are difficult to obtain, and show that the vendor is serious about security)

  - If you can talk with a vendor, look for processes like defensive programming, which is a software development best practice that means as code is developed or reviewed, they are constantly looking for opportunities for things to go badly
    - e.g. treating all input routines as untrusted until proven otherwise

- 8.4.2 Open source

  - Open source is typically released with licensing allowing code access and inspection so devs can look for security issues
    - typically, however, this means that there is no sevice or support that comes with the software and requires in-house support for configuration, and security testing
    - it also means that both open-source devs as well as adversaries are able to review the code for vulns
    - the greatest risk of open-source software is relying on outdated versions -- especially true of shared libraries
    - an org should develop processes to ensure that all open-source software is periodically updated, likely in a way that differs from the process for updating COTS

- 8.4.3 Third-party

  - **Third-party software**: (AKA outsourced software) is software made specifically for an org by a third party
    - third-party software is not considered COTS, since the software is custom or customized
    - third-party software may rely on open-source software, but since it's customized, it may have different or additional vulns
    - it's best practice to use a third-party to do an external audit and security assessment; this should be built into the vendor's contract, with passing the audit conditional for finalizing software purchase

- 8.4.4 Managed services (e.g. Software as a Service (SaaS), Infrastructure as a Service (IaaS), Platform as a Service (PaaS))

  - As orgs continue to migrate to the cloud (SaaS, IaaS, PaaS), they should increase the security assessment of those services
  - The top reasons for cloud breaches continues to be misconfigurations, lack of visibility into access settings, and poor access controls
    - cloud service providers have tools to help mitigate these issues, and orgs should consider bringing in third-party experts to help if they don't have the internal expertise

8.5 Define and apply secure coding guidelines and standards (OSG-9 Chpts 20,21)

- **Secure Coding Guidelines and Standards**: best practices identified by a variety of software and security professionals, that when used correctly can dramatically reduce the number of exploitable vulnerabilities introduced during development that remain in the operationally-deployed system

- 8.5.1 Security weaknesses and vulnerabilities at the source-code level

  - A source code vulnerability is a code defect providing a threat actor with an opportunity to compromise the security of a software system

        - source code vulns are caused by design or implementation flaws
        - **design flaw**: if dev did everything correctly, there would still be a vulnerability
        - **implementation flaw**: dev incorrectly implemented part of a good design
    - the OWASP top 10 vulnerabilities for 2021:
        - Broken access control
        - Cryptographic failures
        - Injection
        - Insecure design
        - Security misconfiguration
        - Vulnerable and outdated components
        - Identification and authentication failures
        - Software and data integrity failures
        - Security logging and monitoring failures
        - Server Side Request Forgery (SSRF)

- 8.5.2 Security of Application Programming Interfaces (APIs)

    - **Application Programming Interface (API)**: specifies the manner in which a software component interacts with other components
        - API's reduce the effort of providing secure component interactions by providing easy implementation for security controls
        - API's reduce code maintenance by encouraging software resue, and keeping the location of changes in one place
        - **Parameter validation**: ensuring that any API parameter is checked against being malformed, invalid, or malicious helps ensure API secure use; validation confirms that the parameter values being received by an app are within defined limits before they are processed by the system

- 8.5.3 Security coding practices

    - Secure coding practices can be summarized as standards and guidelines
        - **standards**: mandatory activities, actions, or rules
        - **guidelines**: recommended actions or ops guidelines that provide flexibility for unforeseen circumstances
        - orgs greatly reduce source code vulns by enforcing secure coding standards and maintaining coding guidelines that reflect best practices
    - To be considered a standard, coding practice must meet the following:
        - reducees the risk of a particular type of vuln
        - enforceable across all of an org's software development efforts
        - verifiably implemented
    - Note: secure coding standards, rigorously applied, is the best way to reduce source code vulns; coding standards ensures devs always do certain things in a certain way, while avoiding others
    - Secure coding guidelines are recommended practices that tend to be less specific than standards - e.g. consistently formatted code comments, or keeping code funtions short/tight

- 8.5.4 Software-defined security

- **Software-defined security (SDS or SDSec)**: a security model in which security functions such as firewalling, IDS/IPS, and network segmentation are implemented in software within an SDN environment
    - one of the advantages of this approach is that sensors (for systems like IDS/IPS) can be dynamically repositioned depending on the threat
    - SDS provides a decoupling from physical devices, because it abstracts security functions into software that can run on any compatible physical or virtual infrastructure, critical for supporting cloud services dynamic scaling and virtualized data centers
- DevSecOps supports the concept of software-defined security, where security controls are actively managed into the CI/CD pipeline

---