EthicalHacking

Ethical Hacking Essentials (EHE)

Table of Contents

Module 01: Information Security Fundamentals

- Information Security Fundamentals
  - Information security is a state of well-being of information and infrastructure in which the possibility of theft, tampering, and disruption of information and services is low or tolerable
  - Need for Security
    - Evolution of technology, focused on ease of use
    - Rely on the use of computers for accessing, providing, or just storing information
    - Increased network environment and network-based applications
    - Direct impact of security breach on the corporate asset base and goodwill
    - Increasing complexity of computer infrastructure administration and management
  - Elements of Information Security
    - Confidentiality
    - Integrity
    - Availability
    - Authenticity
    - Non-repudiation
  - The Security, Functionality, and Usability Triangle
    - Security: Restrictions

    - Functionality: Features
    - Usability: GUI

  - Security Challenges

    - Compliance to government laws and regulations
    - Lack of qualified and skilled cybersecurity professionals
    - Difficulty in centralizing security in a distributed computing environment
    - Difficulty in overseeing end-to-end processes due to complex IT infrastructure
    - Fragmented and complex privacy and data protection regulations
    - Use of a serverless architecture and applications that rely on third-party cloud providers
    - Compliance issues and issues with data removal and retrieval due to the implementation of Bring Your Own Device (BYOD) policies in companies
    - Relocation of sensitive data from legacy data centers to the cloud without proper configuration
    - Weak links in supply-chain management
    - Increase in cybersecurity risks such as data loss and unpatched vulnerabilities and errors due to the usage of shadow IT
    - Shortage of research visibility and training for IT employees

  - Motives, Goals, and Objectives of Information Security Attacks

    - Attacks = Motive (Goal) + Method + Vulnerability
    - A motive originates out of the notion that the target system stores or processes something valuable, and this leads to the threat of an attack on the system
    - Attackers try various tools and attack techniques to exploit vulnerabilities in a computer system or its security policy and controls in order to fulfil their motives
    - Motives behind information security attacks
      - Disrupting business continuity
      - Stealing information and manipulating data
      - Creating fear and chaos by disrupting critical infrastructures
      - Causing financial loss to the target
      - Damaging the reputation of the target

  - Classification of Attacks

    - Passive Attacks
      - Do not tamper with the data and involve intercepting and monitoring network traffic and data flow on the target network
      - Examples include sniffing and eavesdropping, Footprinting, Network traffic analysis and Decryption of weakly encrypted traffic
    - Active Attacks
      - Tamper with the data in transit or disrupt the communication or services between the systems to bypass or break into secured systems
      - Examples include DoS, Man-in-the-Middle, session hijacking, and SQL injection
    - Close-in Attacks
      - Are performed when the attacker is in close physical proximity with the target system or network in order to gather, modify, or disrupt access to information

        - Examples include social engineering such as eavesdropping, shoulder surfing, and dumpster diving
    - Insider Attacks
        - Involve using privileged access to violate rules or intentionally cause a threat to the organization’s information or information systems
        - Examples include theft of physical devices and planting keyloggers, backdoors, and malware
    - Distribution Attacks
        - Occur when attackers tamper with hardware or software prior to installation
        - Attackers tamper with the hardware or software at its source or in transit

- Information Security Attack Vectors

    - Cloud Computing Threats:
        - Cloud computing is an on-demand delivery of IT capabilities where sensitive data of organizations, and their clients is stored. Flaw in one client’s application cloud allow attackers to access other client’s data
    - Advanced Persistent Threats (APT):
        - An attack that is focused on stealing information from the victim machine without the user being aware of it
    - Viruses and Worms:
        - The most prevalent networking threat that are capable of infecting a network within seconds
    - Ransomware:
        - Restricts access to the computer system’s files and folders and demands an online ransom payment to the malware creator(s) in order to remove the restrictions
    - Mobile Threats:
        - Focus of attackers has shifted to mobile devices due to increased adoption of mobile devices for business and personal purposes and comparatively lesser security controls
    - Botnet
        - A huge network of the compromised systems used by an intruder to perform various network attacks
    - Insider Attack
        - An attack performed on a corporate network or on a single computer by an entrusted person (insider) who has authorized access to the network
    - Phishing
        - The practice of sending an illegitimate email falsely claiming to be from a legitimate site in an attempt to acquire a user’s personal or account information
    - Web Application Threats
        - Attackers target web applications to steal credentials, set up phishing site, or acquire private information to threaten the performance of the website and hamper its security
    - IoT Threats
        - IoT devices include many software applications that are used to access the device remotely

            - Flaws in the IoT devices allows attackers access into the device remotely and perform various attacks

- Information Security Laws and Regulations

    - Payment Card Industry Data Security Standard (PCI DSS) https://www.pcisecuritystandards.org

        - A proprietary information security standard for organizations that handle cardholder information for major debit, credit, prepaid, e-purse, ATM, and POS cards
        - PCI DSS applies to all entities involved in payment card processing — including merchants, processors, acquirers, issuers, and service providers, as well as all other entities that store, process, or transmit cardholder data
        - PCI Data Security Standard, High Level Overview:
            - Build and Maintain a Secure Network
                - Install and maintain a firewall configuration to protect cardholder data
                - Do not use vendor-supplied defaults for system passwords and other security parameters
            - Protect Cardholder Data
                - Protect stored cardholder data
                - Encrypt transmission of cardholder data across open, public networks
            - Maintain a Vulnerability Management Program
                - Use and regularly update anti-virus software or programs
                - Develop and maintain secure systems and applications
            - Implement Strong Access Control Measures
                - Restrict access to cardholder data by business need to know
                - Assign a unique ID to each person with computer access
                - Restrict physical access to cardholder data
            - Regularly Monitor and Test Networks
                - Track and monitor all access to network resources and cardholder data
                - Regularly test security systems and processes
            - Maintain an Information Security Policy
                - Maintain a policy that addresses information security for all personnel

    - ISO/IEC 27001:2013 https://www.iso.org

        - Specifies the requirements for establishing, implementing, maintaining, and continually improving an information security management system within the context of the organization
        - It is intended to be suitable for several different types of use, including:
            - Use within organizations to formulate security requirements and objectives
            - Use within organizations as a way to ensure that security risks are cost-effectively managed
            - Use within organizations to ensure compliance with laws and regulations
            - Defining new information security management processes
            - Identifying and clarifying existing information security management processes
            - Use by the management of organizations to determine the status of information security management activities
            - Implementing business-enabling information security

        - Use by organizations to provide relevant information about information security to customers

- Health Insurance Portability and Accountability Act (HIPAA) https://www.hhs.gov

    - Electronic Transaction and Code Set Standards
        - Requires every provider who does business electronically to use the same health care transactions, code sets, and identifiers
    - Privacy Rule
        - Provides federal protections for the personal health information held by covered entities and gives patients an array of rights with respect to that information
    - Security Rule
        - Specifies a series of administrative, physical, and technical safeguards for covered entities to use to ensure the confidentiality, integrity, and availability of electronically protected health information
    - National Identifier Requirements
        - Requires that health care providers, health plans, and employers have standard national numbers that identify them attached to standard transactions
    - Enforcement Rule
        - Provides the standards for enforcing all the Administration Simplification Rules

- Sarbanes Oxley Act (SOX) https://www.sec.gov

    - Enacted in 2002, the Sarbanes-Oxley Act is designed to protect investors and the public by increasing the accuracy and reliability of corporate disclosures
    - The key requirements and provisions of SOX are organized into 11 titles:
        - Title I:
            - Public Company Accounting Oversight Board (PCAOB) provides independent oversight of public accounting firms providing audit services (“auditors”)
        - Title II
            - Auditor Independence establishes the standards for external auditor independence, intended to limit conflicts of interest and address new auditor approval requirements, audit partner rotation, and auditor reporting requirements
        - Title III
            - Corporate Responsibility mandates that senior executives take individual responsibility for the accuracy and completeness of corporate financial reports
        - Title IV
            - Enhanced Financial Disclosures describe enhanced reporting requirements for financial transactions, including off-balance-sheet transactions, pro-forma figures, and the stock transactions of corporate officers
        - Title V
            - Analyst Conflicts of Interest consist of measures designed to help restore investor confidence in the reporting of securities analysts
        - Title VI
            - Commission Resources and Authority defines practices to restore investor confidence in securities analysts

        - Title VII
            - Studies and Reports includes the effects of the consolidation of public accounting firms, the role of credit rating agencies in the operation of securities markets, securities violations and enforcement actions, and whether investment banks assisted Enron, Global Crossing, or others to manipulate earnings and obfuscate true financial conditions
        - Title VIII
            - Corporate and Criminal Fraud Accountability describes specific criminal penalties for fraud by the manipulation, destruction, or alteration of financial records, or other interference with investigations while providing certain protections for whistle-blowers
        - Title X
            - White Collar Crime Penalty Enhancement increases the criminal penalties associated with white-collar crimes and conspiracies. It recommends stronger sentencing guidelines and specifically adds the failure to certify corporate financial reports as a criminal offense
        - Title IX
            - Corporate Tax Returns states that the Chief Executive Officer should sign the company tax return
        - Title XI
            - Corporate Fraud Accountability identifies corporate fraud and record tampering as criminal offenses and assigns them specific penalties. It also revises sentencing guidelines and strengthens their penalties. This enables the SEC to temporarily freeze large or unusual payments

- The Digital Millennium Copyright Act (DMCA) https://www.copyright.gov

    - The DMCA is a United States copyright law that implements two 1996 treaties of the World Intellectual Property Organization (WIPO)
    - It defines the legal prohibitions against the circumvention of technological protection measures employed by copyright owners to protect their works, and against the removal or alteration of copyright management information

- The Federal Information Security Management Act (FISMA) https://csrc.nist.gov

    - The FISMA provides a comprehensive framework for ensuring the effectiveness of information security controls over information resources that support Federal operations and assets. It includes:
        - Standards for categorizing information and information systems by mission impact
        - Standards for minimum security requirements for information and information systems
        - Guidance for selecting appropriate security controls for information systems
        - Guidance for assessing security controls in information systems and determining security control effectiveness
        - Guidance for security authorization of information systems

- General Data Protection Regulation (GDPR) https://gdpr.eu

        - GDPR regulation was put into effect on May 25, 2018 and one of the most stringent privacy and security laws globally
        - The GDPR will levy harsh fines against those who violate its privacy and security standards, with penalties reaching tens of millions of euros
        - GDPR Data Protection Principles:
            - Lawfulness, fairness, and transparency
            - Purpose limitation
            - Data minimization
            - Accuracy
            - Storage limitation
            - Integrity and confidentiality
            - Accountability
    - Data Protection Act 2018 (DPA) https://www.legislation.gov.uk
        - The DPA is an act to make provision for the regulation of the processing of information relating to individuals; to make provision in connection with the Information Commissioner's functions under specific regulations relating to information; to make provision for a direct marketing code of practice, and connected purposes
        - The DPA protects individuals concerning the processing of personal data, in particular by:
            - Requiring personal data to be processed lawfully and fairly, based on the data subject's consent or another specified basis,
            - Conferring rights on the data subject to obtain information about the processing of personal data and to require inaccurate personal data to be rectified, and
            - Conferring functions on the Commissioner, giving the holder of that office responsibility to monitor and enforce their provisions
    - Cyber Law in Different Countries
        - United States
            - Section 107 of the Copyright Law mentions the doctrine of "fair use" https://www.copyright.gov
            - Online Copyright Infringement Liability Limitation Act https://www.uspto.gov
            - The Lanham (Trademark) Act (15 USC §§ 1051 - 1127) https://www.uspto.gov
            - The Electronic Communications Privacy Act https://fas.org
            - Foreign Intelligence Surveillance Act https://fas.org
            - Protect America Act of 2007 https://www.justice.gov
            - Privacy Act of 1974 https://www.justice.gov
            - National Information Infrastructure Protection Act of 1996 https://www.nrotc.navy.mil
            - Computer Security Act of 1987 https://csrc.nist.gov
            - Freedom of Information Act (FOIA) https://www.foia.gov
            - Computer Fraud and Abuse Act https://energy.gov
            - Federal Identity Theft and Assumption Deterrence Act https://www.ftc.gov
        - Australia https://www.legislation.gov.au
            - The Trade Marks Act 1995

    - The Patents Act 1990
    - The Copyright Act 1968
    - Cybercrime Act 2001

  - United Kingdom https://www.legislation.gov.uk

    - The Copyright, Etc. and Trademarks (Offenses And Enforcement) Act 2002
    - Trademarks Act 1994 (TMA)
    - Computer Misuse Act 1990
    - The Network and Information Systems Regulations 2018
    - Communications Act 2003
    - The Privacy and Electronic Communications (EC Directive) Regulations 2003
    - Investigatory Powers Act 2016
    - Regulation of Investigatory Powers Act 2000

  - China

    - Copyright Law of the People’s Republic of China (Amendments on October 27, 2001) http://www.npc.gov.cn
    - Trademark Law of the People's Republic of China (Amendments on October 27, 2001) http://www.npc.gov.cn

  - India

    - The Patents (Amendment) Act, 1999, Trade Marks Act, 1999, The Copyright Act, 1957 http://www.ipindia.nic.in
    - Information Technology Act https://www.meity.gov.in

  - Germany

    - Section 202a. Data Espionage, Section 303a. Alteration of Data, Section 303b. Computer Sabotage https://www.cybercrimelaw.net

  - Italy

    - Penal Code Article 615 ter https://www.cybercrimelaw.net

  - Japan

    - The Trademark Law (Law No. 127 of 1957), Copyright Management Business Law (4.2.2.3 of 2000) https://www.iip.or.jp

  - Canada

    - Copyright Act (R.S.C., 1985, c. C-42), Trademark Law, Canadian Criminal Code Section 342.1 https://laws-lois.justice.gc.ca

  - Singapore

    - Computer Misuse Act https://sso.agc.gov.sg

  - South Africa

- Trademarks Act 194 of 1993 http://www.cipc.co.za
- Copyright Act of 1978 https://www.nlsa.ac.za

- South Korea

    - Copyright Law Act No. 3916 https://www.copyright.or.kr
    - Industrial Design Protection Act https://www.kipo.go.kr

- Belgium

    - Copyright Law, 30/06/1994 https://www.wipo.int
    - Computer Hacking https://www.cybercrimelaw.net

- Brazil

    - Unauthorized modification or alteration of the information system https://www.domstol.no

- Hong Kong

    - Article 139 of the Basic Law https://www.basiclaw.gov.hk

Module 02: Ethical Hacking Fundamentals

- Cyber Kill Chain Methodology

    - The cyber kill chain methodology is a component of intelligence-driven defense for the identification and prevention of malicious intrusion activities

    - It helps security professionals to understand the adversary’s tactics, techniques, and procedures beforehand

        - Reconnaissance
            - Gather data on the target to probe for weak points
            - Activities of the adversary include the following:
                - Gathering information about the target organization by searching the Internet or through social engineering
                - Performing analysis of various online activities and publicly available information
                - Gathering information from social networking sites and web services
                - Obtaining information about websites visited
                - Monitoring and analyzing the target organization’s website
                - Performing Whois, DNS, and network footprinting
                - Performing scanning to identify open ports and services
        - Weaponization
            - Create a deliverable malicious payload using an exploit and a backdoor
            - The following are the activities of the adversary:
                - Identifying appropriate malware payload based on the analysis
                - Creating a new malware payload or selecting, reusing, modifying the available malware payloads based on the identified vulnerability

        - Creating a phishing email campaign
        - Leveraging exploit kits and botnets
    - Delivery
      - Send weaponized bundle to the victim using email, USB, etc.
      - The following are the activities of the adversary:
        - Sending phishing emails to employees of the target organization
        - Distributing USB drives containing malicious payload to employees of the target organization
        - Performing attacks such as watering hole on the compromised website
        - Implementing various hacking tools against the operating systems, applications, and servers of the target organization
    - Exploitation
      - Exploit a vulnerability by executing code on the victim’s system
      - Exploiting software or hardware vulnerabilities to gain remote access to the target system
    - Installation
      - Install malware on the target system
      - The following are the activities of the adversary:
        - Downloading and installing malicious software such as backdoors
        - Gaining remote access to the target system
        - Leveraging various methods to keep backdoor hidden and running
        - Maintaining access to the target system
    - Command and Control
      - Create a command and control channel to communicate and pass data back and forth
      - The following are the activities of the adversary:
        - Establishing a two-way communication channel between the victim’s system and the adversary-controlled server
        - Leveraging channels such as web traffic, email communication, and DNS messages
        - Applying privilege escalation techniques
        - Hiding any evidence of compromise using techniques such as encryption
    - Actions on Objectives
      - Perform actions to achieve intended objectives/goals

  - Tactics, Techniques, and Procedures (TTPs)

    - The term Tactics, Techniques, and Procedures (TTPs) refers to the patterns of activities and methods associated with specific threat actors or groups of threat actors

    - Tactics are the guidelines that describe the way an attacker performs the attack from beginning to the end

    - Techniques are the technical methods used by an attacker to achieve intermediate results during the attack

    - Procedures are organizational approaches that threat actors follow to launch an attack

- Adversary Behavioral Identification
    - Adversary behavioral identification involves the identification of the common methods or techniques followed by an adversary to launch attacks on or to penetrate an organization's network
    - It gives the security professionals insight into upcoming threats and exploits
    - Adversary Behaviors
        - Internal Reconnaissance
        - Use of PowerShell
        - Unspecified Proxy Activities
        - Use of Command-Line Interface
        - HTTP User Agent
        - Command and Control Server
        - Use of DNS Tunneling
        - Use of Web Shell
        - Data Staging
- Indicators of Compromise (IoCs)
    - Indicators of Compromise (IoCs) are the clues, artifacts, and pieces of forensic data found on the network or operating system of an organization that indicate a potential intrusion or malicious activity in the organization's infrastructure
    - IoCs act as a good source of information regarding the threats that serve as data points in the intelligence process
    - Security professionals need to perform continuous monitoring of IoCs to effectively and efficiently detect and respond to evolving cyber threats
- Categories of Indicators of Compromise
    - Understanding IoCs helps security professionals to quickly detect the threats against the organization and protect the organization from evolving threats
    - For this purpose, IoCs are divided into four categories:
        - Email Indicators
            - Used to send malicious data to the target organization or individual
            - Examples include the sender's email address, email subject, and attachments or links
        - Network Indicators
            - Useful for command and control, malware delivery, identifying the operating system, and other tasks
            - Examples include URLs, domain names, and IP addresses
        - Host-Based Indicators
            - Found by performing an analysis of the infected system within the organizational network
            - Examples include filenames, file hashes, registry keys, DLLs, and mutex
        - Behavioral Indicators
            - Used to identify specific behavior related to malicious activities
            - Examples include document executing PowerShell script, and remote command execution

  - Listed below are some of the key Indicators of Compromise (IoCs):
    - Unusual outbound network traffic
    - Unusual activity through a privileged user account
    - Illegitimate files and software
    - Geographical anomalies
    - Multiple login failures
    - Increased database read volume
    - Large HTML response size
    - Multiple requests for the same file
    - Mismatched port-application traffic
    - Unusual usage of ports and protocols
    - Suspicious registry or system file changes
    - Unusual DNS requests
    - Malicious emails
    - Unexpected patching of systems
    - Signs of Distributed Denial-of-Service (DDoS) activity
    - Service interruption and the defacement
    - Bundles of data in the wrong places
    - Web traffic with superhuman behavior
    - A drastic increase in bandwidth usage
    - Malicious hardware

- Hacking Concepts and Hacker Classes
  - What is Hacking?
    - Hacking refers to exploiting system vulnerabilities and compromising security controls to gain unauthorized or inappropriate access to a system’s resources
    - It involves modifying system or application features to achieve a goal outside of the creator’s original purpose
    - Hacking can be used to steal and redistribute intellectual property, leading to business loss
  - Who is a Hacker?
    - An intelligent individual with excellent computer skills who can create and explore computer software and hardware
    - For some hackers, hacking is a hobby to see how many computers or networks they can compromise
    - Some hackers’ intentions can either be to gain knowledge or to probe and do illegal things
    - Some hack with malicious intent such as to steal business data, credit card information, social security numbers, email passwords, and other sensitive data
  - Hacker Classes/Threat Actors
    - Black Hats
      - Individuals with extraordinary computing skills; they resort to malicious or destructive activities and are also known as crackers

- White Hats
  - Individuals who use their professed hacking skills for defensive purposes and are also known as security analysts
- Gray Hats
  - Individuals who work both offensively and defensively at various times
- Suicide Hackers
  - Individuals who aim to bring down the critical infrastructure for a "cause" and are not worried about facing jail terms or any other kind of punishment
- Script Kiddies
  - An unskilled hacker who compromises a system by running scripts, tools, and software that were developed by real hackers
- Cyber Terrorists
  - Individuals with a wide range of skills who are motivated by religious or political beliefs to create the fear through the large-scale disruption of computer networks
- State-Sponsored Hackers
  - Individuals employed by the government to penetrate and gain top-secret information from, and damage the information systems of other governments
- Hacktivist
  - Individuals who promote a political agenda by hacking, especially by using hacking to deface or disable website
- Hacker Teams
  - A consortium of skilled hackers having their own resources and funding. They work together in synergy for researching the state-ofthe- art technologies
- Industrial Spies
  - Individuals who perform corporate espionage by illegally spying on competitor organizations and focus on stealing information such as blueprints and formulas
- Insider
  - Any employee (trusted person) who has access to critical assets of an organization. They use privileged access to violate rules or intentionally cause harm to the organization’s information system
- Criminal Syndicates
  - Groups of individuals that are involved in organized, planned, and prolonged criminal activities. They illegally embezzle money by performing sophisticated

cyberattacks

    - Organized Hackers

        - Miscreants or hardened criminals who use rented devices or botnets to perform various cyber-attacks to pilfer money from victims

- Different Phases of Hacking Cycle

    - Hacking Phases

        - Reconnaissance
        - Scanning
        - Gaining Access
        - Maintaining Access
        - Clearing Tracks

    - Hacking Phase: Reconnaissance

        - Reconnaissance refers to the preparatory phase where an attacker seeks to gather information about a target prior to launching an attack
        - Reconnaissance Types
            - Passive Reconnaissance
                - Involves acquiring information without directly interacting with the target
                - For example, searching public records or news releases
            - Active Reconnaissance
                - Involves directly interacting with the target by any means
                - For example, telephone calls to the target's help desk or technical department

    - Hacking Phase: Scanning

        - Scanning refers to the pre-attack phase when the attacker scans the network for specific information based on information gathered during reconnaissance
        - Scanning can include the use of dialers, port scanners, network mappers, ping tools, and vulnerability scanners
        - Attackers extract information such as live machines, port, port status, OS details, device type, and system uptime to launch attack

    - Hacking Phase: Gaining Access

        - Gaining access refers to the point where the attacker obtains access to the operating system or applications on the target computer or network
        - The attacker can gain access at the operating system, application, or network levels
        - The attacker can escalate privileges to obtain complete control of the system
        - Examples include password cracking, buffer overflows, denial of service, and session hijacking

    - Hacking Phase: Maintaining Access

        - Maintaining access refers to the phase when the attacker tries to retain their ownership of the system
        - Attackers may prevent the system from being owned by other attackers by securing their exclusive access with backdoors, rootkits, or Trojans
        - Attackers can upload, download, or manipulate data, applications, and configurations on the owned system
        - Attackers use the compromised system to launch further attacks

    - Hacking Phase: Clearing Tracks

        - Clearing tracks refers to the activities carried out by an attacker to hide malicious acts
        - The attacker's intentions include obtaining continuing access to the victim's system, remaining unnoticed and uncaught, and deleting evidence that might lead to their prosecution
        - The attacker overwrites the server, system, and application logs to avoid suspicion

- Ethical Hacking Concepts, Scope, and Limitations

    - What is Ethical Hacking?
        - Ethical hacking involves the use of hacking tools, tricks, and techniques to identify vulnerabilities and ensure system security
        - It focuses on simulating the techniques used by attackers to verify the existence of exploitable vulnerabilities in a system's security
        - Ethical hackers perform security assessments for an organization with the permission of concerned authorities
        - Consider the following definitions:
            - The noun "hacker" refers to a person who enjoys learning the details of computer systems and stretching their capabilities.
            - The verb "to hack" describes the rapid development of new programs or the reverse engineering of existing software to make it better or more efficient in new and innovative ways.
            - The terms "cracker" and "attacker" refer to persons who employ their hacking skills for offensive purposes.
            - The term "ethical hacker" refers to security professionals who employ their hacking skills for defensive purposes.
    - Why Ethical Hacking is Necessary
        - To beat a hacker, you need to think like one!
        - Ethical hacking is necessary as it allows for counter attacks against malicious hackers through anticipating the methods used to break into the system
        - Reasons why organizations recruit ethical hackers
            - To prevent hackers from gaining access to the organization's information systems
            - To uncover vulnerabilities in systems and explore their potential as a security risk
            - To analyze and strengthen an organization's security posture
            - To provide adequate preventive measures in order to avoid security breaches
            - To help safeguard customer data
            - To enhance security awareness at all levels in a business
        - Ethical Hackers Try to Answer the Following Questions

        - What can an intruder see on the target system? (Reconnaissance and Scanning phases)
        - What can an intruder do with that information? (Gaining Access and Maintaining Access phases)
        - Does anyone at the target organization notice the intruders’ attempts or successes? (Reconnaissance and Covering Tracks phases)
        - Are all components of the information system adequately protected, updated, and patched?
        - How much time, effort, and money are required to obtain adequate protection?
        - Are the information security measures in compliance with legal and industry standards?
    - Scope and Limitations of Ethical Hacking
        - Scope
            - Ethical hacking is a crucial component of risk assessment, auditing, counter fraud, and information systems security best practices
            - It is used to identify risks and highlight remedial actions. It also reduces ICT costs by resolving vulnerabilities
        - Limitations
            - Unless the businesses already know what they are looking for and why they are hiring an outside vendor to hack systems in the first place, chances are there would not be much to gain from the experience
            - An ethical hacker can only help the organization to better understand its security system; it is up to the organization to place the right safeguards on the network
    - Skills of an Ethical Hacker
        - Technical Skills
            - In-depth knowledge of major operating environments, such as Windows, Unix, Linux, and Macintosh
            - In-depth knowledge of networking concepts, technologies, and related hardware and software
            - A computer expert adept at technical domains
            - The knowledge of security areas and related issues
            - High technical knowledge of how to launch sophisticated attacks
        - Non-Technical Skills
            - The ability to quickly learn and adapt new technologies
            - A strong work ethic and good problem solving and communication skills
            - Commitment to an organization’s security policies
            - An awareness of local standards and laws

- Ethical Hacking Tools

    - Reconnaissance Using Advanced Google Hacking Techniques

        - Google hacking refers to the use of advanced Google search operators for creating complex search queries to extract sensitive or hidden information that helps attackers find vulnerable targets
        - Popular Google advanced search operators http://www.googleguide.com
            - [cache:] Displays the web pages stored in the Google cache

      - [link:] Lists web pages that have links to the specified web page
      - [related:] Lists web pages that are similar to the specified web page
      - [info:] Presents some information that Google has about a particular web page
      - [site:] Restricts the results to those websites in the given domain
      - [allintitle:] Restricts the results to those websites containing all the search keywords in the title
      - [intitle:] Restricts the results to documents containing the search keyword in the title
      - [allinurl:] Restricts the results to those containing all the search keywords in the URL
      - [inurl:] Restricts the results to documents containing the search keyword in the URL
      - [location:] Finds information for a specific location

  - Reconnaissance Tools

    - Web Data Extractor http://www.webextractor.com
      - It extracts targeted contact data (email, phone, and fax) from the website, extracts the URL and meta tags (title, description, keyword) for website promotion, and so on
    - Whois Lookup
      - https://whois.domaintools.com
      - https://www.tamos.com
    - IMCP Traceroute
    - TCP Traceroute
    - UDP Traceroute

  - Scanning Tools

    - Nmap https://nmap.org
      - Use Nmap to extract information such as live hosts on the network, open ports, services (application name and version), types of packet filters/ firewalls, as well as operating systems and versions used
    - MegaPing http://www.magnetosoft.com
      - Includes scanners such as Comprehensive Security Scanner, Port scanner (TCP and UDP ports), IP scanner, NetBIOS scanner, and Share Scanner
    - Unicornscan
      - In Unicornscan, the OS of the target machine can be identified by observing the TTL values in the acquired scan result
    - Hping2/Hping3 http://www.hping.org
    - NetScanTools Pro https://www.netscantools.com
    - SolarWinds Port Scanner https://www.solarwinds.com
    - PRTG Network Monitor https://www.paessler.com
    - OmniPeek Network Protocol Analyzer https://www.liveaction.com

  - Enumeration Tools

    - Nbtstat Utility https://docs.microsoft.com
      - The nbtstat utility in Windows displays NetBIOS over TCP/IP (NetBT) protocol statistics, NetBIOS name tables for both the local and remote computers, and the NetBIOS name cache

        - nbtstat [-a RemoteName] [-A IP Address] [-c] [-n] [-r] [-R] [-RR] [-s] [-S] [Interval]
      - NetBIOS Enumerator http://nbtenum.sourceforge.net
        - NetBIOS Enumerator helps to enumerate details, such as NetBIOS names, Usernames, Domain names, and MAC addresses, for a given range of IP addresses
      - Other NetBIOS Enumeration Tools:
        - Global Network Inventory http://www.magnetosoft.com
        - Advanced IP Scanner https://www.advancedip-scanner.com
        - Hyena https://www.systemtools.com
        - Nsauditor Network Security Auditor https://www.nsauditor.com

Module 03: Information Security Threats and Vulnerability Assessment

- Threat and Threat Sources

  - What is a Threat?
    - A threat is the potential occurrence of an undesirable event that can eventually damage and disrupt the operational and functional activities of an organization
    - Attackers use cyber threats to infiltrate and steal data such as individual’s personal information, financial information, and login credentials
    - Examples of Threats
      - An attacker stealing sensitive data of an organization
      - An attacker causing a server to shut down
      - An attacker tricking an employee into revealing sensitive information
      - An attacker infecting a system with malware
      - An attacker spoofing the identity of an authorized person to gain access
      - An attacker modifying or tampering with the data transferred over a network
      - An attacker remotely altering the data in a database server
      - An attacker performing URL redirection or URL forwarding
      - An attacker performing privilege escalation for unauthorized access
      - An attacker executing denial-of-service (DoS) attacks for making resources unavailable
      - An attacker eavesdropping on a communication channel without authorized access
  - Threat Sources
    - Natural
      - Fires
      - Floods
      - Power failures
    - Unintentional
      - Unskilled administrators
      - Accidents
      - Lazy or untrained employees
    - Intentional
      - Internal
        - Fired employee
        - Disgruntled employee

            - Service providers
            - Contractors
        - External
            - Hackers
            - Criminals
            - Terrorists
            - Foreign intelligence agents
            - Corporate raiders

- Malware and its Types

    - Malware is malicious software that damages or disables computer systems and gives limited or full control of the systems to the malware creator for the purpose of theft or fraud

    - Malware programmers develop and use malware to:

        - Attack browsers and track websites visited
        - Slow down systems and degrade system performance
        - Cause hardware failure, rendering computers inoperable
        - Steal personal information, including contacts
        - Erase valuable information, resulting in substantial data loss
        - Attack additional computer systems directly from a compromised system
        - Spam inboxes with advertising emails

    - Different Ways for Malware to Enter a System

        - Instant Messenger applications
        - Portable hardware media/removable devices
        - Browser and email software bugs
        - Untrusted sites and freeware web applications/ software
        - Downloading files from the Internet
        - Email attachments
        - Installation by other malware
        - Bluetooth and wireless networks

    - Common Techniques Attackers Use to Distribute Malware on the Web. Security Threat Report (https://www.sophos.com)

        - Black hat Search Engine Optimization (SEO)
            - Ranking malware pages highly in search results
        - Social Engineered Click-jacking
            - Tricking users into clicking on innocent-looking webpages
        - Spear-phishing Sites
            - Mimicking legitimate institutions in an attempt to steal login credentials
        - Malvertising
            - Embedding malware in ad-networks that display across hundreds of legitimate, high-traffic sites
        - Compromised Legitimate Websites
            - Hosting embedded malware that spreads to unsuspecting visitors

    - Drive-by Downloads
      - Exploiting flaws in browser software to install malware just by visiting a web page
    - Spam Emails
      - Attaching the malware to emails and tricking victims to click the attachment

  - Components of Malware

    - The components of a malware software depend on the requirements of the malware author who designs it for a specific target to perform intended tasks
      - Crypter: Software that protects malware from undergoing reverse engineering or analysis
      - Downloader: A type of Trojan that downloads other malware from the Internet on to the PC
      - Dropper: A type of Trojan that covertly installs other malware files on to the system
      - Exploit: A malicious code that breaches the system security via software vulnerabilities to access information or install malware
      - Injector: A program that injects its code into other vulnerable running processes and changes how they execute to hide or prevent its removal
      - Obfuscator: A program that conceals its code and intended purpose via various techniques, and thus, makes it hard for security mechanisms to detect or remove it
      - Packer: A program that allows all files to bundle together into a single executable file via compression to bypass security software detection
      - Payload: A piece of software that allows control over a computer system after it has been exploited
      - Malicious Code: A command that defines malware’s basic functionalities such as stealing data and creating backdoors

  - Types of Malware

    - Trojans
    - Viruses
    - Ransomware
    - Computer Worms
    - Rootkits
    - PUAs or Grayware
    - Spyware
    - Keylogger
    - Botnets
    - Fileless Malware

  - What is a Trojan?

    - It is a program in which the malicious or harmful code is contained inside an apparently harmless program or data, which can later gain control and cause damage
    - Trojans get activated when a user performs certain predefined actions
    - Trojans create a covert communication channel between the victim computer and the attacker for transferring sensitive data

  - Indications of Trojan Attack

    - The computer screen blinks, flips upside-down, or is inverted so that everything is displayed backward
    - The default background or wallpaper settings change automatically
    - Web pages suddenly open without input from the user
    - The color settings of the operating system (OS) change automatically
    - Antivirus programs are automatically disabled
    - Pop-ups with bizarre messages suddenly appear

  - How Hackers Use Trojans

    - Delete or replace critical operating system files
    - Record screenshots, audio, and video of victim's PC
    - Use victim's PC for spamming and blasting email messages
    - Download spyware, adware, and malicious files
    - Disable firewalls and antivirus
    - Create backdoors to gain remote access
    - Steal personal information such as passwords, security codes, and credit card information
    - Encrypt the data and lock out the victim from accessing the machine

  - Common Ports used by Trojans

  - Types of Trojans

    - Remote Access Trojans
    - Backdoor Trojans
    - Botnet Trojans
    - Rootkit Trojans
    - E-Banking Trojans
    - Point-of-Sale Trojans
    - Defacement Trojans
    - Service Protocol Trojans
    - Mobile Trojans
    - IoT Trojans
    - Security Software Disabler Trojans
    - Destructive Trojans
    - DDoS Attack Trojans
    - Command Shell Trojans

  - Creating a Trojan

    - Trojan Horse construction kits help attackers to construct Trojan horses of their choice
    - The tools in these kits can be dangerous and can backfire if not properly executed
    - Trojan Horse Construction Kits
      - DarkHorse Trojan Virus Maker
      - Trojan Horse Construction Kit
      - Senna Spy Trojan Generator
      - Batch Trojan Generator
      - Umbra Loader - Botnet Trojan Maker
      - Theef RAT Trojan

        - Theef is a Remote Access Trojan written in Delphi. It allows remote attackers access to the system via port 9871

  - What is a Virus?

    - A virus is a self-replicating program that produces its own copy by attaching itself to another program, computer boot sector or document
    - Viruses are generally transmitted through file downloads, infected disk/flash drives, and as email attachments
    - Characteristics of Viruses
      - Infect other programs
      - Transform themselves
      - Encrypt themselves
      - Alter data
      - Corrupt files and programs
      - Self-replicate

  - Purpose of Creating Viruses

    - Inflict damage on competitors
    - Realize financial benefits
    - Vandalize intellectual property
    - Play pranks
    - Conduct research
    - Engage in cyber-terrorism
    - Distribute political messages
    - Damage networks or computers
    - Gain remote access to a victim's computer

  - Indications of Virus Attack

    - Processes require more resources and time, resulting in degraded performance
    - Computer beeps with no display
    - Drive label changes and OS does not load
    - Constant antivirus alerts
    - Computer freezes frequently or encounters an error such as BSOD
    - Files and folders are missing
    - Suspicious hard drive activity
    - Browser window “freezes”

  - Stages of Virus Lifecycle

    - Design: Development of virus code using programming languages or construction kits.
    - Replication: The virus replicates for a period within the target system and then spreads itself.
    - Launch: The virus is activated when the user performs specific actions such as running an infected program.
    - Detection: The virus is identified as a threat infecting target system.
    - Incorporation: Antivirus software developers assimilate defenses against the virus.

    - Execution of the damage routine: Users install antivirus updates and eliminate the virus threats.

  - How does a Computer Get Infected by Viruses?

    - When a user accepts files and downloads without properly checking the source
    - Opening infected e-mail attachments
    - Installing pirated software
    - Not updating and not installing new versions of plug-ins
    - Not running the latest antivirus application
    - Clicking malicious online ads
    - Using portable media
    - Connecting to untrusted networks

  - Types of Viruses

    - System or Boot Sector Virus
    - File and Multipartite Virus
    - Macro and Cluster Virus
    - Stealth/Tunneling Virus
    - Encryption Virus
    - Sparse Infector Virus
    - Polymorphic Virus
    - Metamorphic Virus
    - Overwriting File or Cavity Virus
    - Companion/Camouflage Virus
    - Shell and File Extension Virus
    - FAT and Logic Bomb Virus
    - Web Scripting Virus
    - Email and Armored Virus
    - Add-on and Intrusive Virus
    - Direct Action or Transient Virus
    - Terminate & Stay Resident Virus

  - Creating a Virus

    - A virus can be created in two different ways:
      - Writing a Virus Program
      - Using Virus Maker Tools
        - DELmE’s Batch Virus Maker
        - Bhavesh Virus Maker SKW
        - Deadly Virus Maker
        - SonicBat Batch Virus Maker
        - TeraBIT Virus Maker
        - Andreinick05's Batch Virus Maker

  - Ransomware

    - A type of malware that restricts access to the computer system’s files and folders

        - Demands an online ransom payment to the malware creator(s) to remove the restrictions
        - Dharma
            - Dharma is a dreadful ransomware that attacks victims through email campaigns; the ransom notes ask the victims to contact the threat actors via a provided email address and pay in bitcoins for the decryption service
        - eCh0raix
        - SamSam
        - WannaCry
        - Petya and NotPetya
        - GandCrab
        - MegaCortex
        - LockerGoga
        - NamPoHyu
        - Ryuk
        - Cryptgh0st
        - Ransomware Families
            - Cerber
            - CTB-Locker
            - Sodinokibi
            - BitPaymer
            - CryptXXX
            - Cryptorbit ransomware
            - Crypto Locker Ransomware
            - Crypto Defense Ransomware
            - Crypto Wall Ransomware

    - Computer Worms

        - Malicious programs that independently replicate, execute, and spread across the network connections
        - Consume available computing resources without human interaction
        - Attackers use worm payloads to install backdoors in infected computers
        - Monero
        - Bondat
        - Beapy

    - How is a Worm Different from a Virus?

        - A Worm Replicates on its own
            - A worm is a special type of malware that can replicate itself and use memory but cannot attach itself to other programs
        - A Worm Spreads through the Infected Network
            - A worm takes advantage of file or information transport features on computer systems and automatically spreads through the infected network, but a virus does not

    - Worm Makers

    - Internet Worm Maker Thing
      - Internet Worm Maker Thing is an open-source tool used to create worms that can infect victim's drives, files, show messages, and disable antivirus software
    - Some additional worm makers are as follows:
      - Batch Worm Generator
      - C++ Worm Generator

  - Rootkits

    - Rootkits are programs that hide their presence as well as attacker’s malicious activities, granting them full access to the server or host at that time, and in the future

    - Rootkits replace certain operating system calls and utilities with their own modified versions of those routines that, in turn, undermine the security of the target system causing malicious functions to be executed

    - A typical rootkit comprises of backdoor programs, DDoS programs, packet sniffers, log-wiping utilities, IRC bots, etc.

    - The attacker places a rootkit by:

      - Scanning for vulnerable computers and servers on the web
      - Wrapping it in a special package like a game
      - Installing it on public computers or corporate computers through social engineering
      - Launching a zero-day attack (privilege escalation, buffer overflow, Windows kernel exploitation, etc.)

    - Objectives of a rootkit:

      - To root the host system and gain remote backdoor access
      - To mask attacker tracks and presence of malicious applications or processes
      - To gather sensitive data, network traffic, etc. from the system to which attackers might be restricted or possess no access
      - To store other malicious programs on the system and act as a server resource for bot updates

  - Potentially Unwanted Application or Applications (PUAs)

    - Also known as grayware or junkware, are potentially harmful applications that may pose severe risks to the security and privacy of data stored in the system where they are installed

    - Installed when downloading and installing freeware using a third-party installer or when accepting a misleading license agreement

    - Covertly monitor and alter the data or settings in the system, similarly to other malware

    - Types of PUAs

      - Adware
      - Torrent

      - Marketing
      - Cryptomining
      - Dialers

  - Adware

    - A software or a program that supports advertisements and generates unsolicited ads and pop-ups

    - Tracks the cookies and user browsing patterns for marketing purposes and collects user data

    - Consumes additional bandwidth, and exhausts CPU resources and memory

    - Indications of Adware

      - Frequent system lag
      - Inundated advertisements
      - Incessant system crash
      - Disparity in the default browser homepage
      - Presence of new toolbar or browser add-ons
      - Slow Internet

  - Spyware

    - A stealthy program that records the user's interaction with the computer and the Internet without the user's knowledge and sends the information to the remote attackers

    - Hides its process, files, and other objects in order to avoid detection and removal

    - Spyware Propagation

      - Drive-by download
      - Masquerading as anti-spyware
      - Web browser vulnerability exploits
      - Piggybacked software installation
      - Browser add-ons
      - Cookies

    - What Does the Spyware Do?

      - Steals users’ personal information and sends it to a remote server or hijacker
      - Monitors users’ online activity
      - Displays annoying pop-ups
      - Redirects a web browser to advertising sites
      - Changes the browser’s default settings
      - Changes firewall settings

  - Keylogger

    - Keystroke loggers are programs or hardware devices that monitor each keystroke as the user types on a keyboard, logs onto a file, or transmits them to a remote location

        - It allows the attacker to gather confidential information about the victim such as email ID, passwords, banking details, chat room activity, IRC, and instant messages

        - What a Keylogger can Do?

            - Record every keystroke typed on the user's keyboard
            - Capture screenshots at regular intervals, showing user activity such as typed characters or clicked mouse buttons
            - Track the activities of users by logging Window titles, names of launched applications, and other information
            - Monitor the online activity of users by recording addresses of the websites visited and with keywords entered
            - Record all login names, bank and credit card numbers, and passwords, including hidden passwords or data displayed in asterisks or blank spaces
            - Record online chat conversations
            - Make unauthorized copies of both outgoing and incoming email messages

    - Botnets

        - A Botnet is a collection of compromised computers connected to the Internet to perform a distributed task

        - Attackers distribute malicious software that turns a user's computer into a bot

        - Bot refers to a program or an infected system that performs repetitive work or acts as an agent or as a user interface to control other programs

        - Why Attackers use Botnets?

            - Perform DDoS attacks, which consume the bandwidth of the victim's computers
            - Use sniffer to steal information from one botnet and use it against another botnet
            - Perform keylogging to harvest account login information for services
            - Use to spread new bots
            - Perpetrate a "click fraud" by automating clicks
            - Perform mass identity theft

    - Fileless Malware

        - Fileless malware, also known as non-malware, infects legitimate software, applications, and other protocols existing in the system to perform various malicious activities

        - Leverages any existing vulnerabilities to infect the system

        - Resides in the system's RAM

        - Injects malicious code into the running processes such as Microsoft Word, Flash, Adobe PDF Reader, Javascript, and PowerShell

        - Reasons for Using Fileless Malware in Cyber Attacks

            - Stealthy in nature
                - Exploits legitimate system tools

        - Living-off-the-land
            - Exploits default system tools
        - Trustworthy
            - Uses tools that are frequently used and trusted

    - Fileless Propagation Techniques

        - Phishing emails
        - Legitimate applications
        - Native applications
        - Infection through lateral movement
        - Malicious websites
        - Registry manipulation
        - Memory code injection
        - Script-based injection

- Malware Countermeasures

    - Trojan Countermeasures

        - Avoid opening email attachments received from unknown senders
        - Block all unnecessary ports at the host and use a firewall
        - Avoid accepting programs transferred by instant messaging
        - Harden weak default configuration settings and disable unused functionality, including protocols and services
        - Monitor the internal network traffic for odd ports or encrypted traffic
        - Avoid downloading and executing applications from untrusted sources
        - Install patches and security updates for the OS and applications
        - Scan external USB drives and DVDs with antivirus software before using them
        - Restrict permissions within the desktop environment to prevent installation of malicious applications
        - Avoid typing commands blindly and implementing pre-fabricated programs or scripts
        - Manage local workstation file integrity through checksums, auditing, and port scanning
        - Run host-based antivirus, firewall, and intrusion detection software

    - Virus and Worm Countermeasures

        - Install antivirus software that detects and removes infections as they appear
        - Pay attention to the instructions while downloading files or programs from the Internet
        - Regularly update antivirus software
        - Avoid opening attachments received from unknown senders, as viruses spread via email attachments
        - Since virus infections can corrupt data, ensure that you perform regular data backups
        - Schedule regular scans for all drives after the installation of antivirus software

    - Do not accept disks or programs without checking them first using a current version of an antivirus program
    - Do not boot the machine with an infected bootable system disk
    - Stay informed about the latest virus threats
    - Check DVDs for virus infection
    - Ensure that pop-up blockers are turned on and use an Internet firewall
    - Perform disk clean-up and run a registry scanner once a week
    - Run anti-spyware or anti-adware once a week
    - Do not open files with more than one file-type extension
    - Be cautious with files sent through instant messenger applications

- Rootkit Countermeasures

    - Reinstall OS/applications from a trusted source after backing up critical data
    - Maintain well-documented automated installation procedures
    - Perform kernel memory dump analysis to determine the presence of rootkits
    - Harden the workstation or server against the attack
    - Do not download any files/programs from untrusted sources
    - Install network and host-based firewalls and frequently check for updates
    - Ensure the availability of trusted restoration media
    - Update and patch OSs, applications, and firmware
    - Regularly verify the integrity of system files using cryptographically strong digital fingerprint technologies
    - Regularly update antivirus and anti-spyware software
    - Keep anti-malware signatures up to date
    - Avoid logging into an account with administrative privileges
    - Adhere to the least privilege principle
    - Ensure that the chosen antivirus software possesses rootkit protection
    - Do not install unnecessary applications, and disable the features and services not in use
    - Refrain from engaging in dangerous activities on the Internet
    - Close any unused ports
    - Periodically scan the local system using host-based security scanners
    - Increase the security of the system using two-step or multi-step authentication, so that an attacker will not gain root access to the system to install rootkits
    - Never read emails, browse websites, or open documents while handling an active session with a remote server

- Spyware Countermeasures

    - Try to avoid using any computer system that you do not have a complete control over.
    - Never adjust your Internet security setting level too low because it provides many chances for spyware to be installed on your computer. Therefore, always set your Internet browser security settings to either high or medium to protect your computer from spyware.
    - Do not open suspicious emails and file attachments received from unknown senders. There is a high likelihood that you will allow a virus, freeware, or spyware

    onto the computer. Do not open unknown websites linked in spam mail messages, retrieved by search engines, or displayed in pop-up windows because they may mislead you into downloading spyware.
    - Enable a firewall to enhance the security level of your computer.
    - Regularly update the software and use a firewall with outbound protection.
    - Regularly check Task Manager and MS Configuration Manager reports.
    - Regularly update virus definition files and scan the system for spyware.
    - Install anti-spyware software. Anti-spyware is the first line of defense against spyware. This software prevents spyware from installing on your system. It periodically scans and protects your system from spyware.
    - Keep your OS up to date.
        - Windows users should periodically perform a Windows or Microsoft update.
        - For users of other OSs or software products, refer to the information given by the OS vendors, and take essential steps against any vulnerability identified.
    - Perform web surfing safely and download cautiously.
        - Before downloading any software, ensure that it is from a trusted website. Read the license agreement, security warning, and privacy statements associated with the software thoroughly to gain a clear understanding before downloading it.
        - Before downloading freeware or shareware from a website, ensure that the site is safe. Likewise, be cautious with software programs obtained through P2P fileswapping software. Before installing such programs, perform a scan using antispyware software.
    - Do not use administrative mode unless it is necessary, because it may execute malicious programs such as spyware in administrator mode. Consequently, attackers may take complete control of your system.
    - Do not download free music files, screensavers, or emoticons from the Internet because when you do, there is a possibility that are downloading spyware along with them.
    - Beware of pop-up windows or web pages. Never click anywhere on the windows that display messages such as “your computer may be infected,” or claim that they can help your computer to run faster. If you click on such windows, your system may become infected with spyware.
    - Carefully read all disclosures, including the license agreement and privacy statement, before installing any application.
    - Do not store personal or financial information on any computer system that is not totally under your control, such as in an Internet café.

- PUAs/Adware Countermeasures

    - Always use whitelisted, trusted, and authorized software vendors and websites for downloading software.
    - Always read the end-user license agreement (EULA) and any other terms and conditions before installing any program.
    - Always turn on the option to detect PUAs in the OS or antivirus software.
    - Regularly update the OS and antivirus software to detect and patch the latest PUAs.

    - Uncheck unnecessary options while performing software setup to prevent the automatic installation of PUAs.
    - Avoid installing programs through the “express method” or “recommended method” and instead choose custom installation.
    - Be vigilant towards social engineering techniques and phishing attacks to avert the download of PUAs.
    - Install trusted antivirus, anti-adware, or ad-blocker software to detect and block adware and other malicious programs.
    - Use paid software versions and avoid downloading freeware and other shareware programs provided by third-party vendors.
    - Employ a firewall to filter data transmission and to send only authorized and trusted content.
    - Carefully examine URLs and email addresses, and avoid clicking on suspicious links.
    - Take time to research and read online reviews before downloading any software or plug-in.
    - Attempt to search for the software in a search engine, instead of clicking on ads redirecting to software download.

- Keylogger Countermeasures

    - Use pop-up blockers and avoid opening junk emails.
    - Install anti-spyware/antivirus programs and keep the signatures up to date.
    - Install professional firewall software and anti-keylogging software.
    - Recognize phishing emails and delete them.
    - Regularly update and patch system software.
    - Do not click on links in unsolicited or dubious emails that may direct you to malicious sites.
    - Use keystroke interference software that insert randomized characters into every keystroke.
    - Antivirus and anti-spyware software can detect any installed software, but it is better to detect these programs before installation. Scan the files thoroughly before installing them onto the computer and use a registry editor or process explorer to check for keystroke loggers.
    - Use the Windows on-screen keyboard accessibility utility to enter a password or any other confidential information. Use your mouse to enter any information such as passwords and credit card numbers into the fields, by using your mouse instead of typing the passwords with the keyboard. This will ensure that your information is confidential.
    - Use an automatic form-filling password manager or a virtual keyboard to enter usernames and passwords, as this will avoid exposure through keyloggers. This automatic form-filling password manager will remove the need to type your personal, financial, or confidential details such as credit card numbers and passwords via the keyboard.
    - Keep your hardware systems secure in a locked environment and frequently check the keyboard cables for attached connectors, USB port, and computer games such as the PS2 that may have been used to install keylogger software.
    - Use software that frequently scan and monitor changes in your system or network.

    - Install a host-based IDS, which can monitor your system and disable the installation of keyloggers.
    - Use one-time password (OTP) or other authentication mechanisms such as two-step or multi-step verification to authenticate users.
    - Enable application whitelisting to block downloading or installing of unwanted software such as keyloggers.
    - Use VPN to enable an additional layer of protection through encryption.
    - Use process-monitoring tools to detect suspicious processes and system activities.
    - Regularly patch and update software and the OS.

- Fileless Malware Countermeasures

    - Remove all the administrative tools and restrict access through Windows Group Policy or Windows AppLocker
    - Disable PowerShell and WMI when not in use
    - Disable macros and use only digitally signed trusted macros
    - Install whitelisting solutions such as McAfee Application Control to block unauthorized applications and code running on your systems
    - Never enable macros in MS Office documents
    - Disable PDF readers to run JavaScript automatically
    - Disable Flash in the browser settings
    - Implement two-factor authentication to access critical systems or resources connected to the network
    - Implement multi-layer security to detect and defend against memory-resident malware
    - Use User Behavior Analytics (UBA) solutions to detect threats hidden within your data
    - Ensure the ability to detect system tools such as PowerShell and WMIC, and whitelisted application scripts against malicious attacks
    - Run periodic antivirus scans to detect infections and keep the antivirus program updated
    - Install browser protection tools and disable automatic plugin downloads
    - Schedule regular security checks for applications and regularly patch the applications
    - Regularly update the OS with the latest security patches
    - Examine all the running programs for any malicious or new signatures and heuristics
    - Enable endpoint security with active monitoring to protect networks when accessed remotely
    - Examine the indicators of compromise on the system and the network
    - Regularly check the security logs especially when excessive amounts of data leave the network
    - Restrict admin rights and provide the least privileges to the user level to prevent privilege escalation attacks
    - Use application control to prevent Internet browsers from spawning script interpreters such as PowerShell and WMIC.
    - Carefully examine the changes in the system’s usual behavior patterns compared with the baselines

        - Use next-generation antivirus (NGAV) software that employs advanced technology such as ML (machine learning) and AI (artificial intelligence) to avoid new polymorphic malware
        - Use baseline and search for known tactics, techniques, and procedures (TTPs) used by many adversarial groups
        - Ensure that you use Managed Detection and Response (MDR) services that can perform threat hunting
        - Ensure that you use tools such as Blackberry Cylance and Microsoft Enhanced Mitigation Experience Toolkit to combat fileless attacks
        - Disable unused or unnecessary applications and service features
        - Uninstall applications that are not important
        - Block all the incoming network traffic or files with the .exe format

- Vulnerabilities

    - Refers to the existence of weakness in an asset that can be exploited by threat agents

    - Common Reasons behind the Existence of Vulnerability

        - Hardware or software misconfiguration
        - Insecure or poor design of the network and application
        - Inherent technology weaknesses
        - Careless approach of end users

    - Vulnerability Classification

        - Misconfiguration
            - An application running with debug enabled
            - Unnecessary administrative ports that are open for an application
            - Running outdated software on the system
            - Running unnecessary services on a machine
            - Outbound connections to various Internet services
            - Using misconfigured SSL certificates or default certificates
            - Improperly authenticated external systems
            - Incorrect folder permissions
            - Default accounts or passwords
            - Set up or configuration pages enabled
            - Disabling security settings and features
        - Default Installations
        - Buffer Overflows
        - Unpatched Servers
        - Design Flaws
        - Operating System Flaws
        - Application Flaws
        - Open Services
        - Default Passwords
        - Zero-day/Legacy Platform vulnerabilities

    - Examples of Network Security Vulnerabilities

        - TCP/IP protocol vulnerabilities
            - HTTP, FTP, ICMP, SNMP, SMTP are inherently insecure
        - Operating System vulnerabilities
            - An OS can be vulnerable because:
                - It is inherently insecure
                - It is not patched with the latest updates
        - Network Device Vulnerabilities
            - Various network devices such as routers, firewall, and switches can be vulnerable due to:
            - Lack of password protection
            - Lack of authentication
            - Insecure routing protocols
            - Firewall vulnerabilities
            - User account vulnerabilities
                - Originating from the insecure transmission of user account details such as usernames and passwords, over the network
            - System account vulnerabilities
                - Originating from setting of weak passwords for system accounts
            - Internet service misconfiguration
                - Misconfiguring internet services can pose serious security risks. For example, enabling JavaScript and misconfiguring IIS, Apache, FTP, and Terminal services, can create security vulnerabilities in the network
            - Default password and settings
                - Leaving the network devices/products with their default passwords and settings
            - Network device misconfiguration
                - Misconfiguring the network device
            - Unwritten Policy
                - Unwritten security policies are difficult to implement and enforce
            - Lack of Continuity
                - Lack of continuity in implementing and enforcing the security policy
            - Politics
                - Politics may cause challenges for implementation of a consistent security policy
            - Lack of awareness
                - Lack of awareness of the security policy

    - Impact of Vulnerabilities

        - Information disclosure: A website or application may expose system-specific information.
        - Denial of service: Vulnerabilities may prevent users from accessing website services or other resources.
        - Privilege escalation: Attackers may gain elevated access to a protected system or resources.
        - Unauthorized access: Attackers may gain unauthorized access to a system, a network, data, or an application.

    - Identity theft: Attackers may be able to steal the personal or financial information of users to commit fraud with their identity.
    - Data exfiltration: Vulnerabilities may lead to the unauthorized retrieval and transmission of sensitive data.
    - Reputational damage: Vulnerabilities may cause reputational damage to a company's products and security. Reputational damage has a direct impact on customers, sales, and profit.
    - Financial loss: Reputational damage may lead to business loss. Further, vulnerability exploitation may lead to expenses for recovering damaged IT infrastructure.
    - Legal consequences: If customers' personal data are compromised, the organization may need to face legal consequences in the form of fines and regulatory sanctions.
    - Hold footprints: Vulnerabilities may allow attackers to stay undetected even after executing an attack.
    - Remote code execution: Vulnerabilities may allow the execution of arbitrary code from remote servers.
    - Malware installation: Vulnerabilities can make it easy to infect with and spread viruses in a network.
    - Data modification: Vulnerabilities may allow attackers to intercept and alter data in transit.

  - Vulnerability Research

    - The process of analyzing protocols, services, and configurations to discover vulnerabilities and design flaws that will expose an operating system and its applications to exploit, attack, or misuse

    - Vulnerabilities are classified based on severity level (low, medium, or high) and exploit range (local or remote)

    - An administrator needs vulnerability research:

      - To gather information about security trends, newly discovered threats, attack surfaces, attack vectors and techniques
      - To find weaknesses in the OS and applications and alert the network administrator before a network attack
      - To understand information that helps prevent security problems
      - To know how to recover from a network attack

  - Resources for Vulnerability Research

    - Microsoft Vulnerability Research (MSVR) (https://www.microsoft.com)
    - Dark Reading (https://www.darkreading.com)
    - SecurityTracker (https://securitytracker.com)
    - Trend Micro (https://www.trendmicro.com)
    - Security Magazine (https://www.securitymagazine.com)
    - PenTest Magazine (https://pentestmag.com)
    - SC Magazine (https://www.scmagazine.com)
    - Exploit Database (https://www.exploit-db.com)
    - SecurityFocus (https://www.securityfocus.com)
    - Help Net Security (https://www.helpnetsecurity.com)

    - HackerStorm (http://www.hackerstorm.co.uk)
    - Computerworld (https://www.computerworld.com)
    - WindowsSecurity (http://www.windowsecurity.com)
    - D'Crypt (https://www.d-crypt.com)

- What is Vulnerability Assessment?

    - Vulnerability assessment is an in-depth examination of the ability of a system or application, including current security procedures and controls, to withstand the exploitation
    - It recognizes, measures, and classifies security vulnerabilities in a computer system, network, and communication channels
    - A vulnerability assessment may be used to:
        - Identify weaknesses that could be exploited
        - Predict the effectiveness of additional security measures in protecting information resources from attack
    - Limitations of Vulnerability Assessment
        - Vulnerability-scanning software is limited in its ability to detect vulnerabilities at a given point in time
        - Vulnerability-scanning software must be updated when new vulnerabilities are discovered or when improvements are made to the software being used
        - Software is only as effective as the maintenance performed on it by the software vendor and by the administrator who uses it
        - Vulnerability Assessment does not measure the strength of security controls
        - Vulnerability-scanning software itself is not immune to software engineering flaws that might lead to it missing serious vulnerabilities
        - Human judgment is needed to analyze the data after scanning and identifying the false positives and false negatives.

- Information Obtained from the Vulnerability Scanning

    - The OS version running on computers or devices
    - IP and Transmission Control Protocol/User Datagram Protocol (TCP/UDP) ports that are listening
    - Applications installed on computers
    - Accounts with weak passwords
    - Files and folders with weak permissions
    - Default services and applications that might have to be uninstalled
    - Errors in the security configuration of common applications
    - Computers exposed to known or publicly reported vulnerabilities
    - EOL/EOS software information
    - Missing patches and hotfixes
    - Weak network configurations and misconfigured or risky ports
    - Help to verify the inventory of all devices on the network

- Vulnerability Scanning Approaches

    - Active Scanning

      - The attacker interacts directly with the target network to find vulnerabilities
      - Example: An attacker sends probes and specially crafted requests to the target host in the network to identify vulnerabilities
    - Passive Scanning
      - The attacker tries to find vulnerabilities without directly interacting with the target network
      - Example: An attacker guesses the operating system information, applications, and application and service versions by observing the TCP connection setup and teardown

  - Vulnerability Scoring Systems and Databases

    - Common Vulnerability Scoring System (CVSS) https://www.first.org https://nvd.nist.gov
      - An open framework for communicating the characteristics and impacts of IT vulnerabilities
      - Its quantitative model ensures repeatable accurate measurement, while enabling users to view the underlying vulnerability characteristics used to generate the scores
    - Common Vulnerabilities and Exposures (CVE) https://cve.mitre.org
      - A publicly available and free-to-use list or dictionary of standardized identifiers for common software vulnerabilities and exposures
    - National Vulnerability Database (NVD) https://nvd.nist.gov
      - A U.S. government repository of standards based vulnerability management data represented using the Security Content Automation Protocol (SCAP)
      - These data enable the automation of vulnerability management, security measurement, and compliance
      - The NVD includes databases of security checklist references, security-related software flaws, misconfigurations, product names, and impact metrics
    - Common Weakness Enumeration (CWE) https://cwe.mitre.org
      - A category system for software vulnerabilities and weaknesses
      - It is sponsored by the National Cybersecurity FFRDC, which is owned by The MITRE Corporation, with support from US-CERT and the National Cyber Security Division of the U.S. Department of Homeland Security
      - It has over 600 categories of weaknesses, which enable CWE to be effectively employed by the community as a baseline for weakness identification, mitigation, and prevention efforts

- Vulnerability Assessment

  - Types of Vulnerability Assessment

    - Active Assessment
      - Uses a network scanner to find hosts, services, and vulnerabilities
    - External Assessment
      - Assesses the network from a hacker's perspective to discover exploits and vulnerabilities that are accessible to the outside world
    - Host-based Assessment
      - Conducts a configuration-level check to identify system configurations, user directories, file systems, registry settings, etc., to evaluate the possibility of

compromise

  - Application Assessment
    - Tests and analyzes all elements of the web infrastructure for any misconfiguration, outdated content, or known vulnerabilities
  - Passive Assessment
    - Used to sniff the network traffic to discover present active systems, network services, applications, and vulnerabilities present
  - Internal Assessment
    - Scans the internal infrastructure to discover exploits and vulnerabilities
  - Networkbased Assessment
    - Determines possible network security attacks that may occur on the organization’s system
  - Database Assessment
    - Focuses on testing databases, such as MYSQL, MSSQL, ORACLE, POSTGRESQL, etc., for the presence of data exposure or injection type vulnerabilities
  - Wireless Network Assessment
    - Determines the vulnerabilities in the organization’s wireless networks
  - Credentialed Assessment
    - Assesses the network by obtaining the credentials of all machines present in the network
  - Manual Assessment
    - In this type of assessment, the ethical hacker manually assesses the vulnerabilities, vulnerability ranking, vulnerability score, etc.
  - Distributed Assessment
    - Assesses the distributed organization assets, such as client and server applications, simultaneously through appropriate synchronization techniques
  - Non-Credentialed Assessment
    - Assesses the network without acquiring any credentials of the assets present in the enterprise network
  - Automated Assessment
    - In this type of assessment, the ethical hacker employs various vulnerability assessment tools, such as Nessus, Qualys, GFI LanGuard, etc.

- Vulnerability-Management Life Cycle

  - Identify Assets and Create a Baseline
  - Vulnerability Scan
  - Risk Assessment
  - Remediation
  - Verification
  - Monitor

- Vulnerability Assessment Tools

  - Qualys Vulnerability Management https://www.qualys.com
    - A cloud-based service that offers immediate global visibility into IT system areas that might be vulnerable to the latest Internet threats and how to protect them

        - Aids in the continuous identification of threats and monitoring of unexpected changes in a network before they become breaches
      - OpenVAS https://www.openvas.org
        - A framework of several services and tools offering a comprehensive and powerful vulnerability scanning and vulnerability management solution
      - GFI LanGuard https://www.gfi.com
        - Scans, detects, assesses, and rectifies security vulnerabilities in a network and connected devices
      - Other Vulnerability Assessment Tools
        - Nessus Professional https://www.tenable.com
        - Nikto https://cirt.net
        - Qualys FreeScan https://freescan.qualys.com
        - Acunetix Web Vulnerability Scanner https://www.acunetix.com
        - Nexpose https://www.rapid7.com
        - Network Security Scanner https://www.beyondtrust.com
        - SAINT Security Suite https://www.carson-saint.com
        - beSECURE (AVDS) https://www.beyondsecurity.com
        - Core Impact https://www.coresecurity.com
        - N-Stalker Web Application Security Scanner https://www.nstalker.com

  - Vulnerability Exploitation

    - Vulnerability exploitation involves the execution of multiple complex, interrelated steps to gain access to a remote system
    - The steps involved are as follows:
      - Identify the vulnerability https://www.exploit-db.com https://www.securityfocus.com
      - Determine the risk associated with the vulnerability
      - Determine the capability of the vulnerability
      - Develop the exploit
      - Select the method for delivering: local or remote
      - Generate and deliver the payload
      - Gain remote access

Module 04: Password Cracking Techniques and Countermeasures

- Password Cracking Techniques

  - Password cracking techniques are used to recover passwords from computer systems

  - Attackers use password cracking techniques to gain unauthorized access to vulnerable systems

  - Most of the password cracking techniques are successful because of weak or easily guessable passwords

  - Password Complexity

    - Passwords that contain letters, special characters, and numbers: ap1@52

        - Passwords that contain only numbers: 23698217
        - Passwords that contain only special characters: &*#@!(%)
        - Passwords that contain letters and numbers: meet123
        - Passwords that contain only letters: POTHMYDE
        - Passwords that contain only letters and special characters: bob@&ba
        - Passwords that contain only special characters and numbers: 123@$45
        - Passwords that contain only uppercase and lowercase letters, such as: RuNnEr
        - Passwords that contain more than 20 characters comprising a phrase: such as Hardtocrackveryeasily
        - Passwords that contain shortcut codes or acronyms, such as L8r_L8rNot2day (i.e., later, later, not today)
        - Passwords that contain frequently used words specifying websites, such as ABT2_uz_AMZ! (i.e., about to use Amazon!)
        - Passwords that contain the first letters of words of a long sentence, such as TffcievwMi16wiwdm5g (i.e., the first foreign country I ever visited was Mexico in 2016 when I was doing my 5th grade)

    - Microsoft Authentication

        - Security Accounts Manager (SAM) Database
            - Windows stores user passwords in SAM, or in the Active Directory database in domains
            - Passwords are never stored in clear text and are hashed, and the results are stored in the SAM
        - NTLM Authentication
            - The NTLM authentication protocol types are as follows: NTLM authentication protocol and LM authentication protocol
            - These protocols store the user’s password in the SAM database using different hashing methods
        - Kerberos Authentication
            - Microsoft has upgraded its default authentication protocol to Kerberos which provides a stronger authentication for client/server applications than NTLM

    - Types of Password Attacks

        - Non-Electronic Attacks
            - The attacker does not need technical knowledge to crack the password, hence it is known as a non-technical attack
                - Shoulder Surfing
                - Social Engineering
                - Dumpster Diving
        - Active Online Attacks
            - The attacker performs password cracking by directly communicating with the victim’s machine
                - Dictionary, Brute Forcing, and Rule-based Attack
                - Hash Injection Attack
                - LLMNR/NBT-NS Poisoning
                - Trojan/Spyware/Keyloggers

            - Password Guessing
    - Passive Online Attacks
        - The attacker performs password cracking without communicating with the authorizing party
            - Wire Sniffing
            - Man-in-the-Middle Attack
            - Replay Attack
    - Offline Attacks
        - The attacker copies the target's password file and then tries to crack passwords on his own system at a different location
            - Rainbow Table Attack (Pre-Computed Hashes)
            - Distributed Network Attack

- Dictionary, Brute-Force, and Rule-based Attack

    - Dictionary Attack
        - A dictionary file is loaded into the cracking application that runs against user accounts
    - Brute-Force Attack
        - The program tries every combination of characters until the password is broken
    - Rule-based Attack
        - This attack is used when the attacker gets some information about the password

- Password Guessing

    - The attacker creates a list of all possible passwords from the information collected through social engineering or any other way and manually inputs them on the victim's machine to crack the passwords
    - The following are the steps involved in password guessing:
        - Find a valid user
        - Create a list of possible passwords
        - Rank passwords from high to low probability
        - Key in each password, until the correct password is discovered

- Default Passwords

    - A default password is a password supplied by the manufacturer with new equipment (e.g., switches, hubs, routers) that is password protected
    - Attackers use default passwords present in the list of words or dictionary to perform password guessing attack
    - Online Tools to Search Default Passwords
        - https://open-sez.me
        - https://www.fortypoundhead.com
        - https://cirt.net
        - http://www.defaultpassword.us
        - https://www.routerpasswords.com
        - https://default-password.info

- Trojans/Spyware/Keyloggers

    - The attacker installs a Trojan/Spyware/Keylogger on the victim's machine to collect the victim's usernames and passwords
    - The Trojan/Spyware/Keylogger runs in the background and sends back all user credentials to the attacker

  - Hash Injection/Pass-the-Hash (PtH) Attack

    - A hash injection/PtH attack allows an attacker to inject a compromised hash into a local session and use the hash to validate network resources
    - The attacker finds and extracts a logged-on domain admin account hash
    - The attacker uses the extracted hash to log on to the domain controller

  - LLMNR/NBT-NS Poisoning

    - LLMNR and NBT-NS are the two main elements of Windows operating systems that are used to perform name resolution for hosts present on the same link
    - The attacker cracks the NTLMv2 hash obtained from the victim’s authentication process
    - The extracted credentials are used to log on to the host system in the network

  - Pass the Ticket Attack

    - Pass the Ticket is a technique used for authenticating a user to a system that is using Kerberos without providing the user’s password
    - To perform this attack, the attacker dumps Kerberos tickets of legitimate accounts using credential dumping tools
    - The attacker then launches a pass the ticket attack either by stealing the ST/TGT from an end-user machine, or by stealing the ST/TGT from a compromised Authorization Server
    - The attacker uses the retrieved ticket to gain unauthorized access to the target network services
    - Tools such as Mimikatz, Rubeus, and Windows Credentials Editor are used by attackers to launch such attacks

  - Wire Sniffing

    - Attackers run packet sniffer tools on the local area network (LAN) to access and record the raw network traffic
    - The captured data may include sensitive information such as passwords (FTP, rlogin sessions, etc.) and emails
    - Sniffed credentials are used to gain unauthorized access to the target system

  - Man-in-the-Middle and Replay Attacks

    - In an MITM attack, the attacker acquires access to the communication channels between the victim and the server to extract the information needed
    - In a replay attack, packets and authentication tokens are captured using a sniffer. After the relevant information is extracted, the tokens are placed back on the network to gain access

  - Rainbow Table Attack

    - Rainbow Table

            - A precomputed table that contains word lists like dictionary files, brute force lists, and their hash values
        - Compare the Hashes
            - The hash of passwords is captured and compared with the precomputed hash table. If a match is found, then the password gets cracked
        - Easy to Recover
            - It is easy to recover passwords by comparing the captured password hashes to the precomputed tables

- Password Cracking Tools

    - L0phtCrack https://www.l0phtcrack.com
        - A tool designed to audit passwords and recover applications
    - ophcrack https://ophcrack.sourceforge.io
        - A Windows password cracker based on rainbow tables. It comes with a Graphical User Interface and runs on multiple platforms
    - RainbowCrack http://project-rainbowcrack.com
        - RainbowCrack cracks hashes with rainbow tables. It uses a time-memory tradeoff algorithm to crack hashes
    - John the Ripper https://www.openwall.com
    - hashcat https://hashcat.net
    - THC-Hydra https://github.com
    - Medusa http://foofus.net

- Password Cracking Countermeasures

    - Disallow use of the same password during a password change
    - Disallow password sharing
    - Disallow the use of passwords that can be found in a dictionary
    - Do not use cleartext protocols and protocols with weak encryption
    - Set the password change policy to 30 days
    - Do not use any system default passwords
    - Make passwords hard to guess by requiring 8-12 alphanumeric characters consisting of a combination of uppercase and lowercase letters, numbers, and symbols
    - Ensure that applications neither store passwords in memory nor write them to disks in clear text
    - Use a random string (salt) as a prefix or suffix to the password before encryption
    - Disallow the use of passwords such as date of birth, spouse, child’s, or pet’s name
    - Lockout an account subjected to too many incorrect password guesses
    - Use two-factor or multi-factor authentication, for example, using CAPTCHA to prevent automated attacks

Module 05: Social Engineering Techniques and Countermeasures

- Social Engineering Concepts and its Phases

    - Social engineering is the art of convincing people to reveal confidential information

- Social engineers depend on the fact that people are unaware of the valuable information to which they have access and are careless about protecting it
- Common Targets of Social Engineering
  - Receptionists and Help-Desk Personnel
  - Technical Support Executives
  - System Administrators
  - Users and Clients
  - Vendors of the Target Organization
  - Senior Executives
- Impact of Social Engineering Attack on an Organization
  - Economic Losses
  - Damage to Goodwill
  - Loss of Privacy
  - Dangers of Terrorism
  - Lawsuits and Arbitration
  - Temporary or Permanent Closure
- Behaviors Vulnerable to Attacks
  - Authority
  - Intimidation
  - Consensus or Social Proof
  - Scarcity
  - Urgency
  - Familiarity or Liking
  - Trust
  - Greed
- Factors that Make Companies Vulnerable to Attacks
  - Insufficient Security Training
  - Unregulated Access to Information
  - Several Organizational Units
  - Lack of Security Policies
- Why is Social Engineering Effective?
  - Social engineering does not deal with network security issues; instead, it deals with the psychological manipulation of a human being to extract desired information
    - Security policies are as strong as their weakest link, and human behavior is the most susceptible factor
    - It is difficult to detect social engineering attempts
    - There is no method that can be applied to ensure complete security from social engineering attacks
    - There is no specific software or hardware to defend against a social engineering attack

  - Phases of a Social Engineering Attack
    - Research the Target Company
      - Dumpster diving, websites, employees, tour of the company, etc.
    - Select a Target
      - Identify frustrated employees of the target company
    - Develop a Relationship
      - Develop a relationship with the selected employees
    - Exploit the Relationship
      - Collect sensitive account and financial information, as well as current technologies

- Social Engineering Techniques
  - Types of Social Engineering
    - Human-based Social Engineering
      - “Sensitive information is gathered by interaction”.
      - Techniques:
        - Impersonation
        - Vishing
        - Eavesdropping
        - Shoulder Surfing
        - Dumpster Diving
        - Reverse Social Engineering
        - Piggybacking
        - Tailgating
    - Computer-based Social Engineering
      - “Sensitive information is gathered with the help of computers”.
      - Techniques:
        - Phishing
        - Pop-up Window Attacks
        - Spam Mail
        - Instant Chat Messenger
        - Scareware
    - Mobile-based Social Engineering
      - “Sensitive information is gathered with the help of mobile apps”.
      - Techniques:
        - Publishing Malicious Apps
        - Using Fake Security Apps
        - Repackaging Legitimate Apps
        - SMiShing (SMS Phishing)
  - Human-based Social Engineering: Impersonation
    - The attacker pretends to be someone legitimate or an authorized person
    - Attackers may impersonate a legitimate or authorized person either personally or using a communication medium such as phone, email, etc. to reveal sensitive information

- Impersonation Examples

    - Posing as a Legitimate End User
        - The attacker gives this identity and asks for the sensitive information
        - “Hi! This is John from the Finance Department. I have forgotten my password. Can I get it?”
    - Posing as an Important User
        - The attacker poses as a VIP of a target company, valuable customer, etc.
        - “Hi! This is Kevin, CFO Secretary. I’m working on an urgent project and lost my system’s password. Can you help me out?”

- Impersonation (Vishing)

    - An impersonation technique in which the attacker tricks individuals to reveal personal and financial information using voice technology such as the telephone system, VoIP, etc.
    - Vishing Example
        - Abusing the Over-Helpfulness of Help Desks
            - The attacker calls a company’s help desk, pretends to be someone in a position of authority or relevance and tries to extract sensitive information from the help desk
            - “A man calls a company’s help desk and says he has forgotten his password. He adds that if he misses the deadline on a big advertising project, his boss might fire him.
            - The help desk worker feels sorry for him and quickly resets the password, unwittingly giving the attacker a clear entrance into the corporate network.”

- Eavesdropping

    - Unauthorized listening of conversations, or reading of messages
    - Interception of audio, video, or written communication

- Shoulder Surfing

    - Direct observation techniques such as looking over someone's shoulder to get information such as passwords, PINs, account numbers, etc.

- Dumpster Diving

    - Looking for treasure in someone else’s trash

- Reverse Social Engineering

    - The attacker presents him/herself as an authority and the target seeks his or her advice before or after offering the information that the attacker needs

- Piggybacking

    - An authorized person intentionally or unintentionally allows an unauthorized person to pass through a secure door e.g., “I forgot my ID badge at home. Please help me”

    - Tailgating

        - The attacker, wearing a fake ID badge, enters a secured area by closely following an authorized person through a door that requires key access

  - Computer-based Social Engineering

    - Pop-Up Windows
        - Windows that suddenly pop up while surfing the Internet and ask for user information to login or sign-in
    - Hoax Letters
        - Emails that issue warnings to the user about new viruses, Trojans, or worms that may harm the user's system
    - Chain Letters
        - Emails that offer free gifts such as money and software on condition that the user forwards the mail to a specified number of people
    - Instant Chat Messenger
        - Gathering personal information by chatting with a selected user online to get information such as birth dates and maiden names
    - Spam Email
        - Irrelevant, unwanted, and unsolicited emails that attempt to collect financial information, social security numbers, and network information
    - Scareware
        - Malware that tricks computer users into visiting malware infested websites, or downloading/buying potentially malicious software

  - Computer-based Social Engineering: Phishing

    - Phishing is the practice of sending an illegitimate email claiming to be from a legitimate site in an attempt to acquire a user's personal or account information
    - Phishing emails or pop-ups redirect users to fake webpages that mimic trustworthy sites, which ask them to submit their personal information

  - Types of Phishing

    - Spear Phishing
        - A targeted phishing attack aimed at specific individuals within an organization
    - Whaling
        - An attacker targets high profile executives like CEOs, CFOs, politicians, and celebrities who have complete access to confidential and highly valuable information
    - Pharming
        - The attacker redirects web traffic to a fraudulent website by installing a malicious program on a personal computer or server
    - Spimming
        - A variant of spam that exploits Instant Messaging platforms to flood spam across the networks

  - Examples of Phishing Emails https://its.tntech.edu

  - Phishing Tools

    - ShellPhish https://github.com
      - A phishing tool used to phish user credentials from various social networking platforms such as Instagram, Facebook, Twitter, LinkedIn, etc.
    - BLACKEYE https://github.com
    - PhishX https://github.com
    - Modlishka https://github.com
    - Trape https://github.com
    - Evilginx https://github.com

  - Mobile-based Social Engineering. Publishing Malicious Apps

    - Attackers create malicious apps with attractive features and similar names to popular apps, and publish them in major app stores
    - Users download these apps unknowingly and are infected by malware that sends credentials to attackers

  - Mobile-based Social Engineering: Repackaging Legitimate Apps

  - Mobile-based Social Engineering: Fake Security Applications

  - Mobile-based Social Engineering: SMiShing (SMS Phishing)

    - SMiShing (SMS phishing) is the act of using SMS text messaging system of cellular phones or other mobile devices to lure users into instant action, such as downloading malware, visiting a malicious webpage, or calling a fraudulent phone number
    - SMiShing messages are generally crafted to provoke an instant action from the victim, requiring them to divulge their personal information and account details

- Insider Threats and Identity Theft

  - Insider Threats/Insider Attacks

    - An insider is any employee (trusted person or people) who have access to critical assets of an organization
    - An insider attack involves using privileged access to intentionally violate rules or cause threat to the organization's information or information systems in any form
    - Such attacks are generally performed by a privileged user, disgruntled employee, terminated employee, accidentprone employee, third party, undertrained staff, etc.

  - Reasons for Insider Attacks

    - Financial Gain
    - Steal Confidential Data
    - Revenge
    - Become Future Competitor
    - Perform Competitors Bidding
    - Public Announcement

  - Types of Insider Threats

        - Malicious Insider
            - A disgruntled or terminated employee who steals data or destroys the company’s networks intentionally by introducing malware into the corporate network
        - Negligent Insider
            - Insiders who are uneducated on potential security threats or who simply bypass general security procedures to meet workplace efficiency
        - Professional Insider
            - Harmful insiders who use their technical knowledge to identify weaknesses and vulnerabilities in the company’s network and sell confidential information to competitors or black-market bidders
        - Compromised Insider
            - An insider with access to critical assets of an organization who is compromised by an outside threat actor

    - Why are Insider Attacks Effective?

        - Insider attacks are easy to launch.
        - Preventing insider attacks is difficult; an inside attacker can easily succeed.
        - It is very difficult to differentiate harmful actions from the employee’s regular work. It is hard to identify whether employees are performing malicious activities or not.
        - Even after malicious activity is detected, the employee may refuse to accept responsibility and claim it was a mistake.
        - It is easy for employees to cover their actions by editing or deleting logs to hide their malicious activities.
        - Insider attacks can go undetected for years, and remediation is expensive.
        - It is easy for insiders to access data or systems unrelated to their job role.
        - Insiders can easily misuse resources and steal intellectual property.
        - Insiders can bypass security constraints with minimal effort.
        - Insiders can easily obtain trade secrets and expose them to outsiders.

    - Identity Theft

        - Identity theft is a crime in which an imposter steals your personally identifiable information such as name, credit card number, social security or driver’s license numbers, etc. to commit fraud or other crimes

        - Attackers can use identity theft to impersonate employees of a target organization and physically access facilities

        - The attacker steals people’s identity for fraudulent purposes such as:

            - To open new credit card accounts in the name of the user without paying the bills
            - To open a new phone or wireless account in the user’s name
            - To use the victims’ information to obtain utility services such as electricity, heating, or cable TV
            - To open bank accounts with the intention of writing bogus checks using the victim’s information
            - To clone an ATM or debit card to make electronic withdrawals from the victim’s accounts

        - Types of Identity Theft
            - Child identity theft
            - Criminal identity theft
            - Financial identity theft
            - Driver’s license identity theft
            - Insurance identity theft
            - Medical identity theft
            - Tax identity theft
            - Identity cloning and Concealment
            - Synthetic identity theft
            - Social security identity theft

- Social Engineering Countermeasures
    - Password Policies
        - Periodic password changes
        - Avoiding guessable passwords
        - Account blocking after failed attempts
        - Increasing length and complexity of passwords
        - Improving secrecy of passwords
    - Physical Security Policies Defense Strategy
        - Identification of employees by issuing ID cards, uniforms, etc.
        - Escorting visitors
        - Restricting access to work areas
        - Proper shredding of useless documents
        - Employing security personnel
    - Defense Strategy
        - Social engineering campaign
        - Gap analysis
        - Remediation strategies
    - Insider Threats Countermeasures
        - Separation and rotation of duties
        - Least privileges
        - Controlled access
        - Logging and auditing
        - Employee monitoring
        - Legal policies
        - Archive critical data
        - Employee training on cybersecurity
        - Employee background verification
        - Privileged users monitoring
        - Credentials deactivation for terminated employees

    - Periodic risk assessments
    - Layered defense
    - Physical security
    - Surveillance

  - Identity Theft Countermeasures

    - Secure or shred all documents containing private information
    - Ensure your name is not present on the marketers' hit lists
    - Review your credit card statement regularly and store it securely, out of reach of others
    - Never give any personal information over the phone
    - To keep mail secure, empty the mailbox quickly
    - Suspect and verify all requests for personal data
    - Protect personal information from being publicized
    - Do not display account or contact numbers unless mandatory
    - Monitor online banking activities regularly
    - Never list any personal identifiers on social media websites such as your father's name, pet's name, address, or city of birth.
    - Enable two-factor authentication on all online accounts
    - Never use public Wi-Fi for sharing or accessing sensitive information
    - Install host security tools such as a firewall and anti-virus on your personal computer

  - How to Detect Phishing Emails?

    - It seems to be from a bank, company, or social networking site and has a generic greeting
    - It seems to be from a person listed in your email address book
    - It has an urgent tone or makes a veiled threat
    - It may contain grammatical or spelling mistakes
    - It includes links to spoofed websites
    - It may contain offers that seem to be too good to be true
    - It includes official-looking logos and other information taken from legitimate websites
    - It may contain a malicious attachment

  - Anti-Phishing Toolbar

    - Netcraft https://www.netcraft.com
      - The Netcraft anti-phishing community is a giant neighborhood watch scheme, empowering the most alert and most expert members to defend everyone within the community against phishing attacks
    - PhishTank https://www.phishtank.com
      - PhishTank is a collaborative clearing house for data and information about phishing on the Internet
      - It provides an open API for developers and researchers to integrate anti-phishing data into their apps

  - Social Engineering Tools:

    - Social Engineering Toolkit (SET) https://www.trustedsec.com

        - The Social-Engineer Toolkit (SET) is an open-source Python-driven tool aimed at penetration testing around social engineering
    - SpeedPhish Framework (SPF) (https://github.com)
    - Gophish (https://getgophish.com)
    - King Phisher (https://github.com)
    - LUCY (https://www.lucysecurity.com)
    - MSI Simple Phish (https://microsolved.com)
  - Audit Organization's Security for Phishing Attacks using OhPhish
    - OhPhish https://ohphish.eccouncil.org
    - OhPhish is a web-based portal to test employees’ susceptibility to social engineering attacks
    - OhPhish is a phishing simulation tool that provides the organization with a platform to launch phishing simulation campaigns on its employees

Module 06: Network Level Attacks and Countermeasures

- Sniffing.
- Packet Sniffing Concepts
  - Packet sniffing is the process of monitoring and capturing all data packets passing through a given network using a software application or hardware device
  - It allows an attacker to observe and access the entire network traffic from a given point in order to gather sensitive information such as Telnet passwords, email traffic, syslog traffic, etc.
  - How a Sniffer Works
    - A sniffer turns the NIC of a system to the promiscuous mode so that it listens to all the data transmitted on its segment
  - Types of Sniffing
    - Passive Sniffing
      - Passive sniffing refers to sniffing through a hub, wherein the traffic is sent to all ports
      - It involves monitoring packets sent by others without sending any additional data packets in the network traffic
    - Active Sniffing
      - Active sniffing is used to sniff a switch-based network
      - It involves injecting Address Resolution Packets (ARP) into the network to flood the switch’s Content Addressable Memory (CAM) table, which keeps track of host-port connections
      - Active Sniffing Techniques
        - MAC Flooding

            - DNS Poisoning
            - ARP Poisoning
            - DHCP Attacks
            - Switch Port Stealing
            - Spoofing Attack

    - How an Attacker Hacks the Network Using Sniffers

        - An attacker connects his desktop/laptop to a switch port
        - He/she identifies a victim's machine to target his/her attacks
        - The traffic destined for the victim's machine is redirected to the attacker
        - He/she runs discovery tools to learn about network topology
        - He/she poisons the victim's machine by using ARP spoofing techniques
        - The hacker extracts passwords and sensitive data from the redirected traffic

    - Protocols Vulnerable to Sniffing

        - Telnet and Rlogin
            - Keystrokes including usernames and passwords are sent in clear text
        - HTTP
            - Data is sent in clear text
        - POP
            - Passwords and data are sent in clear text
        - IMAP
            - Passwords and data are sent in clear text
        - SMTP and NNTP
            - Passwords and data are sent in clear text
        - FTP
            - Passwords and data are sent in clear text

- Sniffing Techniques

    - MAC flooding

        - MAC flooding involves the flooding of the CAM table with fake MAC address and IP pairs until it is full
        - The switch then acts as a hub by broadcasting packets to all machines on the network, and therefore, the attackers can sniff the traffic easily
        - Mac Flooding Switches with macof https://www.monkey.org
            - macof is a Unix/Linux tool that floods the switch's CAM tables (131,000 per min) by sending bogus MAC entries

    - DHCP starvation attack

        - DHCP is a configuration protocol that assigns valid IP addresses to host systems out of a pre-assigned DHCP pool
        - DHCP starvation attack is a process of inundating DHCP servers with fake DHCP requests and using all the available IP addresses
        - This results in a denial-of-service attack, where the DHCP server cannot issue new IP addresses to genuine host requests

  - ARP Spoofing Attack
    - Address Resolution Protocol (ARP) is a protocol used for mapping an IP address to a physical machine address which is recognized in the local network
    - ARP spoofing/poisoning involves sending a large number of forged entries to the target machine’s ARP cache
    - ARP Poisoning Tools
      - arpspoof https://linux.die.net
        - arpspoof –i [Interface] –t [Target Host]
        - arpspoof redirects packets from a target host (or all hosts) on the LAN intended for another host on the LAN by forging ARP replies
      - BetterCAP (https://www.bettercap.org)
      - Ettercap (http://www.ettercap-project.org)
      - dsniff (https://www.monkey.org)
      - MITMf (https://github.com)
      - Arpoison (https://sourceforge.net)
  - MAC spoofing/Duplicating
    - A MAC duplicating attack is launched by sniffing a network for MAC addresses of clients who are actively associated with a switch port and re-using one of those addresses
    - By listening to the traffic on the network, a malicious user can intercept and use a legitimate user's MAC address to receive all the traffic destined for the user
    - This attack allows an attacker to gain access to the network and take over someone’s identity on the network
  - DNS poisoning attacks
    - Domain Name Server (DNS) poisoning is the unauthorized manipulation of IP addresses in the DNS cache
    - A corrupted DNS redirects a user request to a malicious website to perform illegal activities
  - Sniffing tools
    - Wireshark https://www.wireshark.org
      - Wireshark lets you capture and interactively browse the traffic running on a computer network
    - SteelCentral Packet Analyzer https://www.riverbed.com
    - Capsa Network Analyzer https://www.colasoft.com
    - Observer Analyzer https://www.viavisolutions.com
    - PRTG Network Monitor https://www.paessler.com
    - SolarWinds Deep Packet Inspection and Analysis https://www.solarwinds.com
- Sniffing Countermeasures
  - Restrict physical access to the network media to ensure that a packet sniffer cannot be installed
  - Use end-to-end encryption to protect confidential information

  - Permanently add the MAC address of the gateway to the ARP cache
  - Use static IP addresses and ARP tables to prevent attackers from adding the spoofed ARP entries for machines in the network
  - Turn off network identification broadcasts and, if possible, restrict the network to authorized users to protect the network from being discovered with sniffing tools
  - Use IPv6 instead of IPv4
  - Use encrypted sessions such as SSH instead of telnet, Secure Copy (SCP) instead of FTP, and SSL for email connection to protect wireless network users against sniffing attacks
  - Use HTTPS instead of HTTP to protect usernames and passwords
  - Use a switch instead of the hub, as a switch delivers data only to the intended recipient
  - Use Secure File Transfer Protocol (SFTP) instead of FTP for secure transfer of files
  - Use PGP and S/MIME, VPN, IPSec, SSL/TLS, SSH, and one-time passwords (OTP)
  - Use POP2 or POP3 instead of POP to download emails from email servers
  - Use SNMPv3 instead of SNMPv1 and SNMPv2 to manage networked devices
  - Always encrypt the wireless traffic with a strong encryption protocol such as WPA or WPA2
  - Retrieve MAC addresses directly from NICs instead of the OS; this prevents MAC address spoofing
  - Use tools to determine if any NICs are running in promiscuous mode
  - Use the concept of Access Control List (ACL) to allow access only to a fixed range of trusted IP addresses in a network
  - Change default passwords to complex passwords
  - Avoid broadcasting SSIDs (Session Set Identifiers)
  - Implement a MAC filtering mechanism on your router
  - Implement network scanning and monitoring tools to detect malicious intrusions, rogue devices, and sniffers connected to the network
  - Sniffer Detection Techniques: Ping Method
    - Sends a ping request to the suspect machine with its IP address and an incorrect MAC address. The Ethernet adapter rejects it, as the MAC address does not match, whereas the suspect machine running the sniffer responds to it as it does not reject packets with a different MAC address
  - Sniffer Detection Techniques: DNS Method

        - Most of the sniffers perform reverse DNS lookups to identify the machine from the IP address
        - A machine generating reverse DNS lookup traffic is very likely to be running a sniffer

    - Sniffer Detection Techniques: ARP Method

        - Only the machine in the promiscuous mode (machine C) caches the ARP information (IP and MAC address mapping)
        - A machine in the promiscuous mode responds to the ping message as it has the correct information about the host sending the ping requests in its cache; the rest of the machines will send an ARP probe to identify the source of the ping request

- Denial-of-Service

    - DoS and DDoS Attacks

        - What is a DoS Attack?
            - Denial-of-Service (DoS) is an attack on a computer or network that reduces, restricts, or prevents accessibility of system resources to its legitimate users
            - Attackers flood the victim system with non-legitimate service requests or traffic to overload its resources
        - What is a DDoS Attack? https://searchsecurity.techtarget.com
            - Distributed denial-of-service (DDoS) is a coordinated attack that involves a multitude of compromised systems (Botnet) attacking a single target, thereby denying service to users of the targeted system

    - DoS/DDoS Attack Techniques: UDP Flood Attack

        - An attacker sends spoofed UDP packets at a very high packet rate to a remote host on random ports of a target server using a large source IP range
        - The flooding of UDP packets causes the server to repeatedly check for non-existent applications at the ports
        - Legitimate applications are inaccessible by the system and give an error reply with an ICMP “Destination Unreachable” packet
        - This attack consumes network resources and available bandwidth, exhausting the network until it goes offline

    - DoS/DDoS Attack Techniques: ICMP Flood Attack

        - ICMP flood attacks are a type of attack in which attackers send large volumes of ICMP echo request packets to a victim system directly or through reflection networks
        - These packets signal the victim’s system to reply, and the resulting combination of traffic saturates the bandwidth of the victim’s network connection, causing it to be overwhelmed and subsequently stop responding to legitimate TCP/IP requests

    - DoS/DDoS Attack Techniques: Ping of Death Attack

        - In a Ping of Death (PoD) attack, an attacker tries to crash, destabilize, or freeze the targeted system or service by sending malformed or oversized packets using a simple ping command

        - For instance, the attacker sends a packet which has a size of 65,538 bytes to the target web server. This packet size exceeds the size limit prescribed by RFC 791 IP, which is 65,535 bytes.

    - DoS/DDoS Attack Techniques: Smurf Attack

        - The attacker spoofs the source IP address with the victim’s IP address and sends a large number of ICMP ECHO request packets to an IP broadcast network
        - This causes all the hosts on the broadcast network to respond to the received ICMP ECHO requests. These responses will be sent to the victim machine, ultimately causing the machine to crash

    - DoS/DDoS Attack Techniques: SYN Flood Attack

        - The attacker sends a large number of SYN requests with fake source IP addresses to the target server (victim)
        - The target machine sends back a SYN/ACK in response to the request and waits for the ACK to complete the session setup
        - The target machine does not get the response because the source address is fake
        - SYN flooding takes advantage of a flaw in the implementation of the TCP three-way handshake in most hosts

    - DoS/DDoS Attack Techniques: Fragmentation Attack

        - These attacks stop a victim from being able to re-assemble fragmented packets by flooding the target system with TCP or UDP fragments, resulting in reduced performance
        - Attackers send a large number of fragmented (1500+ byte) packets to a target web server with a relatively small packet rate
        - Reassembling and inspecting these large fragmented packets consumes excessive resources

    - DoS/DDoS Attack Techniques: Multi-Vector Attack

        - The attackers use combinations of volumetric, protocol, and application-layer attacks to disable the target system or service
        - Attackers rapidly and repeatedly change the form of their DDoS attack (e.g., SYN packets, Layer 7)

    - DoS/DDoS Attack Techniques: Peer-to-Peer Attack

        - Attackers instruct clients of peer-to-peer file sharing hubs to disconnect from their peer-to-peer network and to connect to the victim's fake website
        - Attackers exploit flaws found in the network using the DC++ (Direct Connect) protocol, which is used for sharing all types of files between instant messaging clients
        - Using this method, attackers launch massive denial-of-service attacks and compromise websites

    - DoS/DDoS Attack Techniques: Permanent Denialof-Service Attack

        - Permanent DoS, also known as phlashing, refers to attacks that cause irreversible damage to system hardware

        - Unlike other DoS attacks, it sabotages the system hardware, requiring the victim to replace or reinstall the hardware
        - This attack is carried out using a method known as “bricking a system”
        - Using this method, attackers send fraudulent hardware updates to the victims

    - DoS/DDoS Attack Techniques: Distributed Reflection Denial-of-Service (DRDoS) Attack

        - DRDoS, also known as a spoofed attack, involves the use of multiple intermediary and secondary machines that contribute to the actual DDoS attack against the target machine or application
        - Attackers launch this attack by sending requests to the intermediary hosts, which then redirect the requests to the secondary machines, which in turn reflect the attack traffic to the target

    - DoS/DDoS Attack Tools

        - hping3
            - A command-line-oriented network scanning and packet crafting tool for the TCP/IP protocol that sends ICMP echo requests and supports TCP, UDP, ICMP, and raw-IP protocols
        - High Orbit Ion Cannon (HOIC) https://sourceforge.net
            - HOIC carries out a DDoS to attack any IP address with a user selected port and a user selected protocol
        - Low Orbit Ion Cannon (LOIC) https://sourceforge.net
            - LOIC can be used on a target site to flood the server with TCP packets, UDP packets, or HTTP requests with the intention of disrupting the service of a particular host
        - XOIC (http://anonhacktivism.blogspot.com)
        - HULK (https://siberianlaika.ru)
        - Tor’s Hammer (https://sourceforge.net)
        - Slowloris (https://github.com)
        - PyLoris (https://sourceforge.net)
        - R-U-Dead-Yet (https://sourceforge.net)

    - DoS and DDoS Attack Countermeasures

        - Use strong encryption mechanisms such as WPA2 and AES 256 for broadband networks to defend against eavesdropping
        - Ensure that the software and protocols are up-to-date and scan the machines thoroughly to detect any anomalous behavior
        - Update the kernel to the latest release and disable unused and insecure services
        - Block all inbound packets originating from the service ports to block the traffic from reflection servers
        - Enable TCP SYN cookie protection
        - Prevent the transmission of fraudulently addressed packets at the ISP level
        - Implement cognitive radios in the physical layer to handle jamming and scrambling attacks
        - Configure the firewall to deny external ICMP traffic access
        - Secure remote administration and connectivity testing

        - Perform thorough input validation
        - Stop data processed by the attacker from being executed
        - Prevent the use of unnecessary functions such as gets and strcpy
        - Prevent the return addresses from being overwritten

    - DoS/DDoS Protection Tools

        - Anti DDoS Guardian http://www.beethink.com
            - A DDoS attack protection tool that protects IIS servers, Apache serves, game servers, Camfrog servers, mail servers, etc.
        - Imperva DDoS Protection (https://www.imperva.com)
        - DOSarrest’s DDoS protection service (https://www.dosarrest.com)
        - DDoS-GUARD (https://ddos-guard.net)
        - Cloudflare (https://www.cloudflare.com)
        - F5 (https://f5.com)

- Session Hijacking

    - Session Hijacking Attacks

        - Session hijacking refers to an attack in which an attacker seizes control of a valid TCP communication session between two computers
        - As most authentications only occur at the start of a TCP session, this allows the attacker to gain access to a machine
        - Attackers can sniff all the traffic from the established TCP sessions and perform identity theft, information theft, fraud, etc.
        - The attacker steals a valid session ID and uses it to authenticate himself with the server

    - Why is Session Hijacking Successful?

        - Absence of account lockout for invalid session IDs
        - Weak session-ID generation algorithm or small session IDs
        - Insecure handling of session IDs
        - Indefinite session timeout
        - Most computers using TCP/IP are vulnerable
        - Most countermeasures do not work without encryption

    - Session Hijacking Process

        - Tracking the connection
        - Desynchronizing the connection
        - Injecting the attacker's packet

    - Types of Session Hijacking

        - Passive Session Hijacking
            - In a passive attack, an attacker hijacks a session but sits back, watches, and records all the traffic in that session
        - Active Session Hijacking
            - In an active attack, an attacker finds an active session and seizes control of it

- Session Hijacking in OSI Model

    - Network Level Hijacking
        - Defined as the interception of packets during the transmission between a client and the server in a TCP or UDP session
    - Application Level Hijacking
        - Refers to gaining control over the HTTP's user session by obtaining the session IDs

- Spoofing vs. Hijacking

    - Spoofing Attack
        - An attacker pretends to be another user or machine (victim) to gain access
        - The attacker does not seize control of an existing active session; instead, he or she initiates a new session using the victim's stolen credentials
    - Hijacking
        - Session hijacking is the process of seizing control of an existing active session
        - The attacker relies on the legitimate user to create a connection and authenticate

- Session Hijacking Tools

    - OWASP ZAP https://owasp.org
        - An integrated penetration testing tool for finding vulnerabilities in web applications
    - Burp Suite (https://portswigger.net)
    - bettercap (https://www.bettercap.org)
    - netool toolkit (https://sourceforge.net)
    - WebSploit Framework (https://sourceforge.net)
    - sslstrip (https://pypi.org)

- Session Hijacking Attack Countermeasures

    - Session Hijacking Detection Methods

        - Manual Method
        - Automatic Method
            - IDS
            - IPS

    - Session Hijacking Countermeasures

        - Use the Secure Shell (SSH) to create a secure communication channel.
        - Pass authentication cookies over HTTPS connections.
        - Implement the log-out functionality for the user to end the session.
        - Generate a session ID after a successful login and accept session IDs generated by the server only.
        - Ensure that data in transit are encrypted and implement the defense-in-depth mechanism.
        - Use strings or long random numbers as session keys.
        - Use different usernames and passwords for different accounts.
        - Educate employees and minimize remote access.
        - Implement timeout() to destroy sessions when expired.
        - Avoid including the session ID in the URL or query string.

        - Use switches rather than hubs and limit incoming connections.
        - Ensure client-side and server-side protection software are in the active state and up to date.
        - Use strong authentication (such as Kerberos) or peer-to-peer virtual private networks (VPNs).
        - Configure appropriate internal and external spoof rules on gateways.
        - Use encrypted protocols available in the OpenSSH suite.
        - Use firewalls and browser settings to confine cookies.
        - Protect authentication cookies with SSL.
        - Regularly update platform patches to fix TCP/IP vulnerabilities (e.g., predictable packet sequences).
        - Use IPsec to encrypt session information.
        - Use HTTP Public Key Pinning (HPKP) to allow users to authenticate web servers.
        - Enable browsers to verify website authenticity using network notary servers.
        - Implement DNS-based authentication of named entities.
        - Disable compression mechanisms of HTTP requests.
        - Use cipher-chaining block (CBC) ciphers incorporating random padding up to 255 bytes, thereby making the extraction of confidential information difficult for an attacker.
        - Restrict the cross-site scripts known as cross-site request forgery (CSRF) from the client side.
        - Upgrade web browsers to the latest versions.
        - Use vulnerability scanners such as masscan to detect any insecure configuration of HTTPS session settings on sites.

    - Session Hijacking Detection Tools

        - Wireshark https://www.wireshark.org
            - Wireshark allows you to capture and interactively browse the traffic running on a computer network
        - USM Anywhere (https://cybersecurity.att.com)
        - Check Point IPS (https://www.checkpoint.com)
        - LogRhythm (https://logrhythm.com)
        - SolarWinds Security Event Manager (SEM) (https://www.solarwinds.com)
        - IBM Security Network Intrusion Prevention System (https://www.ibm.com)

Module 07: Web Application Attacks and Countermeasures

- Web Server Operations

    - A web server is a computer system that stores, processes, and delivers web pages to clients via HTTP

    - Web Server Components

        - Document Root

        - Stores critical HTML files related to the web pages of a domain name that will be served in response to the requests
    - Server Root
        - Stores server's configuration, error, executable, and log files
    - Virtual Document Tree
        - Provides storage on a different machine or disk after the original disk is filled up
    - Virtual Hosting
        - Technique of hosting multiple domains or websites on the same server
    - Web Proxy
        - Proxy server that sits between the web client and web server to prevent IP blocking and maintain anonymity

- Web Server Security Issues

    - Attackers usually target software vulnerabilities and configuration errors to compromise web servers

    - Network and OS level attacks can be well defended using proper network security measures such as firewalls, IDS, etc. However, web servers can be accessed from anywhere via the Internet, which renders them highly vulnerable to attacks

    - Common Goals behind Web Server Hacking

        - Stealing credit-card details or other sensitive credentials using phishing techniques
        - Integrating the server into a botnet to perform denial of service (DoS) or distributed DoS (DDoS) attacks
        - Compromising a database
        - Obtaining closed-source applications
        - Hiding and redirecting traffic
        - Escalating privileges

    - Dangerous Security Flaws Affecting Web Server Security

        - Failing to update the web server with the latest patches
        - Using the same system administrator credentials everywhere
        - Allowing unrestricted internal and outbound traffic
        - Running unhardened applications and servers

- Impact of Web Server Attacks

    - Compromise of user accounts
    - Website defacement
    - Secondary attacks from the website
    - Root access to other applications or server
    - Data tampering
    - Data theft
    - Damage reputation of the company

- Why are Web Servers Compromised?

        - Improper file and directory permissions
        - Installing the server with default settings
        - Unnecessary services enabled, including content management and remote administration
        - Security conflicts with the business' ease-of-use requirements
        - Lack of proper security policy, procedures, and maintenance
        - Improper authentication with external systems
        - Default accounts with default or no passwords
        - Unnecessary default, backup, or sample files
        - Misconfigurations in the web server, OS, and networks
        - Bugs in server software, OS, and web applications
        - Misconfigured Secure Sockets Layer (SSL) certificates and encryption settings
        - Administrative or debugging functions that are enabled or accessible on web servers
        - Use of self-signed certificates and default certificates

- Web Server Attacks

    - Web Server Attacks: DNS Server Hijacking

        - Attacker compromises the DNS server and changes the DNS settings so that all the requests coming towards the target web server are redirected to his/her own malicious server

    - Web Server Attacks: DNS Amplification Attack

        - Attacker takes advantage of the DNS recursive method of DNS redirection to perform DNS amplification attacks
        - Attacker uses compromised PCs with spoofed IP addresses to amplify the DDoS attacks on victims' DNS server by exploiting the DNS recursive method

    - Web Server Attacks: Directory Traversal Attacks

        - In directory traversal attacks, attackers use the ../ (dot-dot-slash) sequence to access restricted directories outside the web server root directory
        - Attackers can use the trial and error method to navigate outside the root directory and access sensitive information in the system

    - Web Server Attacks: Website Defacement

        - Web defacement occurs when an intruder maliciously alters the visual appearance of a web page by inserting or substituting provocative, and frequently, offending data
        - Defaced pages expose visitors to some propaganda or misleading information until the unauthorized changes are discovered and corrected

    - Web Server Attacks: Web Server Misconfiguration

        - Server misconfiguration refers to configuration weaknesses in web infrastructure that can be exploited to launch various attacks on web servers such as directory traversal, server intrusion, and data theft
        - The following are some web server misconfigurations:
            - Verbose debug/error messages

            - Anonymous or default users/passwords
            - Sample configuration and script files
            - Remote administration functions
            - Unnecessary services enabled
            - Misconfigured/default SSL certificates

    - Web Server Attacks: HTTP Response-Splitting Attack

        - HTTP response splitting attack involves adding header response data into the input field so that the server splits the response into two responses
        - The attacker can control the first response to redirect the user to a malicious website whereas the other responses are discarded by the web browser

    - Web Server Attacks: Web Cache Poisoning Attack

        - Web cache poisoning attacks the reliability of an intermediate web cache source
        - In this attack, the attackers swap cached content for a random URL with infected content
        - Users of the web cache source can unknowingly use the poisoned content instead of the true and secured content when requesting the required URL through the web cache

    - Web Server Attacks: SSH Brute Force Attack

        - SSH protocols are used to create an encrypted SSH tunnel between two hosts to transfer unencrypted data 01 over an insecure network
        - Attackers can brute force SSH login credentials to gain unauthorized access to an SSH tunnel
        - SSH tunnels can be used to transmit malwares and other exploits to victims without being detected

    - Web Server Attacks: Web Server Password Cracking

        - An attacker tries to exploit weaknesses to hack well-chosen passwords
        - The most common passwords found are password, root, administrator, admin, demo, test, guest, qwerty, pet names, etc.
        - Attackers use different methods such as social engineering, spoofing, phishing, using a Trojan Horse or virus, wiretapping, and keystroke logging
        - Passwords can be cracked manually by guessing or by performing dictionary, brute force, and hybrid attacks using automated tools such as THC Hydra, and Ncrack
        - THC Hydra https://github.com

    - Web Server Attacks: Server-Side Request Forgery (SSRF) Attack

        - Attackers exploit SSRF vulnerabilities in a public web server to send crafted requests to the internal or back end servers
        - Once the attack is successfully performed, the attackers can perform various activities such as port scanning, network scanning, IP address discovery, reading web server files, and bypassing host-based authentication

    - Web Server Attack Tools

        - Metasploit https://www.metasploit.com

        - An exploit development platform that supports fully automated exploitation of web servers, by abusing known vulnerabilities and leveraging weak passwords via Telnet, SSH, HTTP, and SNM
    - Immunity's CANVAS (https://www.immunityinc.com)
    - THC Hydra (https://github.com)
    - HULK DoS (https://github.com)
    - MPack (https://sourceforge.net)
    - w3af (https://w3af.org)

- Web Server Attack Countermeasures

  - Apply restricted ACLs and block remote registry administration.

  - Secure the SAM (stand-alone servers only).

  - Ensure that security-related settings are configured appropriately and that access to the metabase file is restricted with hardened NTFS permissions.

  - Remove unnecessary Internet Server Application Programming Interface (ISAPI) filters from the web server.

  - Remove all unnecessary file shares including the default administration shares, if they are not required.

  - Secure the shares with restricted NTFS permissions.

  - Relocate sites and virtual directories to non-system partitions and use IIS web permissions to restrict access.

  - Remove all unnecessary IIS script mappings for optional file extensions to avoid exploitation of any bugs in the ISAPI extensions that handle these types of files.

  - Enable a minimum level of auditing on the web server and use NTFS permissions to protect log files.

  - Use a dedicated machine as a web server.

  - Create URL mappings to internal servers cautiously.

  - Do not install the IIS server on a domain controller.

  - Use server-side session ID tracking and match connections with timestamps, IP addresses, etc.

  - If a database server, such as Microsoft SQL Server, is to be used as a backend database, install it on a separate server.

  - Use security tools provided with web server software and scanners that automate and simplify the process of securing a web server.

  - Physically protect the web server machine in a secure machine room.

  - Do not connect an IIS Server to the Internet until it is fully hardened.

    - Do not allow anyone to locally log in to the machine except the administrator.
    - Configure a separate anonymous user account for each application, if multiple web applications are hosted.
    - Limit the server functionality to support only the web technologies to be used.
    - Screen and filter incoming traffic requests.
    - Store website files and scripts on a separate partition or drive.
    - Web Server Security Tools
        - Fortify WebInspect Source: https://www.microfocus.com
            - Fortify WebInspect is an automated dynamic testing solution that discovers configuration issues and identifies and prioritizes security vulnerabilities in running applications
        - Acunetix Web Vulnerability Scanner (https://www.acunetix.com)
        - Retina Host Security Scanner (https://www.beyondtrust.com)
        - NetIQ Secure Configuration Manager (https://www.netiq.com)
        - SAINT Security Suite (https://www.carson-saint.com)
        - Sophos Intercept X for Server (https://www.sophos.com)
- Web Application Attacks
- Web Application Architecture and Vulnerability Stack
    - Introduction to Web Applications
        - Web applications provide an interface between end users and web servers through a set of web pages that are generated at the server end or contain script code to be executed dynamically within the client web browser
        - Though web applications enforce certain security policies, they are vulnerable to various attacks such as SQL injection, cross-site scripting, and session hijacking
        - The advantages of web applications are listed below:
            - As they are independent of the operating system, their development and troubleshooting are easy and cost-effective.
            - They are accessible anytime and anywhere using a computer with an Internet connection.
            - The user interface is customizable, making it easy to update.
            - Users can access them on any device having an Internet browser, including PDAs, smartphones, etc.
            - Dedicated servers, monitored and managed by experienced server administrators, store all the web application data, allowing developers to increase their workload capacity.
            - Multiple locations of servers not only increase physical security but also reduce the burden of monitoring thousands of desktops using the program.

        - They use flexible core technologies, such as JSP, Servlets, Active Server Pages, SQL Server, .NET, and scripting languages, which are scalable and support even portable platforms.

  - How Web Applications Work

  - Web Application Architecture

  - Web Services

    - A web service is an application or software that is deployed over the Internet and uses standard messaging protocols such as SOAP, UDDI, WSDL, and REST to enable communication between applications developed for different platforms

    - There are three roles in a web service:

      - Service Provider
      - Service Requester
      - Service Registry

    - There are three operations in a web service architecture:

      - Publish
      - Find
      - Bind

    - There are two artifacts in a web service architecture:

      - Service
      - Service Description

    - Components of Web Service Architecture:

      - UDDI: Universal Description, Discovery, and Integration (UDDI)
      - WSDL: Web Services Description Language
      - WS-Security

    - Characteristics of Web Services

      - XML-based
      - Coarse-grained service
      - Loosely coupled
      - Asynchronous and synchronous support
      - RPC support

    - Types of Web Services

      - SOAP web services
        - It is based on the XML format and is used to transfer data between a service provider and requestor
      - RESTful web services

            - It is based on a set of constraints using underlying HTTP concepts to improve performance

        - Vulnerability Stack

- Web Application Threats and Attacks

    - OWASP Top 10 Application Security Risks https://www.owasp.org

        - Injection
        - Broken Authentication
        - Sensitive Data Exposure
        - XML External Entity (XXE)
        - Broken Access Control
        - Security Misconfiguration
        - Cross-Site Scripting (XSS)
        - Insecure Deserialization
        - Using Components with Known Vulnerabilities
        - Insufficient Logging and Monitoring

    - Injection Flaws

        - Injection flaws are web application vulnerabilities that allow untrusted data to be interpreted and executed as part of a command or query

        - Attackers exploit injection flaws by constructing malicious commands or queries that result in data loss or corruption, lack of accountability, or denial of access

        - SQL Injection

            - It involves the injection of malicious SQL queries into user input forms

        - Command Injection

            - It involves the injection of malicious code through a web application

        - LDAP Injection

            - It involves the injection of malicious LDAP statements

    - Broken Authentication

        - Attackers can exploit vulnerabilities in authentication or session management functions such as exposed accounts, session IDs, logout, password management, timeouts, etc. to impersonate users

        - Session ID in URLs

            - Attackers sniff the network traffic or trick users to get session IDs and then reuse those session IDs for malicious purposes

        - Password Exploitation

        - Attackers can gain access to a web application's password database. If user passwords are not encrypted, an attacker can exploit any user’s password

    - Timeout Exploitation

        - If an application’s timeouts are not set properly and a user closes their browser without logging out from sites accessed through a public computer, an attacker can use the same browser later and exploit that user's privileges

- Sensitive Data Exposure

    - Sensitive data exposure occurs due to flaws like insecure cryptographic storage and information leakage
    - When an application uses poorly written encryption code to securely encrypt and store sensitive data in the database, an attacker can exploit this flaw and steal or modify weakly protected sensitive data such as credit cards numbers, SSNs, and other authentication credentials

- XML External Entity (XXE)

    - XML External Entity attack is a server-side request forgery (SSRF) attack that can occur when a misconfigured XML parser allows applications to parse XML input from an unreliable source
    - Attackers can a refer a victim’s web application to an external entity by including the reference in the malicious XML input
    - When this malicious input is processed by the weakly configured XML parser of a target web application, it enables the attacker to access protected files and services from servers or connected networks

- Broken Access Control

    - Broken access control is a method in which an attacker identifies a flaw related to access control and bypasses the authentication, which allows them to compromise the network

    - It allows an attacker to act as users or administrators with privileged functions and create, access, update or delete every record

    - Insecure Direct Object References

    - Missing Function Level Access Control

- Security Misconfiguration

    - Unvalidated Inputs
        - It refers to a web application vulnerability where input from a client is not validated before being processed by web applications and backend servers
    - Parameter/Form Tampering
        - It involves the manipulation of parameters exchanged between client and server to modify application data
    - Improper Error Handling

    - It gives insight into source code such as logic flaws, and default accounts. Using the information received from an error message, an attacker identifies vulnerabilities to launch various web application attacks
  - Insufficient Transport Layer Protection
    - It supports weak algorithms and uses expired or invalid certificates. Using insufficient transport layer protection exposes user data to untrusted third parties and can lead to account theft

- Cross-Site Scripting (XSS) Attacks

  - Cross-site scripting ('XSS' or 'CSS') attacks exploit vulnerabilities in dynamically generated web pages, enabling malicious attackers to inject clientside scripts into web pages viewed by other users
  - It occurs when unvalidated input data is included in dynamic content that is sent to a user's web browser for rendering
  - Some exploitations that can be performed by XSS attacks are as follows:
    - Malicious script execution
    - Redirecting to a malicious server
    - Exploiting user privileges
    - Ads in hidden IFRAMES and pop-ups
    - Data manipulation
    - Session hijacking
    - Brute-force password cracking
    - Data theft
    - Intranet probing
    - Keylogging and remote monitoring

- Insecure Deserialization

  - Data serialization and deserialization is an effective process of linearizing and delinearizing data objects for transmission to other networks or systems
  - Attackers inject malicious code into serialized data and forward the malicious serialized data to the victim
  - Insecure deserialization deserializes the malicious serialized content along with the injected malicious code, compromising the system or network

- Using Components with Known Vulnerabilities

  - Most web applications that use components such as libraries and frameworks always execute them with full privileges, and flaws in any component can result in serious impact
  - Attackers can identify weak components or dependencies by scanning or by performing manual analysis
  - Attackers search for any vulnerabilities on exploit sites such as Exploit Database (https://www.exploit-db.com), and SecurityFocus (https://www.securityfocus.com)
  - If a vulnerable component is identified, the attacker customizes the exploit as required and execute the attack

- Insufficient Logging and Monitoring

        - Web applications maintain logs to track usage patterns, such as user login credentials and admin login credentials
        - Insufficient logging and monitoring refer to the scenario where the detection software either does not record the malicious event or ignores important details about the event
        - Attackers usually inject, delete, or tamper the web application logs to engage in malicious activities or hide their identities

    - Web Application Attack Tools

        - Burp Suite https://portswigger.net
            - Support the entire web application testing process, from initial mapping and analysis of an application's attack surface to finding and exploiting security vulnerabilities
        - OWASP Zed Attack Proxy (ZAP) https://www.owasp.org
            - Provides automated scanners and tools that allow you to find security vulnerabilities manually
        - Metasploit (https://www.metasploit.com)
        - w3af (http://w3af.org)
        - Nikto (https://cirt.net)
        - Sn1per (https://github.com)
        - WSSiP (https://github.com)

- Web Application Attack Countermeasures

    - SQL Injection Attacks

        - Limit the length of the user input
        - Use custom error messages
        - Monitor DB traffic using an IDS, WAF
        - Disable commands such as xp_cmdshell
        - Isolate the database server and web server
        - Always use a method attribute set for POST and low-privileged account for DB connection
        - Run a database service account with minimal rights
        - Move extended stored procedures to an isolated server
        - Use typesafe variables or functions such as isNumeric() to ensure typesafety
        - Validate and sanitize user inputs passed to the database
        - Avoid using dynamic SQL and do not construct queries with the user input
        - Use prepared statements, parameterized queries, or stored procedures to access the database
        - Display less information and use the "RemoteOnly" customErrors mode to display verbose error messages on the local machine
        - Perform proper escaping and character filtering to avoid special string characters and symbols such as single quotes (')
        - Always set the whitelist logically instead of the blacklist to avoid bad code
        - Use Object Relational Mapping (ORM) frameworks to make the conversion of SQL result sets into code objects more consistent

    - Command Injection Flaws

    - Perform input validation
    - Escape dangerous characters
    - Use language-specific libraries that avoid problems due to shell commands
    - Perform input and output encoding
    - Use a safe API that avoids use of the interpreter entirely
    - Structure requests so that all supplied parameters are treated as data rather than potentially executable content
    - Use parameterized SQL queries
    - Use modular shell disassociation from the kernel
    - Use built-in library functions and avoid calling OS commands directly
    - Implement the least privileges to restrict the permissions to execute the OS commands
    - Avoid executing commands such as exec or system without proper validation and sanitization
    - Prevent the shell interpreter using pcntl_fork and pcntl_exec within the PHP
    - Implement Python as a web framework instead of PHP for application development

  - LDAP Injection Attacks

    - Perform type, pattern, and domain value validation on all input data
    - Make the LDAP filter as specific as possible
    - Validate and restrict the amount of data returned to the user
    - Implement tight access control on the data in the LDAP directory
    - Perform dynamic testing and source code analysis
    - Sanitize all the user-end inputs and escape any special characters
    - Avoid constructing LDAP search filters by concatenating strings
    - Use the AND filter to enforce restrictions on similar entries
    - Use LDAPS (LDAP over SSL) for encrypting and securing the communication on the web servers

  - Broken Authentication and Session Management

    - Use SSL for all authenticated parts of the application
    - Verify whether all the users’ identities and credentials are stored in a hashed form
    - Never submit session data as part of a GET, POST
    - Apply pass phrasing with at least five random words
    - Limit the login attempts and lock the account for a specific period after a certain number of failed attempts
    - Use a secure platform session manager to generate long random session identifiers for secure session development
    - Implement multi-factor authentication mechanisms to prevent guessing, credential stuffing, and brute-forcing
    - Make sure to secure passwords with a cryptographic password hash algorithm or tools such as bcrypt, scrypt, or Argon2
    - Make sure to check weak passwords against a list of the top bad passwords
    - Log authentication failures and send alerts whenever probable attacks are detected

  - Sensitive Data Exposure

    - Do not create or use weak cryptographic algorithms
    - Generate encryption keys offline and store them securely
    - Ensure that encrypted data stored on the disk is not easy to decrypt
    - Use AES encryption for stored data and use TLS with HSTS (HTTP Strict Transport Security) for incoming traffic
    - Classify the data processed, stored, or transmitted by an application and apply controls accordingly
    - Use PCI DSS compliant tokenization or truncation to remove the data soon after its requirement
    - Use proper key management and ensure that all the keys are in place
    - Encrypt all the data in transit using TLS with Perfect Forward Secrecy (PFS) ciphers
    - Disable caching techniques for requests that contain sensitive information

  - XML External Entity

    - Avoid processing XML input containing references to external entities by a weakly configured XML parser
    - XML unmarshaller should be configured securely
    - Parse the document with a securely configured parser
    - Configure the XML processor to use local static DTD and disable any declared DTD included in an XML document
    - Implement whitelisting, input validation, sanitation, and filtering techniques to prevent hostile data within the XML documents
    - Update and patch the latest XML processors and libraries
    - Make sure that the XML/XLS file upload function validates the XML using XSD validation

  - Broken Access Control

    - Perform access control checks before redirecting the authorized user to the requested resource
    - Avoid using insecure IDs to prevent the attacker from guessing them
    - Provide a session timeout mechanism
    - Limit file permissions to authorized users to avoid misuse
    - Avoid client-side caching mechanisms
    - Remove session tokens on the server side on user logout
    - Ensure that minimum privileges are assigned to users to perform only essential actions
    - Enforce access control mechanisms once and re-use them throughout the application

  - Security Misconfiguration

    - Configure all security mechanisms and disable all unused services
    - Setup roles, permissions, and accounts and disable all default accounts or change their default passwords
    - Scan for the latest security vulnerabilities and apply the latest security patches
    - Non-SSL requests to web pages should be redirected to the SSL page
    - Set the ‘secure’ flag on all sensitive cookies
    - Configure the SSL provider to support only strong algorithms

    - Ensure that the certificate is valid and not expired, and that it matches all domains used by the site
    - Backend and other connections should also use SSL or other encryption technologies

  - XSS Attacks

    - Validate all headers, cookies, query strings, form fields, and hidden fields (i.e., all parameters) against a rigorous specification
    - Use testing tools extensively during the design phase to eliminate such XSS holes in the application before it goes into use
    - Use a web application firewall to block the execution of a malicious script
    - Convert all non-alphanumeric characters into HTML character entities before displaying the user input in search engines and forums
    - Encode the input and output and filter metacharacters in the input
    - Never trust websites that use HTTPS when it comes to XSS
    - Filtering the script output can also defeat XSS vulnerabilities by preventing them from being transmitted to users
    - Deploy public key infrastructure (PKI) for authentication, which checks to ascertain that the script introduced is actually authenticated
    - Implement a stringent security policy
    - Web servers, application servers, and web application environments are vulnerable to cross-site scripting. It is difficult to identify and remove XSS flaws from web applications. The best way to find flaws is to perform a security review of the code and search in all the places where the input from an HTTP request comes as an output through HTML.
    - Attacker uses a variety of HTML tags to transmit a malicious JavaScript. Nessus, Nikto, and other tools can help to some extent in scanning websites for these flaws. If the scanning discovers a vulnerability in a website, it is highly likely to be vulnerable to other attacks.

  - Insecure Deserialization

    - Validate untrusted input that is to be serialized to ensure that the serialized data contains only trusted classes
    - Deserialization of trusted data must cross a trust boundary
    - Developers must re-architect their applications
    - Avoid serialization for security-sensitive classes
    - Guard sensitive data during deserialization
    - Filter untrusted serial data
    - Enforce duplicate security manager checks in a class during serialization and deserialization
    - Understand the security permissions given to serialization and deserialization
    - Implement integrity checks or encryption of the serialized objects to prevent data modification or hostile object creation
    - Isolate code that deserializes so that it runs in very-low-privileged environments
    - Log the deserialization exceptions and failures so that the incoming type is not the same as the expected type; otherwise, it throws an exception

  - Using Components with Known Vulnerabilities

        - Regularly check the versions of both client-side and server-side components and their dependencies
        - Continuously monitor sources such as the National Vulnerability Database (NVB) for vulnerabilities in your components
        - Apply security patches regularly
        - Scan the components with security scanners frequently
        - Enforce security policies and best practices for component use
        - Review all the dependencies including transitive dependencies and ensure that they are not vulnerable
        - Maintain a regular inventory of the versions of both client-side and server-side components regularly
        - Make sure to obtain components from official sources and accept only signed packages

    - Insufficient Logging and Monitoring

        - Define the scope of assets covered in log monitoring to include business critical areas
        - Setup a minimum baseline for logging and ensure that it is followed for all assets
        - Ensure that logs are logged with user context so that they are traceable for specific users
        - Ascertain what to log and what log to look for through proactive incident identification
        - Perform sanitization on all event data to prevent log injection attacks
        - Implement a common logging mechanism for the whole application and use effective incident response
        - Ensure all logins, access control failures, and input validation failures can be logged with the necessary user context to identify suspicious accounts
        - Make sure that high-value transactions consist of an audit trail with integrity controls to prevent tampering of the databases such as append-only database tables

    - Web Application Security Testing Tools

        - N-Stalker Web App Security Scanner https://www.nstalker.com
            - N-Stalker web app security scanner checks for vulnerabilities such as SQL injection, XSS, and other known attacks
        - Acunetix WVS (https://www.acunetix.com)
        - Browser Exploitation Framework (BeEF) (http://beefproject.com)
        - Metasploit (https://www.metasploit.com)
        - PowerSploit (https://github.com)
        - Watcher (https://www.casaba.com)

- SQL Injection Attacks

    - SQL injection is a technique used to take advantage of un-sanitized input vulnerabilities to pass SQL commands through a web application for execution by a backend database

    - It is a basic attack used to either gain unauthorized access to a database or retrieve information directly from the database

    - Why Bother about SQL Injection?

        - Based on the use of applications and the way they process user supplied data, SQL injections can be used to implement the following types of attacks:

        - Authentication Bypass: Using this attack, an attacker logs onto an application without providing a valid username and password, and gains administrative privileges.
        - Authorization Bypass: Using this attack, an attacker alters authorization information stored in the database by exploiting an SQL injection vulnerability.
        - Information Disclosure: Using this attack, an attacker obtains sensitive information that is stored in the database.
        - Compromised Data Integrity: Using this attack, an attacker defaces a web page, inserts malicious content into web pages, or alters the contents of a database.
        - Compromised Availability of Data: Using this attack, an attacker deletes the database information, delete logs, or audit information stored in a database.
        - Remote Code Execution: Using this attack, an attacker compromises the host OS.

  - SQL Injection and Server-side Technologies

    - Server-side Technology
      - Powerful server-side technologies like ASP.NET and database servers allow developers to create dynamic, data-driven websites, and web apps with incredible ease
    - Exploit
      - The power of ASP.NET and SQL can easily be exploited by hackers using SQL injection attacks
    - Susceptible Databases
      - All relational databases, SQL Server, Oracle, IBM DB2, and MySQL, are susceptible to SQL-injection attacks
    - Attack
      - SQL injection attacks do not exploit a specific software vulnerability, instead they target websites and web apps that do not follow secure coding practices for accessing and manipulating data stored in a relational database

  - Types of SQL Injection

    - There are three main types of SQL injection:
      - In-band SQL Injection
      - Blind/Inferential SQL Injection
      - Out-of-Band SQL Injection

  - In-Band SQL Injection

    - Attackers use the same communication channel to perform the attack and retrieve the results
    - Types of in-band SQL Injection
      - Error-based SQL Injection: Attackers intentionally insert bad input into an application, thereby causing it to throw database errors
      - Illegal/Logically Incorrect Query: Attackers send an incorrect query to the database intentionally to generate an error message that may be helpful in performing further attacks

            - Union SQL Injection: Attackers use a UNION clause to add a malicious query to the requested query
            - System Stored Procedure: Attackers exploit databases' stored procedures to perpetrate their attacks
            - Tautology: Attackers inject statements that are always true so that the queries always return results after evaluating the WHERE condition SELECT * FROM users WHERE name = ‘’ OR ‘1’=‘1’;
            - End of Line Comment: After injecting the code into a specific field, legitimate code that follows is nullified using end of line comments SELECT * FROM user WHERE name = 'x' AND userid IS NULL; --';
            - In-line Comments: Attackers integrate multiple vulnerable inputs into a single query using inline comments INSERT INTO Users (UserName, isAdmin, Password) VALUES(‘Attacker', 1, /*’, 0, ’*/’mypwd’)
            - Piggybacked Query: Attackers inject additional malicious query into the original query. Consequently, the DBMS executes multiple SQL queries SELECT * FROM EMP WHERE EMP.EID = 1001 AND EMP.ENAME = ’Bob’; DROP TABLE DEPT;

    - Error Based SQL Injection

        - Error based SQL Injection forces the database to perform some operation in which the result will be an error
        - This exploitation may differ depending on the DBMS

    - Union SQL Injection

        - This technique involves joining a forged query to the original query
        - The result of a forged query will be joined to the result of the original query, thereby allowing it to obtain the values of fields of other tables

    - Blind/Inferential SQL Injection

        - No Error Message
            - Blind SQL Injection is used when a web application is vulnerable to an SQL injection, but the results of the injection are not visible to the attacker
        - Generic Page
            - Blind SQL injection is identical to a normal SQL Injection, except that a generic custom page is displayed when an attacker attempts to exploit an application rather than seeing a useful error message
        - Time-intensive
            - This type of attack can become time intensive because a new statement must be crafted for each bit recovered

    - Blind SQL Injection: No Error Message Returned

    - Blind SQL Injection: WAITFOR DELAY (YES or NO Response)

    - Blind SQL Injection: Boolean Exploitation

        - Multiple valid statements that evaluate true and false are supplied in the affected parameter in the HTTP request

        - By comparing the response page between both conditions, the attackers can infer whether or not the injection was successful

    - Blind SQL Injection: Heavy Query

        - Attackers use heavy queries to perform a time delay SQL injection attack without using time delay functions
        - A heavy query retrieves a significant amount of data and takes a long time to execute in the database engine
        - Attackers generate heavy queries using multiple joins on system tables

    - Out-of-Band SQL injection

        - In Out-of-Band SQL injection, the attacker needs to communicate with the server and acquire features of the database server used by the web application
        - Attackers use different communication channels to perform the attack and obtain the results
        - Attackers use DNS and HTTP requests to retrieve data from the database server
        - For example, in a Microsoft SQL Server, an attacker exploits the xp_dirtree command to send DNS requests to a server controlled by the attacker

    - SQL Injection Tools

        - sqlmap http://sqlmap.org
            - sqlmap automates the process of detecting and exploiting SQL injection flaws and the taking over of database servers
        - Mole (https://sourceforge.net)
        - Blisqy (https://github.com)
        - blind-sql-bitshifting (https://github.com)
        - NoSQLMap (https://github.com)
        - SQL Power Injector (http://www.sqlpowerinjector.com)

- SQL Injection Attack Countermeasures

    - Make no assumptions about the size, type, or content of the data that is received by your application

    - Test the size and data type of the input and enforce appropriate limits to prevent buffer overruns

    - Test the content of string variables and accept only expected values

    - Reject entries that contain binary data, escape sequences, and comment characters

    - Never build Transact-SQL statements directly from user input and use stored procedures to validate user input

    - Implement multiple layers of validation and never concatenate user input that is not validated

    - Avoid constructing dynamic SQL with concatenated input values

    - Ensure that the web config files for each application do not contain sensitive information

  - Use the most restrictive SQL account types for applications
  - Use network, host, and application intrusion detection systems to monitor injection attacks
  - Perform automated black box injection testing, static source code analysis, and manual penetration testing to probe for vulnerabilities
  - Keep untrusted data separate from commands and queries
  - In the absence of parameterized API, use specific escape syntax for the interpreter to eliminate special characters
  - Use a secure hash algorithm such as SHA256 to store user passwords rather than plaintext
  - Use the data access abstraction layer to enforce secure data access across an entire application
  - Ensure that the code tracing and debug messages are removed prior to deploying an application
  - Design the code such that it traps and handles exceptions appropriately
  - Apply least privilege rules to run the applications that access the DBMS
  - Validate user-supplied data as well as data obtained from untrusted sources on the server side
  - Avoid quoted/delimited identifiers as they significantly complicate all whitelisting, blacklisting, and escaping efforts
  - Use a prepared statement to create a parameterized query to block the execution of the query
  - Ensure that all user inputs are sanitized before using them in dynamic SQL statements
  - Use regular expressions and stored procedures to detect potentially harmful code
  - SQL Injection Detection Tools
    - Damn Small SQLi Scanner (DSSS) https://github.com
      - DSSS is an SQL injection vulnerability scanner that scans the web application for various SQL injection vulnerabilities
    - OWASP ZAP (https://www.owasp.org)
    - Snort (https://www.snort.org)
    - Burp Suite (https://portswigger.net)
    - HCL AppScan (https://www.hcltech.com)
    - w3af (https://w3af.org)

Module 08: Wireless Attacks and Countermeasures

- Wireless Terminology
  - GSM: A universal system used for mobile transportation for wireless networks worldwide
  - Bandwidth: Describes the amount of information that may be broadcast over a connection
  - Access point (AP): Used to connect wireless devices to a wireless/wired network

- BSSID: The MAC address of an AP that has set up a Basic Service Set (BSS)
- ISM band: A set of frequencies for the international industrial, scientific, and medical communities
- Hotspot: A place where a wireless network is available for public use
- Association: The process of connecting a wireless device to an AP
- SSID: A unique identifier of 32 alphanumeric characters given to a wireless local area network (WLAN)
- OFDM: Method of encoding digital data on multiple carrier frequencies
- MIMO-OFDM: An air interface for 4G and 5G broadband wireless communications
- DSSS: An original data signal multiplied with a pseudo-random noise spreading the code
- FHSS: A method of transmitting radio signals by rapidly switching a carrier among many frequency channels
- Wireless Networks
  - Wireless network (Wi-Fi) refers to WLANs based on IEEE 802.11 standard, which allows the device to access the network from anywhere within an AP range
  - Devices, such as a personal computer, video-game console, and smartphone, use Wi-Fi to connect to a network resource, such as the Internet, via a wireless network AP
- Advantages and disadvantages of wireless networks:
  - Advantages
    - Installation is fast and easy without the need for wiring through walls and ceilings
    - Easily provides connectivity in areas where it is difficult to lay cables
    - The network can be accessed from anywhere within the range of an AP
    - Public spaces such as airports, libraries, schools, and even coffee shops offer constant Internet connections through WLANs
  - Disadvantages
    - Security may not meet expectations
    - The bandwidth suffers as the number of devices in the network increases
    - Wi-Fi upgrades may require new wireless cards and/or APs
    - Some electronic equipment can interfere with Wi-Fi networks
- Types of Wireless Networks
  - Extension to a Wired Network
  - Multiple Access Points
  - LAN-to-LAN Wireless Network
  - 3G/4G Hotspot
- Wireless Standards

- 802.11: The 802.11 (Wi-Fi) standard applies to WLANs and uses FHSS or DSSS as the frequency-hopping spectrum. It allows an electronic device to establish a wireless connection in any network.
- 802.11a: It is the first amendment to the original 802.11 standard. The 802.11 standard operates in the 5 GHz frequency band and supports bandwidths up to 54 Mbps using orthogonal frequency-division multiplexing (OFDM). It has a high maximum speed but is relatively more sensitive to walls and other obstacles.
- 802.11b: IEEE extended the 802.11 standard by creating the 802.11b specifications in 1999. This standard operates in the 2.4 GHz ISM band and supports bandwidths up to 11 Mbps using direct-sequence spread spectrum (DSSS) modulation.
- 802.11d: The 802.11d standard is an enhanced version of 802.11a and 802.11b that supports regulatory domains. The specifications of this standard can be set in the media access control (MAC) layer.
- IEEE 802.11e: It is used for real-time applications such as voice, VoIP, and video. To ensure that these time-sensitive applications have the network resources they need, 802.11e defines mechanisms to ensure quality of service (QoS) to Layer 2 of the reference model, which is the MAC layer.
- 802.11g: It is an extension of 802.11 and supports a maximum bandwidth of 54 Mbps using OFDM technology. It uses the same 2.4 GHz band as 802.11b. The IEEE 802.11g standard defines high-speed extensions to 802.11b and is compatible with the 802.11b standard, which means 802.11b devices can work directly with an 802.11g AP.
- 802.11i: The IEEE 802.11i standard improves WLAN security by implementing new encryption protocols such as the Temporal Key Integrity Protocol (TKIP) and Advanced Encryption Standard (AES).
- 802.11n: The IEEE 802.11n is a revision that enhances the 802.11g standard with multiple-input multiple-output (MIMO) antennas. It works in both the 2.4 GHz and 5 GHz bands. Furthermore, it is an IEEE industry standard for Wi-Fi wireless local network transportation. Digital Audio Broadcasting (DAB) and WLAN use OFDM.
- 802.11ah: Also called Wi-Fi HaLow, uses 900 MHz bands for extended-range Wi-Fi networks and supports Internet of Things (IoT) communication with higher data rates and wider coverage range than the previous standards.
- 802.11ac: It provides a high-throughput network at a frequency of 5 GHz. It is faster and more reliable than the 802.11n standard. Moreover, it involves Gigabit networking, which provides an instantaneous data-transfer experience.
- 802.11ad: The 802.11ad standard includes a new physical layer for 802.11 networks and works on the 60 GHz spectrum. The data propagation speed in this standard is much higher from those of standards operating on the 2.4 GHz and 5 GHz bands, such as 802.11n.
- 802.12: Media utilization is dominated by this standard because it works on the demand priority protocol. The Ethernet speed with this standard is 100 Mbps. Furthermore, it is compatible with the 802.3 and 802.5 standards. Users currently on those standards can directly upgrade to the 802.12 standard.
- 802.15: It defines the standards for a wireless personal area network (WPAN) and describes the specifications for wireless connectivity with fixed or portable devices.
- 802.15.1 (Bluetooth): Bluetooth is mainly used for exchanging data over short distances on fixed or mobile devices. This standard works on the 2.4 GHz band.

    - 802.15.4 (ZigBee): The 802.15.4 standard has a low data rate and complexity. The specification used in this standard is ZigBee, transmits long-distance data through a mesh network. The specification handles applications with a low data rate of 250 Kbps, but its use increases battery life.
    - 802.15.5: This standard deploys itself on a full-mesh or half-mesh topology. It includes network initialization, addressing, and unicasting.
    - 802.16: The IEEE 802.16 standard is a wireless communications standard designed to provide multiple physical layer (PHY) and MAC options. It is also known as WiMax. This standard is a specification for fixed broadband wireless metropolitan access networks (MANs) that use a point-to-multipoint architecture.

- Wireless Encryption

  - Types of Wireless Encryption

    - 802.11i: An IEEE amendment that specifies security mechanisms for 802.11 wireless networks
    - WEP: An encryption algorithm for IEEE 802.11 wireless networks
    - EAP: Supports multiple authentication methods, such as token cards, Kerberos, and certificates
    - LEAP: A proprietary version of EAP developed by Cisco
    - WPA: An advanced wireless encryption protocol using TKIP and MIC to provide stronger encryption and authentication
    - TKIP: A security protocol used in WPA as a replacement for WEP
    - WPA2 Enterprise: Integrates EAP standards with WPA2 encryption
    - CCMP: An encryption protocol used in WPA2 for stronger encryption and authentication
    - AES: A symmetric-key encryption, used in WPA2 as a replacement for TKIP
    - WPA2: An upgrade to WPA using AES and CCMP for wireless data encryption
    - RADIUS: A centralized authentication and authorization management system
    - PEAP: A protocol that encapsulates the EAP within an encrypted and authenticated transport layer security (TLS) tunnel
    - WPA3: A third-generation Wi-Fi security protocol that uses GCMP-256 for encryption and HMAC-SHA-384 for authentication

  - Wired Equivalent Privacy (WEP) Encryption

    - WEP is a security protocol defined by the 802.11b standard; it was designed to provide a wireless LAN with a level of security and privacy comparable to that of a wired LAN
    - WEP uses a 24-bit initialization vector to form stream cipher RC4 for confidentiality and the CRC-32 checksum for integrity of wireless transmissions
    - It has significant vulnerabilities and design flaws and can therefore be easily cracked

  - Wi-Fi Protected Access (WPA) Encryption

    - WPA is a security protocol defined by 802.11i standards; it uses a Temporal Key Integrity Protocol (TKIP) that utilizes the RC4 stream cipher encryption with 128-bit keys and 64-bit MIC integrity check to provide stronger encryption and authentication
    - WPA uses TKIP to eliminate the weaknesses of WEP by including per-packet mixing functions, message integrity checks, extended initialization vectors, and re-keying

mechanisms

- WPA2 Encryption
    - WPA2 is an upgrade to WPA, and it includes mandatory support for counter mode with cipher block chaining message authentication code protocol (CCMP), an AES-based encryption mode with strong security
    - Modes of Operation
        - WPA2-Personal: It uses a set-up password (pre-shared Key, PSK) to protect unauthorized network accesses
        - WPA2-Enterprise: It includes EAP or RADIUS for centralized client authentication using multiple authentication methods, such as token cards, and Kerberos
- WPA3 Encryption
    - WPA3 is an advanced implementation of WPA2 providing trailblazing protocols and uses the AESGCMP 256 encryption algorithm
    - Modes of Operation
        - WPA3-Personal
            - It is mainly used to deliver passwordbased authentication using the SAE protocol, also known as Dragonfly Key Exchange
            - It is resistant to offline dictionary attacks and key recovery attacks
        - WPA3-Enterprise
            - It protects sensitive data using many cryptographic algorithms
            - It provides authenticated encryption using GCMP-256
            - It uses HMAC-SHA-384 to generate cryptographic keys
            - It uses ECDSA-384 for exchanging keys
- Comparison of WEP, WPA, WPA2, and WPA3

- Wireless Network-Specific Attack Techniques
    - Rogue AP Attack
        - A rogue wireless AP placed into an 802.11 network can be used to hijack the connections of legitimate network users
        - When the user turns on the computer, the rogue wireless AP will offer to connect with the network user’s NIC
        - All the traffic the user enters will pass through the rogue AP, thus enabling a form of wireless packet sniffing
    - Client Mis-Association
        - Attacker sets up a rogue AP outside the corporate perimeter and lures the employees of the organization to connect with it
        - Once associated, attackers may bypass the enterprise security policies
    - Misconfigured AP Attack
        - SSID Broadcast: APs are configured to broadcast SSIDs to authorized users

        - Weak Password: To verify authorized users, network administrators incorrectly use the SSIDs as passwords
        - Configuration Error: SSID broadcasting is a configuration error that enables intruders to steal an SSID and cause the AP assume they are allowed to connect

    - Unauthorized Association

        - Soft APs are client cards or embedded WLAN radios in some PDAs and laptops that can be launched inadvertently or through a virus program
        - Attackers infect a victim's machine and activate soft APs, thus allowing them unauthorized connection to the enterprise network
        - Attackers connect to enterprise networks through soft APs instead of the actual APs

    - Ad-Hoc Connection Attack

        - Wi-Fi clients communicate directly via an ad hoc mode that does not require an AP to relay packets
        - An ad hoc mode is inherently insecure and does not provide strong authentication and encryption
        - Thus, attackers can easily connect to and compromise the enterprise client operating in ad hoc mode

    - Honeypot AP Attack

    - AP MAC Spoofing

        - Hacker spoofs the MAC address of WLAN client equipment to mask as an authorized client
        - Attacker connects to AP as an authorized client and eavesdrops on sensitive information

    - Key Reinstallation Attack (KRACK)

        - All secure Wi-Fi networks use the 4-way handshake process to join the network and generate a fresh encryption key that will be used to encrypt the network traffic
        - The KRACK attack works by exploiting the 4-way handshake of the WPA2 protocol by forcing Nonce reuse
        - KRACK works against all modern protected Wi-Fi networks and allows attackers to steal sensitive information, such as credit card numbers, passwords, chat messages, emails, and photos

    - Jamming Signal Attack

        - All wireless networks are prone to jamming
        - This jamming signal causes a DoS because 802.11 is a CSMA/CA protocol whose collision avoidance algorithms require a period of silence before a radio is allowed to transmit
        - An attacker stakes out the area from a nearby location with a high-gain amplifier drowning out the legitimate AP

    - Wi-Fi Jamming Devices

        - Examples for Wi-Fi jamming devices: http://www.techwisetech.com
            - CPB-3016N-E5G Jammer

        - PCB-2040 Jammer
        - CPB-2060B Jammer
        - CPB-2660H-A4G Jammer
        - CPB-2061 Jammer
        - CPB-2680H-AGP Jammer

  - Cracking WEP Using Aircrack-ng

    - Run airmon-ng in monitor mode
    - Start airodump to discover SSIDs on interface and keep it running; your capture file should contain more than 50,000 IVs to successfully crack the WEP key
    - Associate your wireless card with the target AP
    - Inject packets using aireplay-ng to generate traffic on the target AP
    - Wait for airodump-ng to capture more than 50,000 IVs; crack WEP key using aircrack-ng

  - Cracking WPA-PSK Using Aircrack-ng

    - Monitor wireless traffic with airmon-ng C:>airmon-ng start eth1
    - Collect wireless traffic data with airodump-ng C:>airodump-ng --write capture eth1
    - Deauth the client using Aireplay-ng; the client will try to authenticate with the AP, which will lead to airodump capturing an authentication packet (WPA handshake)
    - Run the capture file through aircrack-ng

  - Wireless Attack Tools

    - Aircrack-ng Suite: http://www.aircrack-ng.org Aircrack-ng is a network software suite consisting of a detector, packet sniffer, WEP and WPA/WPA2-PSK cracker, and an analysis tool for 802.11 wireless networks; the program runs in Linux and Windows
      - Airbase-ng: It captures the WPA/WPA2 handshake and can act as an ad-hoc AP.
      - Aircrack-ng: This program is the de facto WEP and WPA/WPA2 PSK cracking tool.
      - Airdecap-ng: It decrypts WEP/WPA/ WPA2 and can be used to strip wireless headers from Wi-Fi packets.
      - Airdecloak-ng: It removes WEP cloaking from a pcap file.
      - Airdrop-ng: This program is used for the targeted, rule-based de-authentication of users.
      - Aireplay-ng: It is used for traffic generation, fake authentication, packet replay, and ARP request injection.
      - Airgraph-ng: This program creates a client–AP relationship and common probe graph from an airodump file.
      - Airmon-ng: It is used to switch from the managed mode to the monitor mode on wireless interfaces and vice versa.
      - Airodump-ng: This program is used to capture packets of raw 802.11 frames and collect WEP IVs.
      - Airolib-ng: This program stores and manages ESSID and password lists used in WPA/ WPA2 cracking.
      - Airserv-ng: It allows multiple programs to independently use a Wi-Fi card via a client–server TCP connection.

        - Airtun-ng: It creates a virtual tunnel interface to monitor encrypted traffic and inject arbitrary traffic into a network.
        - Easside-ng: This program allows the user to communicate via a WEP-encrypted AP without knowing the WEP key.
        - Packetforge-ng: Attackers can use this program to create encrypted packets that can subsequently be used for injection.
        - Tkiptun-ng: It injects frames into a WPA TKIP network with QoS and can recover MIC keys and keystreams from Wi-Fi traffic.
        - Wesside-ng: This program incorporates various techniques to seamlessly obtain a WEP key in minutes.
        - WZCook: It is used to recover WEP keys from the Wireless Zero Configuration utility of Windows XP.
    - AirMagnet WiFi Analyzer PRO https://www.netally.com It is used to perform reliable Wi-Fi analysis of 802.11a/b/g/n/ax wireless networks without missing any traffic
    - Ettercap https://www.ettercap-project.org
    - Wifiphisher https://wifiphisher.org
    - Reaver https://github.com
    - Fern Wifi Cracker https://github.com
    - Elcomsoft Wireless Security Auditor https://www.elcomsoft.com

- Bluetooth Attacks

  - Bluetooth Stack

    - Bluetooth is a shortrange wireless communication technology that replaces the cables connecting portable or fixed devices while maintaining high levels of security
    - It allows devices to share data over short distances

  - Bluetooth Modes

    - Discoverable modes
      - Discoverable: Sends inquiry responses to all inquiries
      - Limited discoverable: Visible for a certain period of time
      - Non-discoverable: Never answers an inquiry scan
    - Pairing modes
      - Non-pairable mode: Rejects every pairing request
      - Pairable mode: Pairs upon request

  - Bluetooth Hacking

    - Bluetooth hacking refers to the exploitation of Bluetooth stack implementation vulnerabilities to compromise sensitive data in Bluetooth-enabled devices and networks

  - Bluetooth Attacks

    - Bluesmacking: DoS attack, which overflows Bluetooth enabled devices with random packets, causes the devices to crash
    - Bluejacking: The art of sending unsolicited messages over Bluetooth to Bluetooth-enabled devices, such as mobile phones and laptops

        - Bluesnarfing: The theft of information from a wireless device through a Bluetooth connection
        - BlueSniff: Proof of concept code for a Bluetooth wardriving utility
        - Bluebugging: Remotely accessing a Bluetoothenabled device and using its features
        - BluePrinting: The art of collecting information about Bluetooth-enabled devices, such as manufacturer, device model, and firmware version
        - Btlejacking: Detrimental to BLE devices, it is used to bypass security mechanisms and listen to information being shared
        - KNOB Attack: Exploiting a vulnerability in Bluetooth to eavesdrop all the data being shared, such as keystrokes, chats, and documents
        - MAC Spoofing Attack: Intercepting data intended for other Bluetooth-enabled devices
        - Man-in-the-Middle /Impersonation Attack: Modifying data between Bluetoothenabled devices communicating in a Piconet

    - Bluetooth Threats

        - Leakage of calendars and address books: Attackers can steal a user’s personal information and use it for malicious purposes.
        - Bugging devices: Attackers can instruct a smartphone to make a call to other phones without any user interaction. They can even record a user’s conversations.
        - Sending SMS messages: Terrorists could send false bomb threats to airlines using the smartphones of legitimate users.
        - Causing financial losses: Hackers can send many MMS messages with an international user's phone, resulting in a high phone bill.
        - Remote control: Hackers can remotely control a smartphone to make phone calls or connect to the Internet.
        - Social engineering: Attackers can trick Bluetooth users into lowering security or disabling authentication for Bluetooth connections to pair with them and steal their information.
        - Malicious code: Smartphone worms can exploit a Bluetooth connection to replicate and spread itself.
        - Protocol vulnerabilities: Attackers exploit Bluetooth pairings and communication protocols to steal data, make calls, send messages, launch DoS attacks on a device, spy on phones, etc.

    - Bluetooth Attack Tools

        - BluetoothView https://www.nirsoft.net It monitors the activity of Bluetooth devices around you and displays information, such as Device Name, Bluetooth Address, Major Device Type, Minor Device Type, First Detection Time, and Last Detection Time
        - BlueZ (http://www.bluez.org)
        - BtleJack (https://github.com)
        - BTCrawler (http://petronius.sourceforge.net)
        - BlueScan (http://bluescanner.sourceforge.net)
        - Bluetooth Scanner – btCrawler (https://play.google.com)

- Wireless Attack Countermeasures

    - Best Practices for Configuration

    - Change the default SSID after WLAN configuration.
    - Set the router access password and enable firewall protection.
    - Disable SSID broadcasts.
    - Disable remote router login and wireless administration.
    - Enable MAC address filtering on APs or routers.
    - Enable encryption on APs and change passphrases often.
    - Close all unused ports to prevent attacks on Aps.

  - Best Practices for SSID Settings

    - Use SSID cloaking to keep certain default wireless messages from broadcasting the SSID to everyone.
    - Do not use the SSID, company name, network name, or any easy-to-guess string in passphrases.
    - Place a firewall or packet filter between an AP and the corporate Intranet.
    - Limit the strength of the wireless network so that it cannot be detected outside the bounds of the organization.
    - Check the wireless devices for configuration or setup problems regularly.
    - Implement an additional technique for encrypting traffic, such as IPSec over wireless.

  - Best Practices for Authentication

    - Choose WPA2-Enterprise with 802.1x authentication instead of WPA or WEP.
    - Implement WPA2/WPA3-Enterprise wherever possible.
    - Disable the network when not required.
    - Place wireless APs in a secured location.
    - Keep drivers on all wireless equipment updated.
    - Use a centralized server for authentication.
    - Enable server verification on the client side using 802.1X authentication to prevent MITM attacks.
    - Enable two-factor authentication as an added line of defense.
    - Deploy rogue-AP detection or wireless intrusion prevention/detection systems to prevent wireless attacks.

  - Bluetooth Attack Countermeasures

    - Use non-regular patterns as PINs while pairing a device. Key combinations should not be sequential on the keypad.
    - Keep the device in the non-discoverable (hidden) mode.
    - Do not accept any unknown or unexpected request for pairing.
    - Regularly check of all devices paired in the past and delete any suspicious paired device.
    - Keep Bluetooth in the disabled state and enable it only when needed. Disable Bluetooth immediately after the intended task is completed.
    - Always enable encryption when establishing a Bluetooth connection.
    - Set the network range of a Bluetooth-enabled device to the lowest and perform pairing only in a secure area.
    - Install antivirus software that supports host-based security software on Bluetoothenabled devices.

    - Change the default settings of the Bluetooth-enabled device to the best security standard.
    - Use link encryption for all Bluetooth connections.
    - If multiple wireless communications are being used, ensure that encryption is empowered on each link in the communication chain.
    - Avoid sharing sensitive information over Bluetooth-enabled devices.
    - Disable automatic connections to public Wi-Fi networks for protecting Bluetooth devices from unsecured sources.
    - Update the software and drivers of the Bluetooth devices and regularly change the passwords.
    - Use a VPN for secure connections between Bluetooth devices.

  - Wireless Security Tools

    - Cisco Adaptive Wireless IPS https://www.cisco.com Adaptive wireless IPS (WIPS) provides wireless-network threat detection and mitigation against malicious attacks and security vulnerabilities
    - AirMagnet WiFi Analyzer PRO (https://www.netally.com)
    - RFProtect (https://www.arubanetworks.com)
    - WatchGuard WIPS (https://www.watchguard.com)
    - AirMagnet Planner (https://www.netally.com)
    - Extreme AirDefense (https://www.extremenetworks.com)

Module 09: Mobile Attacks and Countermeasures

- Mobile Attack Anatomy

  - Vulnerable Areas in Mobile Business Environment https://www.ibm.com

    - Smartphones offer broad Internet and network connectivity via different channels, such as 3G/4G/5G, Bluetooth, Wi-Fi, and wired computer connections
    - Security threats may arise in different places along these channels during data transmission

  - OWASP Top 10 Mobile Risks https://www.owasp.org

    - Improper Platform Usage
    - Insecure Data Storage
    - Insecure Communication
    - Insecure Authentication
    - Insufficient Cryptography
    - Insecure Authorization
    - Client Code Quality
    - Code Tampering
    - Reverse Engineering
    - Extraneous Functionality

  - Anatomy of a Mobile Attack https://www.nowsecure.com

    - The Device

  - Browser-based Attacks
    - Phishing
    - Framing
    - Clickjacking
    - Man-in-the-Mobile
    - Buffer Overflow
    - Data Caching
  - Phone/SMS-based Attacks
    - Baseband Attacks
    - SMiShing
  - Application-based Attacks
    - Sensitive Data Storage
    - No Encryption/Weak Encryption
    - Improper SSL Validation
    - Configuration Manipulation
    - Dynamic Runtime Injection
    - Unintended Permissions
    - Escalated Privileges
  - The System
    - No Passcode/Weak Passcode
    - iOS Jailbreaking
    - Android Rooting
    - OS Data Caching
    - Passwords and Data Accessible
    - Carrier-loaded Software
    - User-initiated Code
- The Network
  - Wi-Fi (weak encryption/no encryption)
  - Rogue Access Points
  - Packet Sniffing
  - Man-in-the-Middle (MITM)
  - Session Hijacking
  - DNS Poisoning
  - SSLStrip
  - Fake SSL Certificates
- The Data Center/CLOUD
  - Web-server-based attacks
    - Platform Vulnerabilities
    - Server Misconfiguration
    - Cross-site Scripting (XSS)
    - Cross-Site Request Forgery (CSRF)
    - Weak Input Validation
    - Brute-Force Attacks
  - Database Attacks
    - SQL injection
    - Privilege Escalation

            - Data Dumping
            - OS Command Execution

    - How a Hacker can Profit from Mobile Devices that are Successfully Compromised https://www.sophos.com, https://securelist.com

- Mobile Platform Attack Vectors and Vulnerabilities

    - Mobile Attack Vectors

        - Malware
            - Virus and rootkit
            - Application modification
            - OS modification
        - Data Tampering
            - Modification by another application
            - Undetected tamper attempts
            - Jailbroken device
        - Data Exfiltration
            - Extracted from data streams and email
            - Print screen and screen scraping
            - Copy to USB key and loss of backup
        - Data Loss
            - Application vulnerabilities
            - Unapproved physical access
            - Loss of device

    - Mobile Platform Vulnerabilities and Risks

        - Malicious apps in stores
        - Mobile malware
        - App sandboxing vulnerabilities
        - Weak device and app encryption
        - OS and app update issues
        - Jailbreaking and rooting
        - Mobile application vulnerabilities
        - Privacy issues (Geolocation)
        - Weak data security
        - Excessive permissions
        - Weak communication security
        - Physical attacks
        - Insufficient code obfuscation
        - Insufficient transport layer security
        - Insufficient session expiration

    - Security Issues Arising from App Stores

        - Insufficient or no vetting of apps leads to malicious and fake apps entering the app marketplace

    - App stores are common target for attackers to distribute malware and malicious apps
    - Malicious apps can damage other applications and data, and send your sensitive data to attackers

  - App Sandboxing Issues

    - Sandboxing helps protect systems and users by limiting the resources the app can access to the mobile platform; however, malicious applications may exploit vulnerabilities and bypass the sandbox

  - Mobile Spam

    - Unsolicited text/email messages sent to mobile devices from known/unknown phone number and email IDs
    - Spam messages contain advertisements or malicious links that can trick users into revealing confidential information
    - Significant amount of bandwidth is wasted by spam messages
    - Spam attacks are performed for financial gain

  - SMS Phishing Attack (SMiShing) (Targeted Attack Scan)

    - SMS Phishing is the act of trying to acquire personal and financial information by sending SMSs (Instant Messages or IMs) containing deceptive links

  - Why is SMS Phishing Effective?

    - Most consumers access the Internet through a mobile device.
    - Easy to set up a mobile phishing campaign.
    - Difficult to detect and stop it causes harm.
    - Mobile users are not conditioned to receiving spam text messages on their mobile devices.
    - No mainstream mechanism for weeding out spam SMS.
    - Most mobile anti-virus tools do not check SMS.

  - SMS Phishing Attack Examples

  - Pairing Mobile Devices on Open Bluetooth and Wi-Fi Connections

    - Mobile device pairing on open connections (public Wi-Fi/unencrypted Wi-Fi routers) allows attackers to eavesdrop and intercept data transmission using techniques such as;
      - Bluesnarfing (stealing information via Bluetooth)
      - Bluebugging (gaining control over the device via Bluetooth)
    - Sharing data from malicious devices can infect/breach data on the recipient device

  - Agent Smith Attack

    - An Agent smith attack is carried out by persuading the victim to install a malicious app designed and published by an attacker
    - The malicious app replaces legitimate apps, such as WhatsApp, SHAREit, and MX Player
    - The attacker produces a huge volume of advertisements on the victim’s device through the infected app for financial gains

- Exploiting SS7 Vulnerability
    - Signaling System 7 (SS7) is a communication protocol that allows mobile users to exchange communication through another cellular network
    - SS7 is operated depending on mutual trust between operators without any authentication
    - Attackers can exploit this vulnerability to perform a man-in-the-middle attack, impeding the texts and calls between communicating devices
    - Threats Associated with SS7 vulnerability. When the attacker gains access to the SS7 protocol, the victim’s device faces the following risks:
        - Exposing the subscriber’s identity
        - Revealing the network identity
        - Spying on and intercepting the network to steal personal data
        - Allowing phone tapping
        - Performing DoS attacks to damage the reputation of the target telecom operator
        - Tracking geographic locations
- Simjacker: SIM Card Attack
    - Simjacker is a vulnerability associated with a SIM card’s S@T browser, a pre-installed software on SIM cards that is designed to provide a set of instructions
    - Attackers exploit Simjacker to perform various malicious activities, such as capturing the locations of devices, monitoring calls, forcing device browsers to connect to malicious websites, and performing DoS attacks
    - Steps involved in Simjacker attack
        - The attacker sends fraudulent SMS containing hidden code or instructions from a SIM Application Toolkit (STK)
        - The victim receives the malicious SMS and the S@T browser on the SIM card automatically recognizes and processes the hidden instructions or code
        - The injected code performs various activities on the device without the user’s consent
        - The accomplice device receives the user information via SMS, which an attacker can use to track live locations, exfiltrate the device information, and perform many other malicious activities
- Hacking an Android Device Using Metasploit https://www.metasploit.com
    - Attackers use various tools such as Metasploit to create binary payloads, which are sent to the target Android device to gain control over it
- Android Hacking Tools
    - zANTI https://www.zimperium.com An Android app that allows you to perform attacks, such as spoof MAC address, creating a malicious Wi-Fi hotspot, and hijack session
    - Network Spoofer (https://www.digitalsquid.co.uk)
    - Low Orbit Ion Cannon (LOIC) (https://droidinformer.org)
    - DroidSheep (https://droidsheep.info)
    - Orbot Proxy (https://guardianproject.info)
    - PhoneSploit (https://github.com)

- iOS Hacking Tools
    - Elcomsoft Phone Breaker https://www.elcomsoft.com Allows attackers to perform logical and over-the-air acquisition of iOS devices, break into encrypted backups, and obtain and analyze backups, synchronized data, and passwords from Apple iCloud
    - Fing - Network Scanner (https://apps.apple.com)
    - Network Analyzer Master (https://apps.apple.com)
    - Spyic (https://spyic.com)
    - iWepPRO (https://apps.apple.com)
    - Frida (https://www.frida.re)

- Mobile Device Management (MDM) Concept

    - Mobile Device Management (MDM) provides platforms for over-the-air or wired distribution of applications and data and configuration settings for all types of mobile devices, including mobile phones, smartphones, and tablet computers

    - It helps system administrators to deploy and manage software applications across all enterprise mobile devices to secure, monitor, manage, and support mobile devices

    - The basic features of MDM software are as follows:

        - Uses a passcode for the device
        - Remotely locks the device if it is lost
        - Remotely wipes data in the lost or stolen device
        - Detects if the device is rooted or jailbroken
        - Enforces policies and tracks inventory
        - Performs real-time monitoring and reporting

    - Bring Your Own Device (BYOD)

        - Bring your own device (BYOD) refers to a policy that allows an employee to bring their personal devices, such as laptops, smartphones, and tablets, to their workplace and use them to access the organization’s resources by following the access privileges

        - The BYOD policy allows employees to use the devices that they are comfortable with and best fits their preferences and work purposes

        - BYOD Benefits

            - Increased Productivity
            - Employee Satisfaction
            - Work Flexibility
            - Lower Costs

        - BYOD Risks

            - Sharing confidential data on unsecured networks
            - Data leakage and endpoint security issues
            - Improperly disposing of devices
            - Support for many different devices

        - Mixing personal and private data
        - Lost or stolen devices
        - Lack of awareness
        - Ability to bypass organization’s network policy rules
        - Infrastructure issues
        - Disgruntled employees

- Mobile Attack Countermeasures

    - OWASP Top 10 Mobile Controls https://www.owasp.org

    - Identify and protect sensitive data on the mobile device

        - In the design phase, classify the data storage according to the sensitivity and then apply the controls. Process, store, and use data based on its classification
        - Apply validation of the security of API calls to the sensitive data
        - Store the sensitive data on the server instead of the client-side device, as it supports secure network connectivity and other protection mechanisms
        - Use file encryption API provided by the OS or other trusted source when storing data in a device.
        - Use encryption to store sensitive data and store it in a tamper-proof area if possible
        - Restrict access to sensitive data based on contextual information, e.g., location
        - Always make sure to turn off the location, GPS tracking, or other sensitive information when not in use
        - Always be aware of the public shared storage as it is easily vulnerable to data leakage
        - Apply the principle of minimal disclosure and identify the type of data needed in the design phase
        - Use non-persistent identifiers wherever possible, which are not shared with other apps
        - Applications should use remote wipe and kill switch APIs for removing sensitive information from the device in the event of theft or loss

    - Handle password credentials securely on the device

        - Use longer term authorization tokens instead of passwords as per the OAuth model and encrypt tokens in transit using SSL/TLS
        - Leverage the encryption and key-store mechanisms provided by the mobile OS to securely store passwords and authorization tokens
        - Ensure that capabilities such as secure element are used to store keys, credentials, and other sensitive data
        - Allow access to mobile users for changing the passwords on the device
        - Make sure to use measures that allow repeated patterns to curb smudge attacks
        - Make sure that no password or key is visible in the cache or logs
        - Do not store any passwords or secrets in the mobile application binaries, as they can be easily downloaded and reverse engineered

    - Ensure sensitive data are protected in transit

        - Enforce the use of an end-to-end secure channel such as SSL/TLS when sending sensitive information over the network

    - Use complex and well-known encryption algorithms such as AES with appropriate key lengths for enhanced security
    - Ensure the use of certificates signed by trusted CA providers and do not disable or ignore SSL chain validation
    - A secure connection should be established only after verifying the identity of the remote end point for reducing the risk of MITM attacks
    - Sending sensitive data using SMS or MMS from or to the mobile end points should be avoided

  - Implement user authentication, authorization, and session management correctly

    - The authentication mechanism strength must depend on the sensitivity of the data being processed by the application and its access to valuable resources
    - Ensure that session management is handled properly after the initial authentication using appropriate secure protocols
    - Use unpredictable session identifiers with high entropy and repeated application of SHA1 for combining random variables
    - Use contexts such as IP location for adding security to authentication
    - Ensure the use of additional authentication factors for mobile applications that give access to sensitive data using voice, fingerprint, or other behavioral inputs
    - Use authentication that depends on the end-user identity rather than the device identity

  - Keep the backend APIs (services) and the platform (server) secure

    - Perform detailed code checking for sensitive data that is transferred unintentionally between the mobile device, web-server backend, and other external interfaces
    - All the backend services for the mobile apps should be tested for vulnerabilities periodically using any static code analyzer tools and fuzzing tools
    - Ensure that the backend platform is running with a hardened configuration with the latest security patches applied to the OS and web server
    - Adequate logs are reserved at the backend for detecting and responding to incidents and for performing forensics
    - Use rate limiting and throttling on a per-user/IP basis for reducing the risk of DDoS attacks
    - Ensure testing for DoS vulnerabilities that make the server flooded with resourceintensive application calls
    - Perform use case testing and abuse case testing to determine the vulnerabilities; also perform testing of the backend web services/REST

  - Secure data integration with third-party services and applications

    - Always scrutinize the authenticity of any third-party code or libraries used in the mobile application
    - Regularly update the latest security patches and keep track of all the third-party APIs and framework
    - Validate all the data received and sent before processing for non-trusted third-party applications

  - Pay specific attention to the collection and storage of consent for the collection and use of the user’s data

    - Create a privacy policy that covers the usage of personal data and make it available to users when making consent choices such as at install time or at run time or via opt-out mechanisms
    - Check if any application is collecting Personally Identifiable Information (PII)
    - Review the communication mechanisms to check for any accidental leaks
    - Always preserve the record of consent to the transfer of PII
    - Ensure that the consent collection mechanism does not overlap or conflict and try to resolve any conflicts

  - Implement controls to prevent unauthorized access to paid-for resources (wallet, SMS, phone calls, etc.)

    - Maintain access logs to paid-for resources in a non-repudiable format and make them available for end-user monitoring
    - Regularly check for any abnormal usage patterns in paid-for resource usage and activate re-authentication
    - Ensure use of the white-list model by default for addressing paid-for resources
    - Authenticate all the API calls to paid-for resources
    - Ensure that the wallet API callbacks do not permit cleartext passwords and other sensitive information
    - Caution users and obtain permission for any type of cost implications for app performance
    - Implement best practices such as low latency and caching to minimize the signaling load on the base stations

  - Ensure secure distribution/provisioning of mobile applications

    - The applications must be designed and provisioned to allow updates for security patches
    - The app stores should monitor the apps for vulnerable code and should be able to remove apps remotely at short notice in the case of an incident
    - Provide a feedback channel for the users to report security problems with the apps

  - Carefully check any runtime interpretation of code for errors

    - Minimize runtime interpretation and the capabilities offered to runtime interpreters and run interpreters with minimum privileges
    - Outline comprehensive escape syntax as appropriate
    - Use fuzz test interpreters and sandbox interpreters

  - General Guidelines for Mobile Platform Security

    - Do not load too many applications and avoid auto-upload of photos to social networks
    - Perform a security assessment of the application architecture
    - Maintain configuration control and management
    - Install applications from trusted application stores
    - Securely wipe or delete the data when disposing of the device
    - Do not share the information within GPS-enabled apps unless it is necessary
    - Never connect two separate networks such as Wi-Fi and Bluetooth simultaneously
    - Disable wireless access such as Wi-Fi and Bluetooth if not in use
      - Ensure that your Bluetooth is “off” by default. Turn it on whenever it is necessary

        - Disable wireless access such as Wi-Fi and Bluetooth if not in use to avoid illegal wireless access to the device
        - Disable sharing/tethering Internet connections over Wi-Fi and Bluetooth when not in use
    - Use Passcode
        - Configure a strong passcode with the maximum possible length to gain access to your mobile devices
        - Set an idle timeout to automatically lock the phone when not in use
        - Enable the lockout/wipe feature after a certain number of attempts
        - Consider an eight-character complex passcode
        - Thwart passcode guessing: set erase data to ON
    - Update OS and Apps
        - Update OS and apps to keep them secure
        - Apply software updates when new releases are available
        - Perform regular software maintenance
    - Enable Remote Management
        - In an enterprise environment, use MDM software to secure, monitor, manage, and support mobile devices deployed across the organization
    - Do not allow Rooting or Jailbreaking
        - Ensure that your MDM solutions prevent or detect rooting/jailbreaking
        - Include this clause in your mobile security policy
    - Use Remote Wipe Services
        - Use remote wipe services such as Find My Device (Android) and Find My iPhone or FindMyPhone (Apple iOS) to locate your device should it be lost or stolen
        - Report a lost or stolen device to IT so that they can disable certificates and other access methods associated with the device
    - Encrypt Storage
        - If supported, configure your mobile device to encrypt its storage with hardware encryption
        - Use device encryption and patch applications
        - Encrypt the device and backups
    - Perform periodic backup and synchronization
        - Use a secure, over-the-air backup-and-restore tool that performs periodic background synchronization
        - (Android) Backup to your Google account so that sensitive enterprise data are not backed up to the cloud
        - Control the location of backups
        - Encrypt backups
        - Keep sensitive data off shared mobile devices. If enterprise information is locally stored on a device, then it is recommended that this device not be openly shared
        - Limit logging data stored on the device
        - Use a secure data-transfer utility or encrypt data in transit to or from the device, to ensure confidentiality and data integrity

- Mobile Security Tools

        - Malwarebytes Security https://play.google.com An antimalware mobile tool that provides protection against malware, ransomware, and other growing threats to Android devices
        - Lookout Personal (https://www.lookout.com)
        - Zimperium’s zIPS (https://www.zimperium.com)
        - BullGuard Mobile Security (https://www.bullguard.com)
        - Norton Security for iOS (https://us.norton.com)
        - Comodo Mobile Security (https://m.comodo.com)

Module 10: IoT and OT Attacks and Countermeasures

- IoT Attacks

- IoT Concepts

    - Internet of Things (IoT), also known as Internet of Everything (IoE), refers to the network of devices having IP addresses and the capability to sense, collect, and send data using embedded sensors, communication hardware and processors

    - In IoT, the term thing is used to refer to a device that is implanted on natural, human-made, or machine-made objects and has the functionality of communicating over the network

    - How the IoT Works

        - Sensing Technology
        - IoT Gateways
        - Cloud Server/Data Storage
        - Remote Control using Mobile App

    - IoT Architecture

        - Application Layer: Delivery of various applications to different users in IoT
        - Middleware Layer: Device management and information management
        - Internet Layer: Connection between endpoints
        - Access Gateway Layer: Protocol translation and messaging
        - Edge Technology Layer: Sensors, devices, machines, and intelligent edge nodes of various types

    - IoT Application Areas and Devices http://www.beechamresearch.com

- IoT Threats and Attacks

    - Challenges of IoT

        - Lack of Security and Privacy: Most IoT devices today, such as household devices, industrial devices, healthcare devices, automobiles, etc., are connected to the Internet and contain important and confidential data. These devices lack even basic security and privacy policies, and hackers can exploit this to carry out malicious activity.
        - Vulnerable Web Interfaces: Many IoT devices come with embedded web server technology that makes them vulnerable to attacks.

    - Legal, Regulatory, and Rights Issue: Due to the interconnection of IoT devices, certain security issues are raised with no existing laws that address these issues.
    - Default, Weak, and Hardcoded Credentials: One of the most common reasons for cyber-attacks on IoT devices is their authentication systems. These devices usually come with default and weak credentials, which can easily be exploited by a hacker to gain unauthorized access to the devices.
    - Clear Text Protocols and Unnecessary Open Ports: IoT devices lack encryption techniques during the transmission of data, which at times causes them to use certain protocols that transmit data in clear text in addition to having open ports.
    - Coding Errors (Buffer Overflow): Most IoT devices today have embedded web services that are subject to the same vulnerabilities that are commonly exploited on web service platforms. As a result, updating such functionality may give rise to issues like buffer overflows, SQL injection, etc. within technology infrastructure.
    - Storage Issues: IoT devices generally come with smaller data storage capacity, but the data collected and transmitted by the devices is limitless. Therefore, this gives rise to data storage, management, and protection issues.
    - Difficult-to-Update Firmware and OS: Upgrading firmware is an essential step toward countering vulnerabilities in a device, but it may impair a device’s functionality. For this reason, developers or manufacturers may hesitate or even refuse to provide product support or make adjustments during the development phase of their products.
    - Interoperability Standard Issues: One of the biggest obstacles for IoT devices is the interoperability issue, which is key to the viability and long-term growth of the entire IoT ecosystem. The issues that arise due to lack of interoperability in IoT devices are the inability of manufacturers to test application programming interfaces (APIs) using common methods and mechanisms, their inability to secure devices using software from third parties, and their inability to manage and monitor devices using a common layer.
    - Physical Theft and Tampering: Physical attacks on IoT devices include tampering with the devices to inject malicious code or files to make the devices work the way the attacker intends, or making hardware modifications to the devices. Counterfeiting the devices may also be an issue when proper physical protection is not present to shield the devices.
    - Lack of Vendor Support for Fixing Vulnerabilities: The firmware of the devices has to be upgraded in order to protect the devices against certain vulnerabilities, but vendors are hesitant, or they usually refuse to get third-party access to their devices.
    - Emerging Economy and Development Issues: With widespread opportunities for IoT devices in every field, multiple layers of complexity are added for policymakers. The new landscape introduced by these devices adds a new dimension for the policymakers, who have to design new blueprints and policies for IoT devices.
    - Handling of Unstructured Data: An increase in the number of connected devices will increase the complexity of handling unstructured data as its volume, velocity, and variety increases. It is important for organizations to understand and determine which data is valuable and actionable.

  - IoT Security Problems

    - APPLICATION: Validation of the inputted string, AuthN, AuthZ, no automatic security updates, default passwords

    - NETWORK: Firewall, improper communications encryption, services, lack of automatic updates
    - MOBILE: Insecure API, lack of communication channels encryption, authentication, lack of storage security
    - CLOUD: Improper authentication, no encryption for storage and communications, insecure web interface
    - IoT: Application + Network + Mobile + Cloud = IoT

  - OWASP Top 10 IoT Threats https://www.owasp.org

    - Weak, Guessable, or Hardcoded Passwords
    - Insecure Network Services
    - Insecure Ecosystem Interfaces
    - Lack of Secure Update Mechanisms
    - Use of Insecure or Outdated Components
    - Insufficient Privacy Protection
    - Insecure Data Transfer and Storage
    - Lack of Device Management
    - Insecure Default Settings
    - Lack of Physical Hardening

  - IoT Threats

    - DDoS Attack: An attacker converts the devices into an army of botnets to target a specific system or server, making it unavailable to provide services.
    - Attack on HVAC Systems: HVAC system vulnerabilities are exploited by attackers to steal confidential information such as user credentials and to perform further attacks on the target network.
    - Rolling Code Attack: An attacker jams and sniffs the signal to obtain the code transferred to a vehicle's receiver; the attacker then uses it to unlock and steal the vehicle.
    - BlueBorne Attack: Attackers connect to nearby devices and exploit the vulnerabilities of the Bluetooth protocol to compromise the device.
    - Jamming Attack: An attacker jams the signal between the sender and the receiver with malicious traffic that makes the two endpoints unable to communicate with each other.
    - Remote Access using Backdoor: Attackers exploit vulnerabilities in the IoT device to turn it into a backdoor and gain access to an organization's network.
    - Remote Access using Telnet: Attackers exploit an open telnet port to obtain information that is shared between the connected devices, including their software and hardware models.
    - Sybil Attack: An attacker uses multiple forged identities to create a strong illusion of traffic congestion, affecting communication between neighboring nodes and networks.
    - Exploit Kits: A malicious script is used by the attackers to exploit poorly patched vulnerabilities in an IoT device.
    - Man-in-the-Middle Attack: An attacker pretends to be a legitimate sender who intercepts all the communication between the sender and receiver and hijacks the communication.
    - Replay Attack: Attackers intercept legitimate messages from valid communication and continuously send the intercepted message to the target device to perform a denial-ofservice attack or crash the target device.

        - Forged Malicious Device: Attackers replace authentic IoT devices with malicious devices if they have physical access to the network.
        - Side-Channel Attack: Attackers perform side-channel attacks by extracting information about encryption keys by observing the emission of signals, i.e., ”side channels”, from IoT devices.
        - Ransomware Attack: Ransomware is a type of malware that uses encryption to block a user’s access to his/her device either by locking the screen or by locking the user’s files.
        - Client Impersonation: An attacker masquerades as a legitimate smart device/server using a malicious device and compromises an IoT client device by impersonating it, to perform unauthorized activities or access sensitive information on behalf of the legitimate client.
        - SQL Injection Attack: Attackers perform SQL injection attacks by exploiting vulnerabilities in the mobile or web applications used to control the IoT devices, to gain access to the devices and perform further attacks on them.
        - SDR-Based Attack: Using a software-based radio communication system, an attacker can examine the communication signals passing through the IoT network and can send spam messages to the interconnected devices.
        - Fault Injection Attack: A fault injection attack occurs when an attacker tries to introduce fault behavior in an IoT device, with the goal of exploiting these faults to compromise the security of that device.
        - Network Pivoting: An attacker uses a malicious smart device to connect and gain access to a closed server, and then uses that connection to pivot other devices and network connections to the server to steal sensitive information.
        - DNS Rebinding Attack: DNS rebinding is a process of obtaining access to a victim’s router using a malicious JavaScript code injected on a web page.

    - Hacking IoT Devices: General Scenario

    - IoT Attacks: DDoS Attack

        - Attacker initiates the attack by exploiting the vulnerabilities in the devices and installing a malicious software in their operating systems
        - Multiple infected IoT devices are referred to as an Army of Botnets
        - The target is attacked with a large volume of requests from multiple IoT devices present in different locations

    - Exploit HVAC

        - Many organizations use Internet-connected heating, ventilation, and air conditioning (HVAC) systems without implementing security mechanisms; this gives attackers a gateway to hack corporate systems

        - HVAC systems have many security vulnerabilities that are exploited by attackers to steal login credentials, gain access to the HVAC system, and perform further attack on the organization’s network

        - Steps followed by an attacker to exploit HVAC systems:

            - Attacker uses Shodan (https://www.shodan.io) and searches for vulnerable industrial control systems (ICSs)

        - Based on the vulnerable ICSs found, the attacker then searches for default user credentials using online tools such as https://www.defpass.com
        - Attacker uses default user credentials to attempt to access the ICS
        - After gaining access to the ICS, the attacker attempts to gain access to the HVAC system remotely through the ICS
        - After gaining access to the HVAC system, an attacker can control the temperature from the HVAC or carry out other attacks on the local network

  - Rolling Code Attack

    - For example, given below are the steps followed by an attacker to perform a rolling-code attack:
        - Victim presses car remote button and tries to unlock the car
        - Attacker uses a jammer that jams the car's reception of the rolling code sent by the victim and simultaneously sniffs the first code
        - The car does not unlock; victim tries again by sending a second code
        - Attacker sniffs the second code
        - On the second attempt by the victim, the attacker forwards the first code, which unlocks the car
        - The recorded second code is used later by the attacker to unlock and steal the vehicle
        - Attackers can make use of tools such as rfcat-rolljam and RFCrack to perform this attack.

  - BlueBorne Attack

    - A BlueBorne attack is performed on Bluetooth connections to gain access and take full control of the target device

    - After gaining access to a device, the attacker can penetrate any corporate network using that device to steal critical information about the organization and spread malware to nearby devices

    - Steps to perform BlueBorne attack:

        - Attacker discovers active Bluetooth-enabled devices around him/her; all Bluetoothenabled devices can be located even if they are not in discoverable mode
        - After locating any nearby device, the attacker obtains the MAC address of the device
        - Now, the attacker sends continuous probes to the target device to determine the OS
        - After identifying the OS, the attacker exploits the vulnerabilities in the Bluetooth protocol to gain access to the target device
        - Now the attacker can perform remote code execution or a man-in-the-middle attack and take full control of the device

  - Jamming Attack

    - Jamming is a type of attack in which the communications between wireless IoT devices are jammed so that they can be compromised

    - An attacker transmits radio signals randomly with the same frequency as the sensor nodes for communication
    - As a result, the network gets jammed, which disables the endpoints from sending or receiving any messages
  - Hacking Smart Grid/Industrial Devices: Remote Access using Backdoor
  - SDR-Based Attacks on IoT
    - The attacker uses software defined radio (SDR) to examine the communication signals in the IoT network and sends spam content or texts to the interconnected devices
      - Replay Attack
        - The attacker obtains the specific frequency used for sharing information between connected devices and captures the original data when a command is initiated by these devices
        - The attacker segregates the command sequence and injects it into the IoT network
      - Cryptanalysis Attack
        - The attacker uses the same procedure as that followed in a replay attack, along with reverse engineering of the protocol to capture the original signal
        - The attacker must be skilled in cryptography, communication theory, and modulation schemes to perform this attack
      - Reconnaissance Attack
        - The attacker obtains information about the target device from the device's specifications
        - The attacker then uses a multimeter to investigate the chipset and mark some identifications such as ground pins to discover the product ID and other information
  - Fault Injection Attacks
    - Fault injection attacks, also known as Perturbation attacks, occur when a perpetrator injects any faulty or malicious program into the system to compromise the system security
    - Types of Fault Injection Attacks
      - Optical, Electro Magnetic Fault Injection (EMFI), Body Bias Injection (BBI)
        - Attackers inject faults into the device by using projecting lasers and electromagnetic pulses
      - Power/Clock/Reset Glitching
        - Attackers inject faults or glitches into the power supply and clock network of the chip
      - Frequency/Voltage Tampering
        - Attackers tamper with the operating conditions, modify the level of the power supply and/or alter the clock frequency of the chip
      - Temperature Attacks
        - Attackers alter the temperature for operating the chip, affecting the whole operating environment

    - Capturing and Analyzing IoT Traffic using Wireshark
        - Run Nmap to identify IoT devices using insecure HTTP ports nmap -p 80,81,8080,8081
        - Run ifconfig to identify your wireless card, here wlan0
        - Run Airmon-ng to put the wireless card in monitor mode airmon-ng start wlan0
        - Run Airodump-ng to scan all the nearby wireless networks airodump-ng start wlan0mon
        - Discover the target wireless network and note down the corresponding channel to sniff the traffic using Wireshark
        - Next, setup your wireless card to listen to the traffic on the same channel using Airmon-ng airmon-ng start wlan0mon 11
        - Launch Wireshark and double-click the interface that was kept in monitor mode, here wlan0mon and start capturing the traffic
    - IoT Attack Tools
        - Firmalyzer https://firmalyzer.com Firmalyzer enables device vendors and security professionals to perform an automated security assessment on software that powers IoT devices (firmware) to identify configuration and application vulnerabilities
        - RIoT Vulnerability Scanner https://www.beyondtrust.com
        - Foren6 https://cetic.github.io
        - IoT Inspector https://www.iot-inspector.com
        - RFCrack https://github.com
        - HackRF One https://greatscottgadgets.com
- IoT Attack Countermeasures
    - Disable the “guest” and “demo” user accounts if enabled
    - Use the “Lock Out” feature to lock out accounts for excessive invalid login attempts
    - Implement a strong authentication mechanism
    - Locate control system networks and devices behind firewalls, and isolate them from the business network
    - Implement IPS and IDS in the network
    - Implement end-to-end encryption and use public key infrastructure (PKI)
    - Use VPN architecture for secure communication
    - Deploy security as a unified, integrated system
    - Allow only trusted IP addresses to access the device from the Internet
    - Disable telnet (port 23)
    - Disable the UPnP port on routers
    - Protect the devices against physical tampering
    - Patch vulnerabilities and update the device firmware regularly

  - Monitor traffic on port 48101, as infected devices attempt to spread the malicious file using port 48101

  - Position of mobile nodes should be verified with the aim of referring one physical node with one vehicle identity only, which means one vehicle cannot have two or more identities

  - Data privacy should be implemented; therefore, the user’s account or identity should be kept protected and hidden from other users

  - Data authentication should be performed to confirm the identity of the original source node

  - Maintain data confidentiality using symmetric key encryption

  - Implement a strong password policy requiring a password at least 8–10 characters long with a combination of letters, numbers, and special characters

  - Use CAPTCHA and account lockout policy methods to avoid brute-force attacks

  - Use devices made by manufacturers with a track record of security awareness

  - Isolate IoT devices on protected networks

  - IoT Security Tools

    - SeaCat.io https://www.teskalabs.com SeaCat.io is a security-first SaaS technology to operate IoT products in a reliable, scalable, and secure manner
    - DigiCert IoT Device Manager (https://www.digicert.com)
    - FortiNAC (https://www.fortinet.com)
    - darktrace (https://www.darktrace.com)
    - Symantec Critical System Protection (https://www.symantec.com)
    - Cisco IoT Threat Defense (https://www.cisco.com)

- OT Attacks

- OT Concepts

  - Operational Technology (OT) is the software and hardware designed to detect or cause changes in industrial operations through direct monitoring and/or controlling of industrial physical devices

  - OT consists of Industrial Control Systems (ICS) to monitor and control the industrial operations

  - Essential Terminology

    - Assets: OT systems consist of physical assets such as sensors and actuators, servers, workstations, network devices, and PLCs, and logical assets such as flow graphics, program logic, databases, firmware, and firewall rules
    - Zones and Conduits: A network segregation technique used to isolate the networks and assets to impose and maintain strong access control mechanisms
    - Industrial Network: A network of automated control systems is known as an industrial network

        - Business Network: It comprises of a network of systems that offer information infrastructure to the business
        - Industrial Protocols: Protocols used for serial communication and communication over standard Ethernet. Ex: S7, CDA, CIP, Modbus, etc.
        - Network Perimeter: It is the outermost boundary of a network zone i.e. closed group of assets
        - Electronic Security Perimeter: It is referred to as the boundary between secure and insecure zones
        - Critical Infrastructure: A collection of physical or logical systems and assets that the failure or destruction of which will severely impact the security, safety, economy, or public health

    - IT/OT Convergence (IIOT)

        - IT/OT convergence is the integration of IT computing systems and OT operation monitoring systems to bridge the gap between IT/OT technologies for improving overall security, efficiency, and productivity

        - The IT/OT convergence can enable smart manufacturing known as industry 4.0, where IoT applications are used in industrial operations

        - Using this Internet of Things (IoT) for industrial operations such as monitoring supply chains, manufacturing and management systems is referred to as Industrial Internet of Things (IIoT)

        - Benefits of merging OT with IT

            - Enhancing Decision Making: Decision making can be enhanced by integrating OT data into business intelligence solutions.
            - Enhancing Automation: Business flow and industrial control operations can be optimized by OT/IT merging; together they can improve the automation.
            - Expedite Business Output: IT/OT convergence can organize or streamline development projects to accelerate business output.
            - Minimizing Expenses: Reduces the technological and organizational overheads.
            - Mitigating Risks: Merging these two fields can improve overall productivity, security, and reliability, as well as ensuring scalability.

    - The Purdue Model

        - The Purdue model is derived from the Purdue Enterprise Reference Architecture (PERA) model, which is a widely used to describe internal connections and dependencies of important components in the ICS networks
        - The three zones are further divided into several operational levels
            - Enterprise Zone (IT Systems)
                - Level 5 (Enterprise Network)
                - Level 4 (Business Logistics Systems)
            - Manufacturing Zone (OT Systems)
                - Level 3 (Operational Systems/Site Operations)
                - Level 2 (Control Systems/Area Supervisory Controls)
                - Level 1 (Basic Controls/Intelligent Devices)

            - Level 0 (Physical Process)
        - Industrial Demilitarized Zone (IDMZ)

- OT Threats and Attacks

    - Challenges of OT

        - Lack of visibility: Broader cybersecurity visibility in the OT network achieves greater security and so one can rapidly respond to any potential threats. However, most organizations do not have clear cybersecurity visibility, making it difficult for the security teams to detect unusual behaviors and signatures.
        - Plain-text passwords: Most industrial site networks use either weak or plain-text passwords. Plain-text passwords lead to weak authentication, which in turn leaves the systems vulnerable to various cyber-reconnaissance attacks.
        - Network complexity: Most OT network environments are complex due to comprising numerous devices, each of which has different security needs and requirements.
        - Legacy technology: OT systems generally use older technologies without appropriate security measures like encryption and password protection, leaving them vulnerable to various attacks. Applying modern security practices is also a challenge.
        - Lack of antivirus protection: Industries using legacy technology and outdated systems are not provided with any antivirus protection, which can update signatures automatically, thus making them vulnerable to malware infections.
        - Lack of skilled security professionals: The cybersecurity skills gap poses a great threat to organizations, as there is a lack of skilled security professionals to discover threats and implement new security controls and defenses in networks.
        - Rapid pace of change: Maintaining the pace of change is the biggest challenge in the field of security, and slow digital transformation can also compromise OT systems.
        - Outdated systems: Most OT devices, such as PLCs, use outdated firmware, making them vulnerable to many modern cyberattacks.
        - Haphazard modernization: As the demand for OT grows, it must stay up to date with the latest technologies. However, due to the use of legacy components in OT system upgrading and patching, updating the system can take several years, which can adversely affect several operations.
        - Insecure connections: OT systems communicate over public Wi-Fi and unencrypted Wi-Fi connections in the IT network for transferring control data, making them susceptible to man-in-the-middle attacks.
        - Usage of rogue devices: Many industrial sites have unknown or rogue devices connected to their networks, which are vulnerable to various attacks.
        - Convergence with IT: OT mostly connects with the corporate network; as a result, it is vulnerable to various malware attacks and malicious insiders. In addition, the OT systems are IT enabled, and the IT security team does not have much experience with the OT systems and protocols.
        - Organizational challenges: Many organizations implement and maintain different security architectures that meet the needs of both IT and OT. This can create some flaws in security management, leaving ways for the attackers to intrude into the systems easily.
        - Unique production networks/proprietary software: Industries follow unique hardware and software configurations that are dependent on industry standards and explicit operational

    demands. The use of proprietary software makes it difficult to update and patch firmware, as multiple vendors control it.
    - Vulnerable communication protocols: OT uses communication protocols such as Modbus and Profinet for supervising, controlling, and connecting different mechanisms such as controllers, actuators, and sensors. These protocols lack in-built security features such as authentication, detection of flaws, or detection of abnormal behavior, making them vulnerable to various attacks.
    - Remote management protocols: Industrial sites use remote management protocols such as RDP, VNC, and SSH. Once the attacker compromises and gains access to the OT network, he/she can perform further exploitation to understand and manipulate the configuration and working of the equipment.

  - OT Threats

    - Maintenance an Administrative Threat
    - Data Leakage
    - Protocol Abuse
    - Potential Destruction of ICS Resources
    - Reconnaissance Attacks
    - Denial-of-Service Attacks
    - HMI-based Attacks
    - Exploiting Enterprise Specific Systems and Tools
    - Spear Phishing
    - Malware Attacks
    - Exploiting Unpatched Vulnerabilities
    - Side-Channel Attacks
    - Buffer Overflow Attacks
    - Exploiting RF Remote Controllers

  - OT Attacks. HMI-based Attacks

    - Attackers often try to compromise the HMI system as it is the core hub that controls the critical infrastructure

    - Attackers gain access to the HMI systems to cause physical damage to the SCADA devices or collect sensitive information related to the critical architecture

    - SCADA vulnerabilities exploited by attackers to perform HMI-based attacks:

      - Memory Corruption
      - Lack of Authorization/Authentication and Insecure Defaults
      - Credential Management
      - Code Injection

  - Side-Channel Attacks

    - Attackers perform a sidechannel attack by monitoring its physical implementation to obtain critical information from a target system
    - Attackers use two techniques namely timing analysis and power analysis to perform sidechannel attacks on the target OT systems

    - Timing Analysis: Attackers monitor the amount of time the device is taking to finish one complete password authentication process to determine the number of correct characters
    - Power Analysis
      - Attackers observe the change in power consumption of semiconductors during clock cycles
      - By observing the power profile, one character of the password can be retrieved comparing the correct character with the wrong character

  - Hacking Programmable Logic Controller (PLC)

    - Programmable Logic Controllers (PLCs) are susceptible to cyber-attacks as they are used for controlling the physical processes of critical infrastructure
    - Attackers identify PLCs exposed to the Internet using online tools such as Shodan
    - Attackers can tamper with the integrity and availability of PLC systems by exploiting pin control operations

  - Hacking Industrial Systems through RF Remote Controllers

    - Most industrial machines are operated via remote controllers that are used in various industries such as manufacturing, logistics, mining, and constructions for automation or to control machines

    - Improper security implementations in the devices operating via remote controllers can pose severe risks to the industrial systems

    - Replay Attack: Attackers record the commands transmitted by an operator and replay them to the target system to gain basic control over the system

    - Command Injection: Attackers alter RF packets or inject their own packets employing reverse engineering techniques to gain complete access over the target machine

    - Re-pairing with Malicious RF controller: Attackers hijack the original remote controller and pair it with the machine using a malicious RF controller, which they disguise as a legitimate one

    - Malicious Reprogramming Attack: Attackers inject malware into the firmware of the remote controllers to maintain a persistent and completely remote access to the system

  - OT Attack Tools

    - ICS Exploitation Framework (ISF) https://github.com ICS Exploitation Framework (ISF) is an exploitation framework based on Python and is like the Metasploit framework
    - SCADA Shutdown Tool https://github.com
    - GRASSMARLIN https://github.com
    - Metasploit https://www.metasploit.com
    - modbus-cli https://github.com
    - PLCinject https://github.com

- OT Attack Countermeasures

  - Regularly conduct a risk assessment to reduce the current risk exposure

  - Use purpose-built sensors to discover the vulnerabilities in the network inactively
  - Incorporate threat intelligence to uncover threats and protect assets by prioritizing OT patches
  - Regularly upgrade OT hardware and software tools
  - Disable unused ports and services
  - Update systems to the latest technologies and patch systems regularly
  - Implement secure configuration and secure coding practices for OT applications
  - Maintain an asset register to track the information and to scrutinize outdated and unsupported systems
  - Perform continuous monitoring and detection of the log data generated by the OT systems for detecting real-time attacks
  - Train employees with the latest security policies and raise awareness of the latest threats and risks
  - Use strong and secure passwords using hashing, and change the default factory-set passwords
  - Secure remote access through multiple layers of defense by implementing two-factor authentication, VPNs, encryption, firewalls, etc.
  - Implement incident response and business continuity plans
  - Secure the network perimeter to filter and prevent unauthorized inbound traffic
  - Regularly scan systems and networks using anti-malware tools
  - Restrict network traffic by using techniques like rate-limiting and whitelisting to prevent DoS and brute-forcing attacks
  - Harden the systems by disabling unused services and functionalities
  - Use only tested and familiar third-party web servers for serving the ICS web applications
  - Ensure ICS vendors add cryptographic signatures to application updates
  - Perform periodic audits of industrial systems to validate the security controls, production, and management systems
  - OT Security Tools
    - Flowmon https://www.flowmon.com Flowmon empowers manufacturers and utility companies to ensure the reliability of their industrial networks to avoid downtime and disruption of service continuity
    - tenable.ot https://www.tenable.com
    - Forescout https://www.forescout.com
    - PA-220R https://www.paloaltonetworks.com
    - Fortinet ICS/SCADA solution https://www.fortinet.com
    - Nozomi Networks Guardian https://www.nozominetworks.com

Module 11: Cloud Computing Threats and Countermeasures

- Cloud Computing Concepts
    - Cloud computing is an on-demand delivery of IT capabilities where IT infrastructure and applications are provided to subscribers as a metered service over a network
    - Characteristics of Cloud Computing
        - On-demand self-service
        - Distributed storage
        - Rapid elasticity
        - Automated management
        - Broad network access
        - Resource pooling
        - Measured service
        - Virtualization technology
    - Limitations of Cloud Computing
        - Limited control and flexibility of organizations
        - Proneness to outages and other technical issues
        - Security, privacy, and compliance issues
        - Contracts and lock-ins
        - Dependence on network connections
        - Potential vulnerability to attacks as every component is online
        - Difficulty in migrating from one service provider to another
    - Types of Cloud Computing Services
        - Infrastructure-as-a-Service (IaaS)
            - Provides virtual machines and other abstracted hardware and operating systems which may be controlled through a service API
            - E.g., Amazon EC2, Microsoft OneDrive, or Rackspace
        - Platform-as-a-Service (PaaS)
            - Offers development tools, configuration management, and deployment platforms on-demand that can be used by subscribers to develop custom applications
            - E.g., Google App Engine, Salesforce, or Microsoft Azure
        - Software-as-a-Service (SaaS)
            - Offers software to subscribers on-demand over the Internet
            - E.g., web-based office applications like Google Docs or Calendar, Salesforce CRM, or Freshbooks
        - Identity-as-a-Service (IDaaS)

        - Offers IAM services including SSO, MFA, IGA, and intelligence collection
        - E.g., OneLogin, Centrify Identity Service, Microsoft Azure Active Directory, or Okta

    - Container-as-a-Service (CaaS)

        - Offers virtualization of container engines, and management of containers, applications, and clusters, through a web portal or API
        - E.g., Amazon AWS EC2, or Google Kubernetes Engine (GKE)

    - Security-as-a-Service (SECaaS)

        - Provides penetration testing, authentication, intrusion detection, anti-malware, security incident, and event management services
        - E.g., eSentire MDR, Switchfast Technologies, OneNeck IT Solutions, or McAfee Managed Security Services

    - Function-as-a-Service (FaaS)

        - Provides a platform for developing, running, and managing application functionalities for microservices
        - E.g., AWS Lambda, Google Cloud Functions, Microsoft Azure Functions, or Oracle Cloud Fn

- Separation of Responsibilities in Cloud

- Cloud Deployment Models

    - Public Cloud: Services are rendered over a network that is open for public use
    - Private Cloud: Cloud infrastructure is operated for a single organization only
    - Community Cloud: Shared infrastructure between several organizations from a specific community with common concerns (security, compliance, jurisdiction, etc.)
    - Hybrid Cloud: Combination of two or more clouds (private, community, or public) that remain unique entities but are bound together, thereby offering the benefits of multiple deployment models
    - Multi Cloud: Dynamic heterogeneous environment that combines workloads across multiple cloud vendors, managed via one proprietary interface to achieve long term business goals

- NIST Cloud Deployment Reference Architecture

    - NIST cloud computing reference architecture defines five major actors:
        - Cloud Consumer
        - Cloud Provider
        - Cloud Carrier
        - Cloud Auditor
        - Cloud Broker

- Cloud Storage Architecture

    - Cloud storage is a data storage medium used to store digital data in logical pools using a network

    - The cloud storage architecture consists of three main layers namely, front-end, middleware, and back-end
    - The Front-end layer is accessed by the end user where it provides APIs for the management of data storage
    - The Middleware layer performs several functions such as data de-duplication and replication of data
    - The Back-end layer is where the hardware is implemented

  - Cloud Service Providers

    - Amazon Web Service (AWS) https://aws.amazon.com
    - Microsoft Azure https://azure.microsoft.com
    - Google Cloud Platform (GCP) https://cloud.google.com
    - IBM Cloud https://www.ibm.com

- Container Technology

  - A container is a package of an application/software including all its dependencies such as library files, configuration files, binaries, and other resources that run independently of other processes in the cloud environment

  - CaaS is a service that includes the virtualization of containers and container management through orchestrators

  - Features:

    - Portability and consistency
    - Security
    - High efficiency and cost effectiveness
    - Scalability
    - Robustness

  - Container Technology Architecture

    - Tier-1: Developer machines; image creation, testing and accreditation
    - Tier-2: Testing and accreditation systems; verification and validation of image contents, signing images and sending them to the registries
    - Tier-3: Registries; storing images and disseminating images to the orchestrators based on requests
    - Tier-4: Orchestrators; transforming images into containers and deploying containers to hosts
    - Tier-5: Hosts; operating and managing containers as instructed by the orchestrator

  - Advantages:

    - Minimum number of resources needed to develop an application
    - Faster detection of software issues and deployment of patches
    - Cost-effectiveness and easy shipping
    - Increased application portability
    - Scalable resources

    - Quick container boot (in seconds) so that applications can be developed in a rapid phase
    - Easy management of isolated applications in containers
    - Easy testing and debugging

  - Disadvantages:

    - Increased complexity
    - Lack of staff expertise results in misconfigurations
    - Increased vulnerability owing to shared resources
    - Questionable container performance
    - Difficulty in selecting a platform to run containers
    - Variations in service discovery (Proxy-based, DNS-based, etc.)

  - Containers Vs. Virtual Machines

    - Virtualization is the ability to run multiple operating systems on a single physical system and share the underlying resources such as a server, storage device, or network

    - Containers are placed on the top of one physical server and host operating system, and share the operating system’s kernel binaries and libraries, thereby reducing the need for reproducing the OS

    - Virtual Machines

      - Heavyweight
      - Run on independent operating systems
      - Hardware-based virtualization
      - Slower provisioning
      - Limited performance
      - Completely isolated making it more secure
      - Created and launched in minutes

    - Containers

      - Lightweight and portable
      - Share a single host operating system
      - OS-based virtualization
      - Scalable and real-time provisioning
      - Native performance
      - Process-level isolation, partially secured
      - Created and launched in seconds

  - What is Docker?

    - Docker is an open source technology used for developing, packaging, and running applications and all its dependencies in the form of containers, to ensure that the application works in a seamless environment
    - Docker provides a Platform-as-a-Service (PaaS) through OS-level virtualization and delivers containerized software packages

  - Microservices Vs. Docker

    - Monolithic applications are broken down into cloud-hosted sub-applications called microservices that work together, each performing a unique task
    - As each microservice is packaged into the Docker container along with the required libraries, frameworks, and configuration files, microservices belonging to a single application can be developed and managed using multiple platforms

  - Docker Networking

    - Docker connects multiple containers and services or other non-Docker workloads together
    - The Docker networking architecture is developed on a set of interfaces known as the Container Network Model (CNM)
    - The CNM provides application portability across heterogeneous infrastructures

  - Container Orchestration

    - An automated process of managing the lifecycles of software containers and their dynamic environments

    - It is used for scheduling and distributing the work of individual containers for microservices-based applications spread across multiple clusters

    - Various tasks can be automated using container orchestrator, such as

      - Provisioning and deployment of containers
      - Failover and redundancy of containers
      - Creating or destroying containers to distribute the load evenly across host infrastructure
      - Moving containers from one host to another on resource exhaustion or host failure
      - Automatic resource allocation between containers
      - Exposing running services to the external environment
      - Performing load balancing, traffic routing, and service discovery between containers
      - Performing a health check of running containers and hosts
      - Ensuring the availability of containers
      - Configuring application-related containers
      - Securing the communication between containers

  - What is Kubernetes?

    - Kubernetes, also known as K8s, is an open-source, portable, extensible, orchestration platform developed by Google for managing containerized applications and microservices

    - Kubernetes provides a resilient framework for managing distributed containers, generating deployment patterns, and performing failover and redundancy for the applications

    - Features provided by Kubernetes:

      - Service discovery: Kubernetes allows a service to be discovered via a DNS name or IP address.
      - Load balancing: When a container receives heavy traffic, Kubernetes automatically distributes the traffic to other containers and performs load balancing.

        - Storage orchestration: Kubernetes allows developers to mount their own storage capabilities, such as local and public cloud storage.
        - Automated rollouts and rollbacks: Kubernetes automates the process of creating new containers, destroying existing containers, and moving all resources from one container to another.
        - Automatic bin packing: Kubernetes can manage a cluster of nodes that run containerized applications. If you specify the resources needed to run the container, such as processing power and memory, Kubernetes can automatically allocate and deallocate resources to the containers.
        - Self-healing: Kubernetes automatically performs a health check of the containers, replaces the failed containers with new containers, destroys failed containers, and avoids advertising unavailable containers to clients.
        - Secret and configuration management: Kubernetes allows users to store and manage sensitive information such as credentials, secure shell (SSH) keys, and OAuth tokens. Application configuration and sensitive information can be deployed and updated without the need to rebuild the container images.

    - Kubernetes Cluster Architecture

- Kubernetes Vs. Docker

    - Docker is open source software that can be installed on any host to build, deploy, and run containerized applications on a single operating system
    - When Docker is installed on multiple hosts with different operating systems, you can use Kubernetes to manage these Docker hosts
    - Kubernetes is a container orchestration platform that automates the process of creating, managing, updating, scaling, and destroying containers
    - Kubernetes can be coupled with any containerization technology such as Docker, Rkt, RunC, and cri-o
    - Both Dockers and Kubernetes are based on microservices architecture, and built using the Go programming language to deploy small, lightweight binaries, and YAML files for specifying application configurations and stacks

- Container Security Challenges

    - Inflow of vulnerable source code: Containers constitute an open-source platform used by developers to regularly update, store, and use images in a repository. This results in an enormous uncontrolled code that may include vulnerabilities, which can compromise security.
    - Large attack surface: The host OS consists of many containers, applications, VMs, and databases in the cloud or on-premises. A large attack surface implies a large number of vulnerabilities and an increased difficulty in detecting them.
    - Lack of visibility: A container engine runs the container, interfaces with the Linux kernel, and creates another layer of abstraction camouflaging the actions of the containers and making it difficult to track activities of specific containers or users.
    - Compromising secrets: Containers require sensitive information, such as API keys, usernames, or passwords, for accessing any services. Attackers who illicitly gain access to this sensitive information can compromise security.

        - DevOps speed: Containers can be executed promptly and, after execution, are stopped and removed. This fugitiveness helps attackers launch attacks and hide themselves without installing any malicious code.
        - Noisy neighboring containers: A container may consume and exhaust all available system resources, which directly affects the operation of other neighboring containers creating a denial-of-service (DoS) attack.
        - Container breakout to the host: Containers that runs as root may break the containment and gain access to the host OS through privilege escalation.
        - Network-based attacks: Attackers may exploit failed containers having active raw sockets and outbound network connections to launch various network-based attacks.
        - Bypassing isolation: Attackers, after compromising the security of a container, may escalate privileges to gain access to other containers or the host itself.
        - Ecosystem complexity: Containers are built, deployed, and managed using multiple vendors and sources. This makes it complex to secure and update the individual components because they originate from different repositories.

    - Container Management Platforms

        - Docker https://www.docker.com A container platform that helps in building, managing, and securing all the applications and deploying them across cloud environments
        - Amazon Elastic Container Service (ECS) (https://aws.amazon.com)
        - Microsoft Azure Container Instances (ACI) (https://azure.microsoft.com)
        - Red Hat OpenShift Container Platform (https://www.openshift.com)
        - Portainer (https://www.portainer.io)
        - HPE Ezmeral Container Platform (https://www.hpe.com)

    - Kubernetes Platforms

        - Kubernetes https://kubernetes.io An open-source container orchestration engine for automating deployment, scaling, and management of containerized applications
        - Amazon Elastic Kubernetes Service (EKS) (https://aws.amazon.com)
        - Docker Kubernetes Service (DKS) (https://www.docker.com)
        - Knative (https://cloud.google.com)
        - IBM Cloud Kubernetes Service (https://www.ibm.com)
        - Google Kubernetes Engine (GKE) (https://cloud.google.com)

- Cloud Computing Threats

    - OWASP Top 10 Cloud Security Risks https://www.owasp.org

    - Data Ownership

        - Organizations use the public cloud for hosting business services instead of a traditional data center.
        - Sometimes using the cloud causes the loss of data accountability and control, whereas using a traditional data center helps in controlling and protecting the data logically and physically.
        - Using the public cloud can jeopardize data recoverability and result in critical risks, which the organization needs to mitigate promptly.

- User Identity Federation
  - Enterprises use services and applications of different cloud providers, creating multiple user identities and complicating the management of multiple user IDs and credentials.
  - Cloud providers have less control over the user lifecycle/offboarding.
- Regulatory Compliance
  - Following regulatory compliance can be complex.
  - Data that is secured in one country may not be secured in another country owing to the lack of transparency and different regulatory laws followed across various countries.
- Business Continuity and Resiliency
  - Performing business continuity in an IT organization ensures that the business can be conducted in a disaster situation.
  - When organizations use cloud services, there is a chance of risk or monetary loss if the cloud provider handles the business continuity improperly.
- User Privacy and Secondary Usage of Data
  - The use of social websites poses a risk to personal data because they are stored in the cloud and most social application providers mine user data for secondary usage.
  - The default share feature in social networking sites can jeopardize the privacy of user personal data.
- Service and Data Integration
  - Organizations must ensure proper protection when proprietary data are transferred from the end-user to the cloud data center.
  - Unsecured data in transit are susceptible to eavesdropping and interception attacks.
- Multi Tenancy and Physical Security
  - Cloud technology uses the concept of multi-tenancy for sharing resources and services among multiple clients, such as networking, databases.
  - Inadequate logical segregation may lead to tenants interfering with each other’s security features.
- Incidence Analysis and Forensic Support
  - When a security incident occurs, investigating applications and services hosted at a cloud provider can be challenging because event logs are distributed across multiple hosts and data centers located at several countries and governed by different laws and policies.
  - Owing to the distributed storage of logs across the cloud, law enforcing agencies may face problem in forensics recovery.
- Infrastructure Security
  - Configuration baselines of the infrastructure should comply with the industry best practices because there is constant risk of malicious actions.

    - Misconfiguration of infrastructure may allow network scanning for vulnerable applications and services to retrieve information, such as active unused ports and default passwords and configurations.

  - Non-Production Environment Exposure

    - Non-production environments are used for application design and development and to test activities internally within an organization.
    - Using non-production environments increases the risk of unauthorized access, information disclosure, and information modification.

  - Cloud Computing Threats

    - Data Breach/Loss
    - Abuse and Nefarious Use of Cloud Services
    - Insecure Interfaces and APIs
    - Insufficient Due Diligence
    - Shared Technology Issues
    - Unknown Risk Profile
    - Unsynchronized System Clocks
    - Inadequate Infrastructure Design and Planning
    - Conflicts between Client Hardening Procedures and Cloud Environment
    - Loss of Operational and Security Logs
    - Malicious Insiders
    - Illegal Access to the Cloud
    - Loss of Business Reputation due to Co-tenant Activities
    - Privilege Escalation
    - Natural Disasters
    - Hardware Failure
    - Supply Chain Failure
    - Modifying Network Traffic
    - Isolation Failure
    - Cloud Provider Acquisition
    - Management Interface Compromise
    - Network Management Failure
    - Authentication Attacks
    - VM-Level Attacks
    - Lock-in
    - Licensing Risks
    - Loss of Governance
    - Loss of Encryption Keys
    - Risks from Changes of Jurisdiction
    - Undertaking Malicious Probes or Scans
    - Theft of Computer Equipment
    - Cloud Service Termination or Failure
    - Subpoena and E-Discovery
    - Improper Data Handling and Disposal
    - Loss/Modification of Backup Data

    - Compliance Risks
    - Economic Denial of Sustainability (EDoS)
    - Lack of Security Architecture
    - Hijacking Accounts
  - Cloud Attacks: Side-Channel Attacks or Cross-guest VM Breaches
    - The attacker compromises the cloud by placing a malicious virtual machine near to a target cloud server and then launches a side-channel attack
    - In a side-channel attack, the attacker runs a virtual machine on the same physical host as the victim's virtual machine and takes advantage of the shared physical resources (processor cache) to steal data (cryptographic keys) from the victim
    - Side-channel attacks can be implemented by any co-resident user due to the vulnerabilities in shared technology resources
  - Cloud Attacks: Wrapping Attack
    - A wrapping attack is performed during the translation of the SOAP message in the TLS layer where attackers duplicate the body of the message and sends it to the server as a legitimate user
  - Cloud Attacks: Man-in-the-Cloud (MITC) Attack
    - MITC attacks are an advanced version of Man-in-the-middle (MITM) attacks
    - The attacker tricks the victim into installing a malicious code, which plants the attacker's synchronization token on the victim's drive
    - Then, the attacker steals the victim's synchronization token and uses the stolen token to gain access to the victim's files
    - Later, the attacker restores the malicious token with the original synchronized token of the victim, thus returning the drive application to its original state and stays undetected
  - Cloud Attacks: Cloud Hopper Attack
    - Cloud Hopper attacks are triggered at the managed service providers (MSPs) and their users
    - Attackers initiate spear-phishing emails with custom-made malware to compromise the accounts of staff or cloud service firms to obtain confidential information
  - Cloud Attacks: Cloud Cryptojacking
    - Cryptojacking is the unauthorized use of the victim's computer to stealthily mine digital currency
    - Cryptojacking attacks are highly lucrative, which involve both external attackers and rogue insiders
    - To perform this attack, the attackers leverage attack vectors like cloud misconfigurations, compromised websites, and client or server-side vulnerabilities
  - Cloud Attacks: Cloudborne Attack
    - Cloudborne is a vulnerability residing in a bare-metal cloud server that enables the attackers to implant a malicious backdoor in its firmware

        - The malicious backdoor can allow the attackers to bypass the security mechanisms and perform various activities such as watching new user’s activity or behavior, disabling the application or server, and intercepting or stealing the data

    - Enumerating S3 Buckets using lazys3

        - Simple storage service (S3) is a scalable cloud storage service used by Amazon AWS where files, folders, and objects are stored via web APIs

        - Attackers often try to the find the bucket’s location and name to test its security and identify vulnerabilities in the bucket implementation

        - lazys3 https://github.com lazys3 is a Ruby script tool that is used to brute-force AWS S3 buckets using different permutations

    - Cloud Attack Tools

        - Nimbostratus https://andresriancho.github.io
            - A tool used for fingerprinting and exploiting Amazon cloud infrastructures
            - It allows attackers to enumerate access to AWS services for the current IAM role, extract the current AWS credentials from metadata, etc.
        - S3Scanner https://github.com
        - Cloud Container Attack Tool (CCAT) https://github.com
        - Pacu https://github.com
        - DumpsterDiver https://github.com
        - GCPBucketBrute https://rhinosecuritylabs.com

- Cloud Attack Countermeasures:

```
  - Enforce data protection, backup, and retention mechanisms
  - Enforce SLAs for patching and vulnerability remediation
  - Vendors should regularly undergo AICPA SAS 70 Type II audits
  - Verify one’s cloud in public domain blacklists
  - Enforce legal contracts in employee behavior policy
  - Prohibit user credentials sharing among users, applications, and services
  - Implement secure authentication, authorization, and auditing controls
  - Check for data protection at both design and runtime
  - Implement strong key generation, storage and management, and destruction practices
  - Monitor the client’s traffic for malicious activities
  - Prevent unauthorized server access using security checkpoints
  - Disclose applicable logs and data to customers
  - Analyze cloud provider security policies and SLAs
  - Assess the security of cloud APIs and log customer network traffic
  - Ensure that the cloud undergoes regular security checks and updates
  - Ensure that physical security is a 24 x 7 x 365 affair
  - Enforce security standards in installation/configuration
  - Ensure that the memory, storage, and network access are isolated
  - Leverage strong two-factor authentication techniques, where possible
  - Apply a baseline security breach notification process
  - Analyze API dependency chain software modules
  - Enforce stringent registration and validation process
  - Perform vulnerability and configuration risk assessment
  - Disclose infrastructure information, security patching, and firewall
details to customers
  - Employ security devices, such as IDS, IPS, and firewall, to guard and
stop unauthorized access to the data stored in the cloud
  - Enforce strict supply chain management and conduct a comprehensive
supplier assessment
  - Enforce stringent security policies and procedures like access control
policy, information security management policy, and contract policy
```

- Side-Channel Attack Countermeasures
    - Implement a virtual firewall in the cloud server back-end of the cloud computing; this prevents the attacker from placing malicious VMs.
    - Implement random encryption and decryption (encrypts data using RSA, 3DES, AES algorithms).
    - Lockdown OS images and application instances to prevent compromising vectors that might provide access.
    - Check for repeated access attempts to local memory and to any hypervisor processes or shared hardware cache by tuning and collecting local process monitoring data and logs for cloud systems.
    - Code the applications and OS components so that they access shared resources, such as memory cache, in a consistent and predictable way. This coding style prevents attackers from collecting sensitive information, such as timing statistics and other behavioral attributes.
- Wrapping Attack Countermeasures
    - Use XML schema validation to detect SOAP messages.
    - Apply authenticated encryption in the XML encryption specification.
- MITC Attack Countermeasures
    - Use an email security gateway to detect the social engineering attacks that can lead to MITCs.
    - Harden the policies of token expiration can prevent this kind of attacks.
    - Use efficient antivirus software that can detect and delete malware.
    - Implement cloud access security broker (CASB) to monitor cloud traffic for detection of anomalies with the generated instances.
    - Monitor employee activities to detect any significant signs of cloud synchronization token abuses.
    - Encrypt the data stored on the cloud and ensure encryption keys are not stored within the same cloud service.
    - Implement two-factor authentication.
- Cloud Hopper Attack Countermeasures

    - Implement multi-factor authentication to prevent compromise of credentials.
    - Ensure mutual co-ordination between customers and CSPs in case of abnormal incidents or activities.
    - Ensure customers are aware and follow the cloud service policies.

  - Cloud Cryptojacking Countermeasures

    - Ensure to implement a strong password policy.
    - Always preserve three different copies of the data in different places and one copy offsite.
    - Ensure to patch the webservers and devices regularly.
    - Use encrypted SSH key pairs instead of passwords for securing access to cloud servers.
    - Implement CoinBlocker URL and IP Blacklist/blackholing in the firewall.
    - Employ real-time monitoring of the web page document object model (DOM) and JavaScript environments for detecting and mitigating malicious activities at an early stage.
    - Use the latest antivirus, anti-malware, and adblocker tools in the cloud.
    - Implement browser extensions for scanning and terminating scripts similar to the CoinHive's miner script.
    - Use endpoint security management technology to detect any rogue applications in the devices.
    - Review all third-party components used by the company’s websites.

  - Cloudborne Attack Countermeasures

    - CSPs should keep the firmware up-to-date.
    - Sanitize the server firmware before it is assigned to new customers.

  - Cloud Security Tools

    - Qualys Cloud Platform https://www.qualys.com An end-to-end IT security solution that provides a continuous, always-on assessment of the global security and compliance posture, with visibility across all IT assets irrespective of where they reside
    - CloudPassage Halo https://www.cloudpassage.com
    - McAfee MVISION Cloud https://www.mcafee.com
    - CipherCloud https://www.ciphercloud.com
    - Netskope Security Cloud https://www.netskope.com
    - Prisma Cloud https://www.paloaltonetworks.com

Module 12: Penetration Testing Fundamentals

- Fundamentals of Penetration Testing and its Benefits

  - Penetration testing is a type of security testing that evaluates an organization’s ability to protect its infrastructure such as network, applications, systems, and users against external as well as internal threats

  - It is an effective way of determining the efficacy of the organization’s security policies, controls, and technologies

  - It involves the active evaluation of the security of the organization’s infrastructure by simulating an attack similar to those performed by real attackers
  - Benefits of Conducting a Penetration Test
    - Reveal vulnerabilities
    - Show real risks
    - Ensure business continuity
    - Reducing client-end attacks
    - Establishing the status of the company in terms of security
    - Guard the reputation of the company
  - Comparing Security Audit, Vulnerability Assessment, and Penetration Testing
    - Security Audit: A security audit checks whether an organization follows a set of standard security policies and procedures
    - Vulnerability Assessment: A vulnerability assessment focuses on discovering the vulnerabilities in an information system but provides no indication of whether the vulnerabilities can be exploited or of the amount of damage that may result from the successful exploitation of the vulnerabilities
    - Penetration Testing: Penetration testing is a methodological approach to security assessment that encompasses a security audit and vulnerability assessment, and it demonstrates whether the vulnerabilities in a system can be successfully exploited by attackers
  - Types of Penetration Assessment: Goal-oriented vs. Compliance-oriented vs. Red-team-oriented
    - Goal-oriented/Objective-oriented Penetration Testing
      - This type of assessments is driven by goals. The objectives of the penetration test are defined, rather than defining the scope of targets
      - The goal of penetration assessment is defined before it begins
      - The job of the pen tester to check whether he/she can achieve the goal and to determine the different ways to achieve the goal
      - Examples:
        - Gain remote access to an internal network
        - Gain access to credit-card information
        - Gain domain administrator access
        - Create a denial of service (DoS) condition against a website
        - Deface a website
    - Compliance-oriented Penetration Testing
      - This type of assessments is driven by compliance requirements. It is testing against adherence to compliance requirements
      - It entails conducting an assessment against the compliance requirements of cyber security standards, frameworks, laws, acts, etc.

        - For example, an organization may ask to perform a security assessment against PCI-DSS requirements
    - Red-team-based Penetration Testing
        - Red-team-based penetration testing is an adversarial goal-based assessment in which the pen tester must mimic the behavior of a real attacker and target the environment
        - This type of assessment has no specific driver
        - For example, an organization may ask to conduct a security assessment for evaluating its overall security. It may include assessing people, networks, applications, physical security, etc.

- Strategies and Phases of Penetration Testing
    - Black-box testing
    - White-box testing
    - Gray-box testing
    - Penetration Testing Process
        - Defining the Scope
            - Extent of testing
            - What will be tested
            - Where testing will be performed from
            - Who will perform testing
        - Performing the Penetration Test
            - Involves gathering all information significant to security vulnerabilities
            - Involves testing the targeted environment such as network configuration, topology, hardware, and software
        - Reporting and Delivering Results
            - Listing vulnerabilities
            - Categorizing risks as high, medium, or low
            - Recommending repairs if vulnerabilities are found
    - Phases of Penetration Testing
        - Pre-attack Phase: Research (Information Gathering)
        - Attack Phase: Testing/Exploitation
        - Post-Attack Phase: Documentation and Reporting
    - Penetration Testing Methodologies
        - Various penetration testing frameworks and methodologies exist to help organizations choose the best method to conduct a successful penetration test

        - Proprietary methodologies

            - EC-Council's Licensed Penetration Tester (LPT)
            - IBM
            - ISS
            - McAfee Foundstone

        - Open-source and public methodologies

            - OSSTMM (Open Source Security Testing Methodology Manual)
            - ISSAF (Information Systems Security Assessment Framework)
            - NIST (National Institute of Standards and Technology)
            - OWASP (Open Web Application Security Project)
            - CREST

- Guidelines and Recommendations for Penetration Testing

    - Characteristics of a Good Penetration Test

        - Establishing the parameters of the penetration test such as objectives, limitations, and justification of procedures
        - Hiring skilled and experienced professionals to perform the test
        - Choosing a suitable set of tests that balance cost and benefits
        - Following a methodology with proper planning and documentation
        - Documenting the result carefully and making it comprehensible for the client

    - When Should Pen Testing Be Performed?

        - Changes have been made in the organization's infrastructure
        - A new threat to the organization's infrastructure has been discovered
        - Hardware or software has been updated or reinstalled
        - The organization's policy has changed

    - Ethics of a Penetration Tester

        - Perform penetration testing with the express written permission of the client.
        - Work according to the nondisclosure and liability clauses of a contract.
        - Test tools in an isolated laboratory prior to an actual penetration test.
        - Inform the client about any possible risks that might emanate from the tests.
        - Notify the client at the first discovery of any highly vulnerable flaws.
        - Deliver social engineering tests results only in a summarized and statistical format.
        - Try to maintain a degree of separation between the criminal hacker and the security professional.

    - Evolving as a Penetration Tester

        - Technologies evolve and change
        - Look outside the workplace to expand knowledge
        - Attend conferences, workshops, and training
        - Join various security groups and discuss current security related topics
        - Keep your career alive by constantly updating your area of knowledge and skill set

    - Read books, journals, and trade magazines
    - Visit various security websites and forums
    - Visit libraries and bookstores to glean information

  - Qualification, Experience, Certifications, and Skills Required for a Pen Tester

    - The quality of penetration testing depends on the tester’s qualifications
    - Penetration testing skills cannot be obtained without years of experience in IT fields such as development, systems administration, or consultancy
    - The tester should possess security certifications such as CEH, CPENT, CISSP, and CISA

  - Qualification

    - Certified Register of Ethical Security Testers (CREST)
    - Cyber-security certifications (CHECK, CTM, CTL, CREST, TIGER, OSCP)
    - A degree in computer security, computer science, or equivalent
    - Recognized security testing certifications (GIAC and CEH)

  - Experience

    - A professional pen tester must have sound knowledge and experience in handling various penetration test tools including open and commercial mapping.
    - They must possess experience in systems, networks, and web-based applications.
    - Experience in using problem-solving techniques and developing a solution to meet vulnerability threats is desirable.
    - They must possess good communication skills to explain technical details to nontechnical parties.
    - They must be proficient at report writing and scripting skills and have good experience at reverse engineering.
    - Consulting experience is an added advantage because they must understand the client’s needs and build a positive relationship with them.

  - Certifications

    - CEH: Certified Ethical Hacker
    - CPENT: Certified Penetration Testing Professional
    - CEPT: Certified Expert Penetration Tester
    - GPEN: GIAC Certified Penetration Tester
    - OSCP: Offensive Security Certified Professional
    - CISSP: Certified Information Systems Security Professional
    - GCIH: GIAC Certified Incident Handler
    - GCFE: GIAC Certified Forensic Examiner
    - GCFA: GIAC Certified Forensic Analyst
    - CCFE: Certified Computer Forensics Examiner
    - CREA: Certified Reverse Engineering Analyst
    - CPTC: Certified Penetration Testing Consultant
    - CPTE: Certified Penetration Testing Engineer
    - CompTIA: Security+
    - CSTA: Certified Security Testing Associate

  - Required skills sets of a penetration tester
    - Strong knowledge of current and emerging technology, methodologies, and tools in the security industry
    - Familiarity with network security concepts, software architecture and design, and engineering processes
    - Knowledge of hardware concepts such as the following:
      - Networking: Transmission Control Protocol/Internet Protocol (TCP/IP) concepts and cabling techniques
      - Ethical hacking techniques: exploits, hacking tools, and so on.
      - Open-source technologies: MySQL and Apache
      - Wireless protocols and devices: 802.11x and Bluetooth
      - Troubleshooting skills
      - Routers, firewalls, and IDS
      - Databases: Oracle and MSSQL
      - OS skills: Windows, Linux, Mainframe, and Mac
      - Web application architecture and Hypertext Transfer Protocol (HTTP) request and response concepts
      - Web servers, mail servers, Simple Network Management Protocol (SNMP) stations, and access devices
  - Communication Skills of a Penetration Tester
    - A penetration tester should have strong interpersonal and communication skills
    - They must have a proven ability to explain the output of a penetration test to a nontechnical client
    - They must have good presentation and reportwriting skills
  - Profile of a Good Penetration Tester
    - Conducted research and development in security
    - Published research papers
    - Presented at various local and international seminars
    - Holds various certifications
    - Member of many reputed organizations such as IEEE
    - Written and published security-related books
    - Previous experience as a pen tester
    - Developed open-source security software tools
    - Participated in “capture the flag” competitions and hackathons
    - Achievements such as appreciation from an organization for work in improving their security
    - Conducted a talk in an international security conference for a chosen topic of relevance
    - Has code configurations in open-source security projects
    - Professional skill set
    - Text free of typos and grammatical mistakes, indicating the ability to write flawless technical reports
  - Responsibilities of a Penetration Tester

        - Perform the penetration testing and risk assessment of the target system.
        - Clearly define the goals of the penetration test, ensure superior quality, and effectively communicate the results.
        - Exploit system vulnerabilities and justify found vulnerabilities.
        - Present reports to superiors on the efficiency of the tests and risk assessments, as well as proposals for risk mitigation.
        - Understand the security of the organization’s servers, network systems, and firewalls relevant to specific business risks
        - Create and design new penetration tools for testing vulnerabilities.
        - Identify the methods and techniques that an attacker could use to exploit weaknesses and logic flaws.
        - Perform social engineering to discover poor password policies or user security practices in an organization.
        - Conduct physical security assessments of servers, systems, and network devices.
        - Investigate web applications, client applications, and standard applications for any vulnerabilities.
        - Include all business considerations such as loss due to downtime and cost of engagement into security strategies.
        - Review and define requirements for information security solutions.
        - Provide feedback, which is very important for the organization to fix security issues.

    - Risks Associated with Penetration Testing

        - Some of the risks arising from penetration testing are as follows:
            - Testers can gain access to protected/sensitive data after a successful penetration test attempt
            - Testers can obtain information about the vulnerabilities existing in the organizational infrastructure
            - DoS penetration tests can take down the organization’s services
            - Using certain pretexts in a social engineering penetration attempt can make employees feel uneasy
        - Organizations can avoid such risks by signing a nondisclosure agreement (NDA) and other legal documents, which include what is allowed and not allowed for the penetration testing team

    - Types of Risks Arising from Penetration Testing

        - Technical risks

            - This type of risks directly arises with targets in the production environment. It includes the following.
                - Failure of the target: Testing continuously consumes a large amount of resources of the target system. This may result in the unavailability of services of the target machine.
                - Disruption of service: The testing process can disrupt some critical services.
                - Loss or exposure of sensitive data: The organization needs to share sensitive data with the pen testers, which may result in the exposure of sensitive data.

        - Organizational risks

            - This type of risks can occur as a side effect of penetration testing. It includes the following.
                - Repetitive and unwanted triggering in the incident-handling processes of the organization
                - Negligence toward monitoring and responding incidents during or after the pen test
                - Disruption in business continuity
                - Loss of reputation

        - Legal risks

            - This type of risks arises from legal obligations due to compliance issues. It includes the violation of laws and clauses in the rules of engagement (ROE).

    - Addressing Risks Associated with Penetration Testing and Avoiding Potential DoS Conditions

        - Use indirect testing: This involves collecting sufficient evidence to prove that a certain vulnerability is likely to exist, instead of directly testing it.
        - Refrain from vulnerability exploitation: Testers should refrain from directly exploiting vulnerabilities. Instead, they should prefer to show the existence of specific vulnerabilities and how they can be exploited.
        - Delay the effect of a test: Testers should attempt to delay the effect of executing a certain test. This will help provide sufficient time to cancel and avoid unwanted risks that may arise from the test.
        - Perform interruptible testing: Testers should be able to pause a certain test if they think that this test may cause unintended consequences.
        - Be careful of using throttled tools: Throttled tools can execute multiple tests simultaneously and can overload the target.
        - Be aware of account lockout functionality: The repetition of a certain test can result in the activation of an account lockout functionality.
        - Use partial isolation and replication of target environment: If possible, testing should be performed on a dedicated test system to avoid any associated risks such as DoS-related situations.
        - Use reserved addresses: If possible, use reserved addresses as the test input to avoid affecting other systems or users.