***Module 30: Web server and application base Attacks***

1. ***Explain MAC spoofing and Email spoofing***

*Ans.* MAC spoofing and email spoofing are both techniques used to impersonate or falsify information in computer communications, but they operate at different layers of the networking stack and have distinct purposes.

**MAC Spoofing:**

* MAC (Media Access Control) spoofing involves changing or impersonating the MAC address of a network interface card (NIC) in a device. The MAC address is a unique identifier assigned to network devices at the hardware level, and it's used for communication on a local network (e.g., Ethernet or Wi-Fi). MAC spoofing is typically used to:
* Bypass network security measures: An attacker may change their MAC address to mimic that of an authorized device to gain unauthorized access to a network.
* Anonymize network traffic: Some users employ MAC spoofing for privacy reasons, as it makes it more challenging to track a specific device on a network.
* Evade network restrictions: In some cases, network administrators may implement access controls or bandwidth limitations based on MAC addresses. Spoofing a MAC address can help evade these restrictions.
* MAC spoofing is generally more relevant in local network settings and is not directly related to email communication.

**Email Spoofing:**

* Email spoofing is a technique used to forge or impersonate the source of an email message to make it appear as though it's sent from a different sender. This is typically done by altering the email header information, which includes the sender's email address, domain, and other metadata. Email spoofing can serve several purposes, including:
* Phishing attacks: Attackers send emails that appear to be from trusted sources, such as banks or government agencies, in an attempt to trick recipients into revealing sensitive information like passwords or credit card numbers.
* Spreading malware: Malicious attachments or links in spoofed emails can lead recipients to download malware or visit infected websites.
* Denial of Service: Spoofed emails can be used to flood an email server with bogus messages, overwhelming the recipient and potentially disrupting email services.

To combat email spoofing, various authentication mechanisms like SPF (Sender Policy Framework), DKIM (DomainKeys Identified Mail), and DMARC (Domain-based Message Authentication, Reporting, and Conformance) have been developed. These technologies help verify the authenticity of email senders and reduce the effectiveness of email spoofing.

In summary, MAC spoofing involves altering the MAC address of a network interface to impersonate another device on a local network, while email spoofing involves forging the sender's email address to deceive recipients. Both techniques can have malicious intentions and are used in different contexts within the realm of computer and network security.

1. ***Perform practical of MITM tool and social engineering Tool***

*Ans. Done in lab.*

1. ***Explain Kali Linux tool SYN Flooding Attack using Metasploit***

*Ans. A SYN flooding attack is a type of denial-of-service (DoS) attack that exploits the three-way handshake process in the Transmission Control Protocol (TCP) to overwhelm a target server by sending a large number of spoofed or maliciously crafted TCP connection requests. The attacker typically floods the target server with a high volume of SYN (synchronize) packets but never completes the handshake by sending the final ACK (acknowledge) packet. This results in the server maintaining half-open connections and consuming resources until it becomes overwhelmed and unresponsive to legitimate traffic.*

*Kali Linux is a popular penetration testing and ethical hacking distribution that includes various tools, including Metasploit, for conducting security assessments. To perform a SYN flooding attack using Metasploit, follow these general steps:*

***Set up your environment:*** *Ensure that you have Kali Linux installed and have Metasploit Framework installed and updated.*

***Launch Metasploit:*** *Open a terminal in Kali Linux and start Metasploit by running the following command:*

*msfconsole*

***Select the SYN flood module:*** *In Metasploit, you can use the "synflood" module to launch a SYN flooding attack. Type the following command:*

*use auxiliary/dos/tcp/synflood*

***Set the required options:*** *You need to specify various parameters, such as the target IP address and port, as well as the number of SYN packets to send. Use the show options command to see the available options and set them accordingly. For example:*

*set RHOST <target\_ip>*  
*set RPORT <target\_port>*  
*set LHOST <your\_ip>*  
*set LPORT <your\_port>*  
*set COUNT <number\_of\_packets>*

*Replace <target\_ip>, <target\_port>, <your\_ip>, <your\_port>, and <number\_of\_packets> with appropriate values for your target and environment.*

***Run the attack:*** *Once you've configured the options, run the SYN flooding attack by executing:*

*exploit*

*Metasploit will start sending a high volume of SYN packets to the target server. As the target server tries to establish connections and receives no ACK packets in response, it can become overwhelmed and unresponsive.*

*Please note that conducting SYN flooding attacks against systems you don't own or have explicit permission to test is illegal and unethical. SYN flooding attacks are considered a form of DoS attack and can have serious consequences. Always ensure you have proper authorization and use these techniques for legitimate security testing, such as assessing the resilience of your own systems or those of clients who have provided explicit consent for testing.*

1. ***Find online email encryption service***

*Ans. There are several online email encryption services that can help you secure your email communications. These services use various encryption methods to protect the content of your emails from unauthorized access. Here are a few popular email encryption services:*

***ProtonMail:*** *ProtonMail is an easy-to-use and secure email service that offers end-to-end encryption for both the email content and attachments. It's known for its strong security features and user-friendly interface.*

***Website:*** [*ProtonMail*](https://protonmail.com/)

***Tutanota:*** *Tutanota is another secure email service that provides end-to-end encryption. It offers a free version with limited features and a premium version for more advanced functionality.*

***Website:*** [*Tutanota*](https://tutanota.com/)

***Hushmail:*** *Hushmail offers encrypted email services with both free and premium plans. It's been in the email encryption business for a long time and is known for its strong encryption standards.*

***Website:*** [*Hushmail*](https://www.hushmail.com/)

*Mailfence: Mailfence is an email service that provides secure and private email, with features like end-to-end encryption, digital signatures, and more. It also offers secure document storage and calendar features.*

***Website:*** [*Mailfence*](https://www.mailfence.com/)

***Virtru:*** *Virtru is an email encryption solution that integrates with various email platforms, including Gmail and Microsoft Outlook. It allows you to send encrypted emails and control who has access to your messages.*

***Website:*** [*Virtru*](https://www.virtru.com/)

***Mailvelope:*** *Mailvelope is a browser extension that works with various web-based email services. It provides OpenPGP encryption, allowing you to encrypt and decrypt your emails directly in your webmail interface.*

***Website:*** [*Mailvelope*](https://www.mailvelope.com/)

*These services offer various levels of security and features, so it's important to choose the one that best fits your needs. Additionally, it's essential to understand how the encryption works and whether both you and your recipients need to be using the same service for end-to-end encryption to be effective. Always take the time to learn how to use the encryption features to ensure your emails remain secure.*

1. ***Types of Firewalls***

*Ans. Firewalls are network security devices or software that are designed to monitor and control incoming and outgoing network traffic based on an organization's previously established security policies. There are several types of firewalls, each with its own approach to filtering and securing network traffic. The main types of firewalls include:*

***Packet Filtering Firewalls:***

* *Packet filtering firewalls work at the network layer (Layer 3) of the OSI model and examine packets of data. They make filtering decisions based on criteria like source and destination IP addresses, source and destination ports, and the protocol used (e.g., TCP, UDP).*
* *They are relatively simple and efficient but offer limited security as they don't inspect the contents of data packets.*

***Stateful Inspection Firewalls:***

* *Stateful inspection firewalls, also known as dynamic packet-filtering firewalls, keep track of the state of active connections. They maintain a state table to record the state of each connection and make filtering decisions based on the state of the connection.*
* *These firewalls are more advanced than packet filtering firewalls and can provide improved security by understanding the context of traffic flows.*

***Proxy Firewalls (Application Layer Firewalls):***

* *Proxy firewalls operate at the application layer (Layer 7) of the OSI model. They act as intermediaries between clients and servers, making requests on behalf of clients.*
* *All incoming and outgoing traffic is relayed through the proxy, allowing it to inspect, filter, and modify data. This provides enhanced security and the ability to perform deep packet inspection.*

***Circuit-Level Gateways:***

* *Circuit-level gateways, or circuit-level proxies, focus on the session layer (Layer 5) of the OSI model. They don't inspect the content of data packets but validate the legitimacy of network sessions.*
* *These firewalls are often used for setting up and controlling network connections, such as VPN tunnels.*

***Application Layer Gateways (ALGs):***

* *ALGs are specialized firewalls that understand and control specific application protocols. They can examine and control data traffic at the application layer.*
* *ALGs are commonly used for protocols like FTP, SIP, and H.323 to ensure that application-specific rules and security measures are enforced.*

***Next-Generation Firewalls (NGFWs):***

* *NGFWs combine the features of traditional firewalls with additional capabilities like deep packet inspection, intrusion prevention, and application-layer filtering.*
* *They offer advanced threat detection and prevention features to protect against a wide range of cyber threats.*

***Hardware Firewalls:***

* *Hardware firewalls are standalone devices designed to protect an entire network. They are often used at the perimeter of a network to filter traffic before it reaches the internal network.*
* *Many modern routers and security appliances include built-in hardware firewalls.*

***Software Firewalls:***

* *Software firewalls are installed on individual devices, such as desktops, laptops, or servers. They provide protection at the device level and can be configured to block or allow specific applications and services.*
* *The choice of firewall type depends on your specific security requirements and the network architecture. Many organizations use a combination of these firewall types to create a layered and comprehensive security posture.*

*The choice of firewall type depends on your specific security requirements and the network architecture. Many organizations use a combination of these firewall types to create a layered and comprehensive security posture.*

1. ***Explain Evading Firewalls***

*Ans. Evading firewalls refers to techniques and strategies used by individuals or entities to bypass or circumvent the security measures implemented by a firewall, with the intention of gaining unauthorized access to a network or server. Evading firewalls is often associated with malicious activities and hacking attempts. Understanding these evasion techniques can help network administrators and security professionals better protect their systems. Here are some common methods used to evade firewalls:*

***Tunneling or VPNs:*** *Attackers can use virtual private networks (VPNs) or tunneling protocols to create an encrypted connection that appears as legitimate traffic to the firewall. This allows them to bypass the firewall's rules and inspection.*

***Proxy Servers:*** *Proxy servers act as intermediaries between the user and the target server. By routing traffic through a proxy server located outside the firewall, attackers can hide their true IP addresses and evade detection.*

***Application Layer Evasion:*** *Some firewalls focus on the application layer and inspect traffic to detect malicious patterns. Attackers may use encoding or obfuscation techniques to make their malicious traffic appear as normal, legitimate data.*

***Protocol-Level Evasion:*** *Attackers can manipulate network protocols to exploit vulnerabilities in the firewall's rule set. For example, they might use fragmentation or other protocol-specific tricks to get malicious data through.*

***IP Spoofing:*** *This involves forging the source IP address of packets to appear as if they are coming from a trusted source within the network. Firewalls can be configured to block traffic from unknown or untrusted IP addresses, so IP spoofing can help attackers avoid these restrictions.*

***DNS Tunneling:*** *Attackers can use DNS (Domain Name System) tunneling to exfiltrate data from a network. By encoding data within DNS queries and responses, they can send and receive information that might otherwise be blocked by a firewall.*

***IP Anonymization and Tor:*** *Services like Tor (The Onion Router) allow users to access the internet while obfuscating their IP addresses by routing traffic through a network of volunteer-run servers. This can be used to evade geographical or IP-based restrictions.*

***Evasion of Intrusion Detection Systems (IDS):*** *Intrusion detection systems work in conjunction with firewalls to identify suspicious activities. Attackers may craft attacks that evade detection by IDS systems while passing through the firewall.*

***Exploiting Firewall Vulnerabilities:*** *In some cases, attackers may identify vulnerabilities in the firewall software itself and exploit them to gain unauthorized access. This can involve exploiting zero-day vulnerabilities or weaknesses in the firewall's configuration.*

*It's important to note that evading firewalls for malicious purposes is illegal and unethical. These techniques are typically used by cybercriminals and hackers to compromise systems and steal data. Organizations should implement strong security practices, regularly update their firewall rules, and employ other security measures, such as intrusion detection systems and regular vulnerability assessments, to mitigate the risk of firewall evasion. Additionally, understanding how these evasion techniques work can help security professionals better defend against them.*

***Web Based Hacking***

1. ***What is Session Hijacking Explain with Techniques?***

*Ans. Session hijacking, also known as session fixation, is a security attack in which an unauthorized person or entity gains control of an established user's session on a web application, typically to impersonate the user and access sensitive information or perform actions on their behalf. Session hijacking can have serious security implications, such as unauthorized access, data theft, or manipulation. Several techniques can be used to perform session hijacking:*

***Session Fixation:***

* *In a session fixation attack, an attacker tricks a user into using a specific session identifier that the attacker already knows. This can be achieved through various means, such as sending a user a malicious link or URL. Once the user logs in with the attacker's session ID, the attacker can take control of the session.*

***Session Sidejacking (Session Sniffing):***

* *Session sidejacking involves intercepting session tokens as they travel over an unsecured network. This is typically done by eavesdropping on unencrypted network traffic, such as unsecured Wi-Fi networks or networks lacking proper encryption protocols.*
* *Tools like Wireshark can be used to capture session tokens, which can then be used by the attacker to hijack the session.*

***Cross-Site Scripting (XSS):***

* *Cross-Site Scripting is a vulnerability that can lead to session hijacking. In an XSS attack, an attacker injects malicious scripts into a website or web application. These scripts run in the context of the user's browser, potentially allowing the attacker to steal session cookies or manipulate the user's session.*

***Man-in-the-Middle (MitM) Attacks:***

* *In MitM attacks, an attacker positions themselves between the user and the target web server, intercepting and possibly modifying the communication between them. This allows the attacker to capture session tokens or manipulate the data exchanged.*
* *Techniques such as ARP spoofing and DNS spoofing can be used to conduct MitM attacks.*

***Session Prediction:***

* *Some web applications generate session IDs using predictable patterns or algorithms. Attackers can guess or predict session IDs, allowing them to hijack sessions. Weak random number generators or algorithms can facilitate this type of attack.*

***Stolen or Leaked Session Data:***

* *If session data is improperly secured, an attacker might gain access to the session tokens directly. This can occur through data breaches, insider threats, or vulnerabilities in the web application.*

*To prevent session hijacking, organizations and developers should implement robust security measures:*

***Use HTTPS:*** *Encrypt web traffic using HTTPS to protect session data during transmission.*

***Implement Secure Session Management:*** *Employ secure session management practices, such as using random, unpredictable session identifiers, and associating session data with client IP addresses.*

***Implement Web Application Firewalls (WAF):*** *WAFs can detect and block malicious requests and attacks, including those related to session hijacking.*

***Use Secure Cookies:*** *Store session identifiers in secure cookies, which can prevent session fixation and make it harder for attackers to access session data.*

***Regularly Audit Code:*** *Periodically review and audit your web application's code for security vulnerabilities, including those that could lead to session hijacking.*

*By implementing these best practices and staying informed about the latest security threats, organizations can reduce the risk of session hijacking and protect user data and privacy.*

1. ***Find DoS/DDoS Attack Tools***

*Ans.*

1. ***Explain SYN Flooding Attack with example***

*Ans. A SYN flooding attack is a type of distributed denial of service (DDoS) attack that exploits the TCP (Transmission Control Protocol) handshake process to overwhelm a target server or network with an excessive number of half-open connections. This attack aims to exhaust the target's resources and make it unavailable to legitimate users.*

*To understand how a SYN flooding attack works, let's break down the TCP handshake process and provide an example:*

***TCP Handshake Process:***

* *A client initiates a connection with a server by sending a TCP SYN (Synchronize) packet to the server.*
* *The server, upon receiving the SYN packet, responds with a SYN-ACK (Synchronize-Acknowledgment) packet to the client.*
* *The client then sends an ACK (Acknowledgment) packet back to the server, and the connection is established.*

***SYN Flooding Attack:***

* *In a SYN flooding attack, the attacker sends a large number of SYN packets to the target server, but the attacker doesn't complete the handshake process by sending the final ACK packet.*
* *The target server allocates resources for each incoming SYN packet, expecting the final ACK to establish a connection.*
* *However, since the attacker doesn't send the final ACK, these connections remain in a half-open state, consuming server resources.*

***Example of a SYN Flooding Attack:***

* *Let's say you are the target server, and an attacker wants to perform a SYN flooding attack on your server.*
* *The attacker uses a botnet (a network of compromised computers) to send a massive number of SYN packets to your server's IP address.*
* *Your server receives a flood of SYN packets from various IP addresses, each requesting a connection.*
* *Your server starts allocating resources for each incoming connection attempt but doesn't receive the expected ACK packets to complete the connections.*
* *As a result, the server's resources, such as memory and processing capacity, become exhausted as it continues to hold onto these half-open connections.*
* *Legitimate users trying to access your server may experience slow response times or even complete unavailability because your server is overwhelmed with these half-open connections.*

*To mitigate SYN flooding attacks, network administrators can implement various strategies, including:*

***SYN Cookies:*** *A technique that allows the server to continue accepting new connection requests even when under attack, without consuming excessive resources.*

***Rate Limiting:*** *Setting limits on the number of new connection requests from a single IP address.*

***Firewalls and Intrusion Detection Systems (IDS):*** *Employing network security measures to detect and block malicious SYN flood traffic.*

***Load Balancers:*** *Distributing incoming traffic across multiple servers to help handle the load and prevent any one server from being overwhelmed.*

***Anomaly Detection:*** *Monitoring network traffic for unusual patterns that may indicate a SYN flooding attack.*

*By implementing these measures, a network can better protect itself against SYN flooding attacks and maintain the availability of its services to legitimate users.*

1. ***List of Web App Hacking Methodology***

*Ans. Web application hacking methodologies are systematic approaches followed by ethical hackers or security professionals to identify vulnerabilities and weaknesses in web applications. Here is a list of some common methodologies and tools used for web application hacking:*

***OWASP Testing Guide:*** *The Open Web Application Security Project (OWASP) provides a comprehensive guide that outlines various testing techniques and methodologies for web application security testing. It covers a wide range of vulnerabilities and attack vectors.*

***OWASP Top Ten:*** *The OWASP Top Ten is a regularly updated list of the most critical web application security risks. Hackers often target these vulnerabilities, so understanding them is crucial.*

***Penetration Testing (Pen Testing):*** *This involves simulating real-world attacks on web applications to identify vulnerabilities. Common tools used include Burp Suite, OWASP ZAP, and Nessus.*

***Burp Suite:*** *A popular web vulnerability scanner and proxy tool used for intercepting and modifying HTTP requests and responses. It's often used for finding security flaws.*

***Nessus:*** *A vulnerability scanner that can identify security issues in web applications and web servers.*

***SQL Injection (SQLi):*** *Attackers exploit poorly sanitized user inputs to manipulate a web application's database. Tools like SQLMap are commonly used for SQL injection testing.*

***Cross-Site Scripting (XSS):*** *In an XSS attack, malicious scripts are injected into web pages, and these scripts are executed in the context of a user's browser. Tools like BeEF and XSStrike can help identify and exploit XSS vulnerabilities.*

***Cross-Site Request Forgery (CSRF):*** *CSRF attacks trick users into performing unwanted actions without their knowledge. Tools like Burp Suite and OWASP ZAP can help identify these vulnerabilities.*

***Path Traversal/Directory Traversal:*** *This involves manipulating file paths to access unauthorized files or directories. Tools like DotDotPwn can automate this process.*

***Command Injection:*** *Attackers exploit vulnerabilities that allow them to execute arbitrary commands on a web server. Tools like OWASP's OWTF can assist in identifying command injection issues.*

***File Upload Vulnerabilities:*** *Attackers may exploit weak file upload functionalities to upload malicious files. Manual testing and tools like Burp Suite are used.*

***Session Management Testing:*** *Evaluating how an application manages sessions, cookies, and user authentication. Tools like OWASP ZAP and Burp Suite are helpful for this.*

***XML External Entity (XXE) Injection:*** *Attackers exploit vulnerabilities in XML parsing to disclose internal files and execute malicious code. Tools like Burp Suite and OWASP ZAP can assist in XXE testing.*

***Server-Side Request Forgery (SSRF):*** *Attackers trick the server into making requests to internal resources. Burp Suite and OWASP ZAP can help identify SSRF vulnerabilities.*

***API Security Testing:*** *With the rise of web APIs, it's crucial to test the security of these interfaces. Tools like Postman and tools for fuzzing and intercepting API requests are used.*

***Content Security Policy (CSP) Bypass:*** *CSP helps prevent XSS attacks. Testing involves attempting to bypass or exploit weaknesses in CSP implementations.*

***Browser Security Testing:*** *Evaluating the security of web browsers and their security features, such as same-origin policy and sandboxing.*

***Web Application Firewalls (WAF) Bypass:*** *Testing the effectiveness of WAFs by attempting to bypass their rules and filters.*

***Client-Side Security Testing:*** *Assessing the security of client-side technologies, such as JavaScript and HTML5.*

***Mobile Application Testing:*** *Mobile apps often interact with web services, so testing their security is essential. Tools like MobSF (Mobile Security Framework) are useful for mobile app testing.*

*Remember that ethical hacking should only be performed on systems you have explicit permission to test. Unauthorized hacking is illegal and unethical. Always follow legal and ethical guidelines when conducting web application security assessments.*

1. ***SQL Injection Methodology***

*Ans. SQL injection is a type of web application vulnerability that allows attackers to execute malicious SQL queries on a web application's database. By exploiting this vulnerability, attackers can gain unauthorized access to data, manipulate databases, and potentially compromise the security of a web application. Below is a methodology for testing and mitigating SQL injection vulnerabilities:*

***Information Gathering:***

***Identify the Target:*** *Choose the web application or specific part of it that you want to test for SQL injection vulnerabilities.*

***Determine Input Points:*** *Identify where user input is accepted by the application, such as search boxes, login forms, or URL parameters.*

***Testing for SQL Injection:***

***Manual Testing:*** *Attempt to inject SQL code directly into input fields. For example, you can try entering single quotes ('), double quotes ("), or SQL keywords like 'OR 1=1--.*

***Automated Tools:*** *Use SQL injection scanning tools like SQLMap, Havij, or OWASP ZAP to automate the detection of SQL injection vulnerabilities.*

***Error-Based SQL Injection:***

***Identify Error Messages:*** *Inject SQL code that triggers error messages. Analyze error messages for information about the database structure, version, and potential vulnerabilities.*

***Blind SQL Injection:***

***Boolean-Based Blind:*** *Send payloads that force the application to respond with "true" or "false" responses, based on whether the injected condition is met.*

***Time-Based Blind:*** *Inject payloads that cause the application to delay its response if the injected condition is met. Observe response times to determine if the injection was successful.*

***Union-Based SQL Injection:***

*Inject UNION SELECT statements to retrieve data from other database tables. The number of columns in the original query must match the number of columns in the UNION SELECT statement.*

***Out-of-Band SQL Injection:***

*In cases where traditional SQL injection techniques don't work, consider exploiting DNS or other out-of-band channels to extract data from the database.*

***Bypassing Filters and WAFs:***

*Some applications use filters or Web Application Firewalls (WAFs) to block SQL injection attempts. Try different encodings, obfuscations, or alternative payloads to bypass these security measures.*

***Data Exfiltration:***

*Once you've successfully identified a SQL injection vulnerability, use it to extract sensitive data from the database. Retrieve usernames, passwords, or other critical information.*

***Post-Exploitation:***

*If you gain unauthorized access to the database, consider the ethical implications of your actions. Don't modify, delete, or steal data without explicit authorization.*

***Remediation and Mitigation:***

*Report the vulnerability to the application owner or administrator.*

*Fix the SQL injection vulnerability by validating and sanitizing user inputs, using parameterized queries (prepared statements), and employing web application firewalls.*

*Regularly monitor and patch your application for security vulnerabilities to prevent future SQL injection attacks.*

*Remember that ethical hacking and penetration testing should always be conducted with proper authorization and in compliance with legal and ethical guidelines. Unauthorized hacking is illegal and unethical.*

1. ***Explain sql injection with any tool***

*Ans. SQL injection is a common web application vulnerability that occurs when an application allows users to enter unsanitized data into SQL queries. It can lead to unauthorized access, data leakage, or even the manipulation of a web application's database. To demonstrate SQL injection, let's use a simple example with a Python-based tool called "SQLMap."*

***Scenario:*** *Imagine a web application that has a search feature, allowing users to search for products by entering a keyword. The application uses user input to construct an SQL query without proper input validation or sanitization, making it vulnerable to SQL injection.*

***Step 1:*** *Installing SQLMap: You can download and install SQLMap from its official website or use package managers like apt or pip if you're using Linux.*

*# On Linux (e.g., Ubuntu)*  
*sudo apt-get install sqlmap*

***Step 2:*** *Identify Vulnerable Input: You need to identify the input field vulnerable to SQL injection. In our example, it's the search box.*

***Step 3:*** *Launch SQLMap: Assuming you have a target URL that includes the search functionality, you can run SQLMap from the command line like this:*

*sqlmap -u "*[*http://example.com/search.php?keyword=test*](http://example.com/search.php?keyword=test)*" --dbs*

*-u specifies the target URL.*

*--dbs tells SQLMap to enumerate the available databases.*

***Step 4:*** *Enumerating Databases: SQLMap will start by probing the input for SQL injection vulnerabilities. If it detects a vulnerability, it will start enumerating the databases on the server.*

***Step 5:*** *Enumerating Tables and Columns: After discovering the databases, SQLMap can list tables within a database, and for each table, it can enumerate the columns. You can do this by specifying the database you want to target and using the --tables and --columns options.*

*sqlmap -u "*[*http://example.com/search.php?keyword=test*](http://example.com/search.php?keyword=test)*" -D database\_name --tables*  
*sqlmap -u "*[*http://example.com/search.php?keyword=test*](http://example.com/search.php?keyword=test)*" -D database\_name -T table\_name --columns*

***Step 6:*** *Dumping Data: You can use SQLMap to extract data from specific tables and columns using the --dump option:*

*sqlmap -u "*[*http://example.com/search.php?keyword=test*](http://example.com/search.php?keyword=test)*" -D database\_name -T table\_name -C column\_name --dump*

*SQLMap will retrieve the data from the specified table and column, displaying it in the terminal.*

*Please note that using SQLMap or any similar tool to exploit web applications without proper authorization is illegal and unethical. This example is for educational purposes only, and you should always obtain explicit permission before conducting any form of penetration testing or security assessments on a web application.*