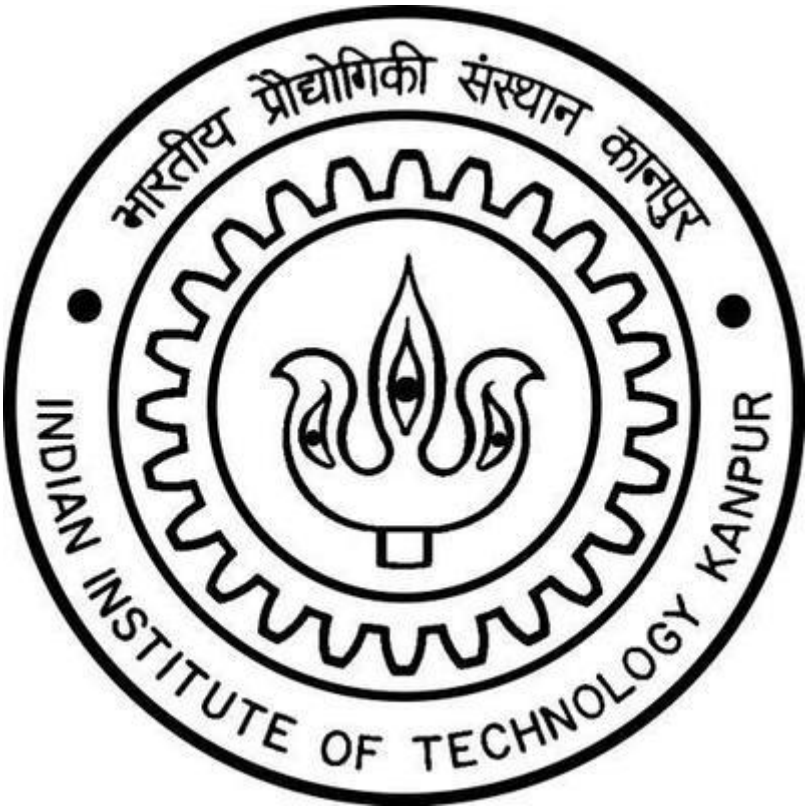
PROJECT REPORT

WEB SERVER PENETRATION TESTING



SUBMITTED BY

SARTHAK SINGH GAUR

(S7S5G4@GMAIL.COM)

IFACET ACADEMY

IIT KANPUR

TABLE OF CONTENTS

EXECUTIVE SUMMARY

The penetration test conducted on the "CertifiedHacker Networks" (https://certifiedhacker.com) web server and e-commerce application has provided valuable insights into the system's security vulnerabilities. The primary focus of the test was to assess the web server's and application's resilience against potential cyber threats and identify areas for improvement. The findings revealed critical vulnerabilities, including an exploitable Cross-Site Scripting (XSS) flaw, missing security headers, and a directory traversal vulnerability.

The web server penetration testing project for "CertifiedHacker Networks" identified critical vulnerabilities, including Cross-Site Scripting (XSS), missing security headers, and a directory traversal flaw. To enhance security, the report recommends strict input validation, output encoding, and CSP implementation to prevent XSS attacks. It also advises enabling HSTS, X-Content-Type-Options, and frame protection headers to mitigate security header issues. Additionally, measures like input validation, file path whitelisting, and restricted access permissions are proposed to address the directory traversal vulnerability effectively.

Implementing these recommendations will significantly improve the security of the "CertifiedHacker Networks" web server and e-commerce application, safeguarding sensitive data and enhancing trust among users. Additionally, it is essential to conduct regular security assessments and stay vigilant to emerging threats to maintain a strong security posture over time.

Overall, this project has been a valuable learning experience that has enriched my skill set and knowledge in cybersecurity. This experience reinforced the significance of adhering to ethical principles in the field of cybersecurity.

INTRODUCTION

This penetration testing project was initiated in response to growing concerns about the security of CertifiedHacker Network's web server hosting its official website and web applications (https://certifiedhacker.com). As cyber threats continue to evolve and become more sophisticated, it has become imperative for organizations to proactively assess their security measures and identify potential vulnerabilities before malicious actors exploit them. Understanding the importance of maintaining a robust security posture, CertifiedHacker Networks decided to conduct a comprehensive web server penetration test.

The primary objective of this penetration testing was to evaluate the security of the web server and its associated applications to ensure the confidentiality, integrity, and availability of critical data and services. By conducting this assessment, CertifiedHacker Networks aimed to identify potential weaknesses and security gaps that could lead to unauthorized access, data breaches, or other detrimental consequences.

The scope of the penetration testing encompassed all aspects of the web server and web applications. This includes the evaluation of the web server's configuration, underlying operating system, web server software, and any supporting services. Additionally, the assessment focused on the security of the web applications hosted on the server, including their authentication mechanisms, input validation, and potential vulnerabilities such as SQL injection and Cross-Site Scripting (XSS).

By undertaking this penetration testing project, CertifiedHacker Networks demonstrated its commitment to maintaining a proactive security stance and safeguarding its online assets. The findings of this assessment will serve as a basis for making informed decisions to address vulnerabilities, strengthen security controls, and enhance employee awareness against social engineering threats. The ultimate goal is to fortify the web server's defenses and protect both the organization and its customers from potential cyber threats.

METHODOLOGY

The penetration testing methodology employed for this web server assessment followed a systematic and structured approach to ensure comprehensive coverage and accuracy in identifying potential vulnerabilities. The testing team utilized a combination of automated scanning, manual testing, and ethical hacking techniques to evaluate the security of the web server and its associated applications. The following steps were undertaken during the testing process:

1. Pre-engagement: The pre-engagement phase involved initial discussions and meetings with CertifiedHacker Network's stakeholders to define the project scope, objectives, and specific requirements. The testing team collaborated closely with the organization's IT and security personnel to understand the web server's architecture, technologies used, and the critical functionalities of the web applications. Legal and ethical considerations were also addressed to ensure compliance with all relevant laws and regulations.

2. Information Gathering: In this phase, the testing team collected information about the web server, its IP address, domain name, and other relevant details through passive reconnaissance techniques. Additionally, the team identified potential entry points and attack vectors by analyzing publicly available information from websites, search engines, and social media platforms. The information gathered during this phase was crucial for understanding the web server's attack surface and tailoring the subsequent testing activities.

3. Vulnerability Scanning: Automated vulnerability scanning tools, such as Nessus and OpenVAS, were utilized to perform an initial assessment of the web server's security. The scanning process identified common known vulnerabilities and misconfigurations in the web server and its applications. This helped the team to prioritize areas that required further investigation and manual testing.

4. Manual Testing: Manual assessment was a critical part of the methodology, as it allowed the team to uncover application-specific vulnerabilities and bypass security controls that automated scanners might miss. The team conducted thorough testing, including attempting SQL injection, Cross-Site Scripting (XSS), and other exploitation techniques to evaluate the web applications' security posture.

5. Exploitation and Proof of Concept (PoC): With prior consent from CertifiedHacker Networks, the team attempted to exploit identified vulnerabilities in a controlled manner. This included demonstrating potential impact scenarios and, where applicable, providing proof-of-

concept (PoC) demonstrations. The goal was to showcase the risks associated with each vulnerability and emphasize the urgency of remediation.

7. Data Protection and Encryption Assessment: The security of sensitive data and customer information handled by the web server and its applications was evaluated. This included an assessment of data protection measures, encryption practices, and compliance with relevant data privacy regulations.

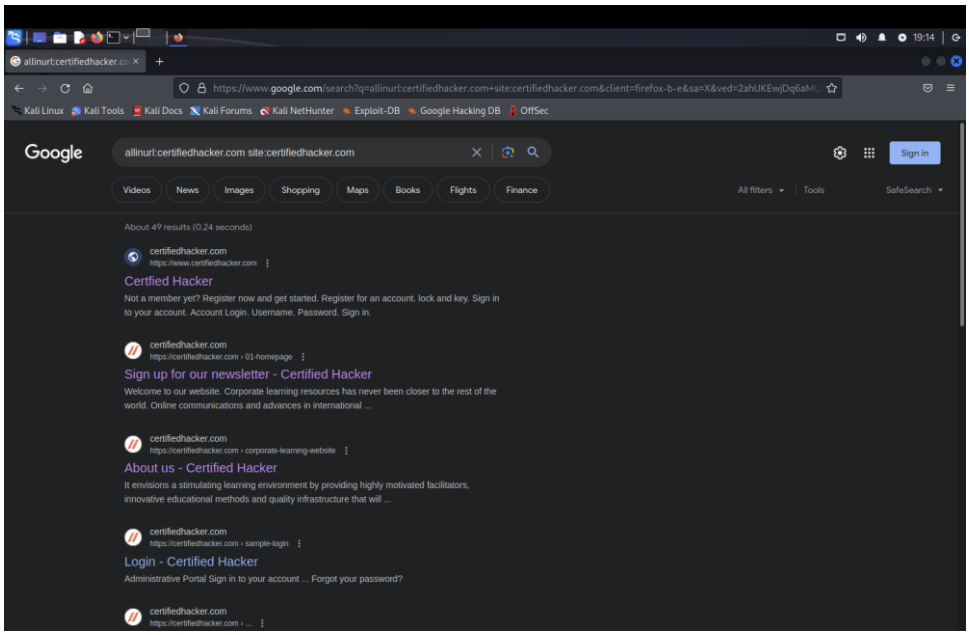
8. Report Generation: After the testing was completed, all findings, including identified vulnerabilities, their severity levels, and potential impact on CertifiedHacker Networks, were documented in a comprehensive report. The report included detailed descriptions of each vulnerability, PoC demonstrations (if applicable), and prioritized recommendations for remediation. The social engineering assessment results and security awareness training recommendations were also included in the report.

9. Review and Remediation: The testing team presented the report to CertifiedHacker Network's stakeholders and addressed any questions or clarifications. The organization's technical team then initiated the remediation process to address the identified vulnerabilities promptly. The recommendations provided in the report served as a guide for implementing appropriate security measures and improving the overall security posture of the web server and applications.

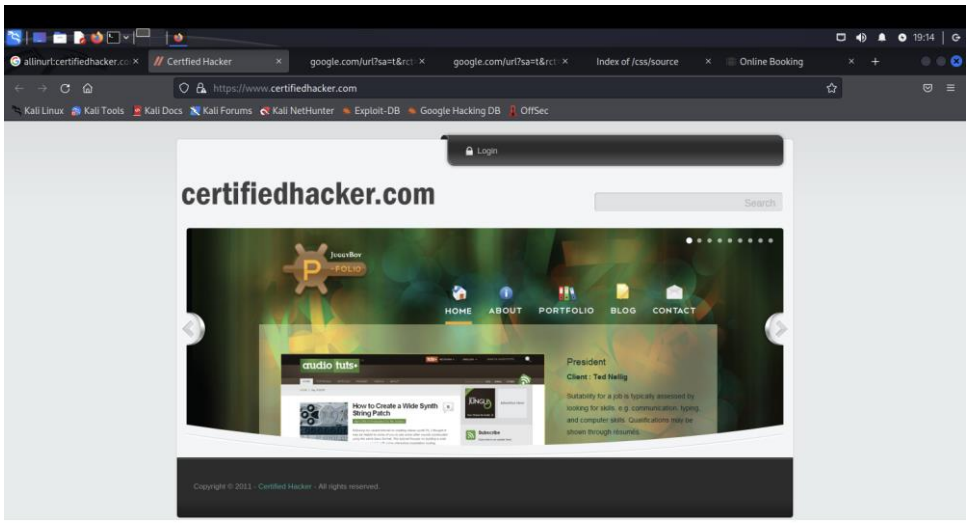
The application of this well-structured methodology enabled the testing team to conduct a thorough and effective web server penetration test. The findings provided CertifiedHacker Network with valuable insights into its web server's security weaknesses and allowed the organization to take proactive measures to address vulnerabilities, secure sensitive data, and enhance employee awareness against social engineering attacks.

FINDINGS

FOOTPRINTING AND RECONNAISSANCE USING ADVANCED GOOGLE HACKING



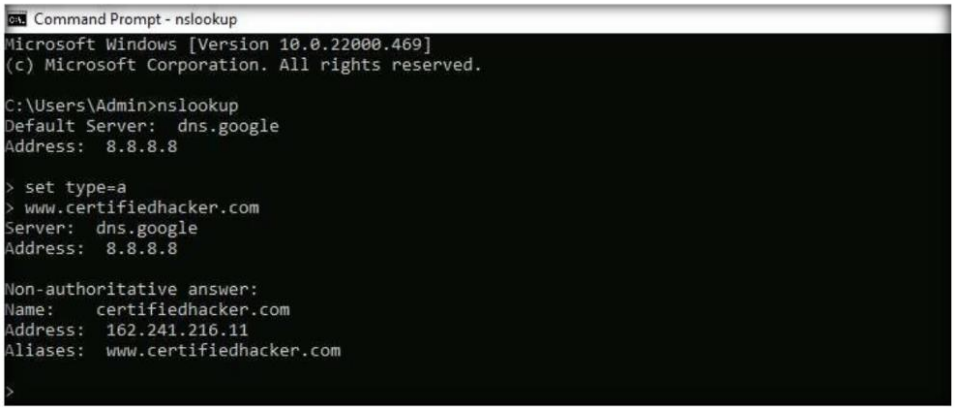
Certified Hacker (https://certifiedhacker.com) is the web server we have been provided with. We use advanced google hacking techniques to gather various details regarding the webpages hosted on the server.



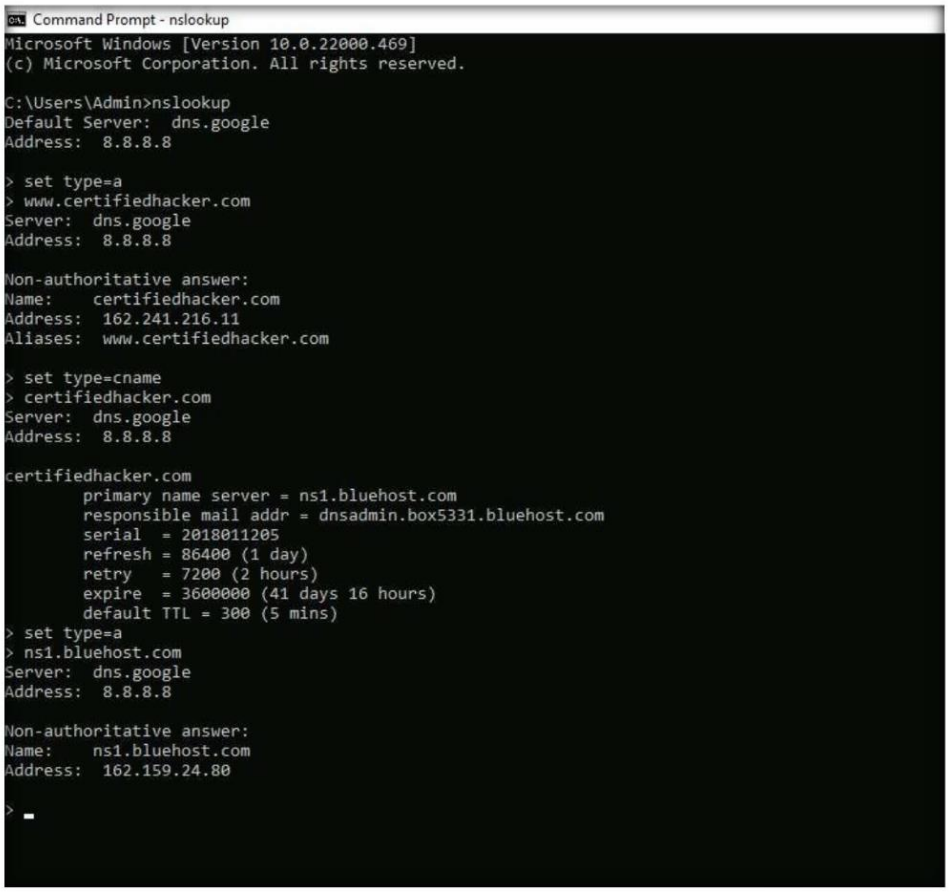
Advanced Google hacking refers to the art of creating complex search engine queries by employing advanced Google operators to extract sensitive or hidden information about a target company from the Google search results. This can provide information about websites that are vulnerable to exploitation.

IP ADDRESS AND NAMESERVER DISCOVERY USING NSLOOKUP

In penetration testing, nslookup (short for Name Server Lookup) is a command-line tool used in penetration testing to query the Domain Name System (DNS). It helps retrieve information about domain names, IP addresses, DNS record types, and performs reverse DNS lookups. Penetration testers use it during the reconnaissance phase to gather target information, identify potential attack vectors, and diagnose DNS issues. Responsible and authorized use is crucial to adhere to ethical guidelines and avoid legal violations.



The webserver has the IP Address as 162.241.216.11



We see that ns1.bluehost.com is the primary name server used by CertifiedHacker.

SERVER TECHNOLOGY AND LOCATION USING NETCRAFT

We use site report feature from Netcraft (https://sitereport.netcraft.com/?url=http://certifiedhacker.com for HTTP and https://sitereport.netcraft.com/?url=https://certifiedhacker.com for HTTPS). Netcraft is a well-known internet security services company that provides a range of tools and services to help organizations and individuals protect their online presence and defend against cyber threats.

Netcraft also offers a variety of services, such as phishing site detection and blocking, malware scanning, and SSL certificate analysis. These services are instrumental in identifying and mitigating potential risks, ensuring the security of web applications, and protecting sensitive data from cyber threats.

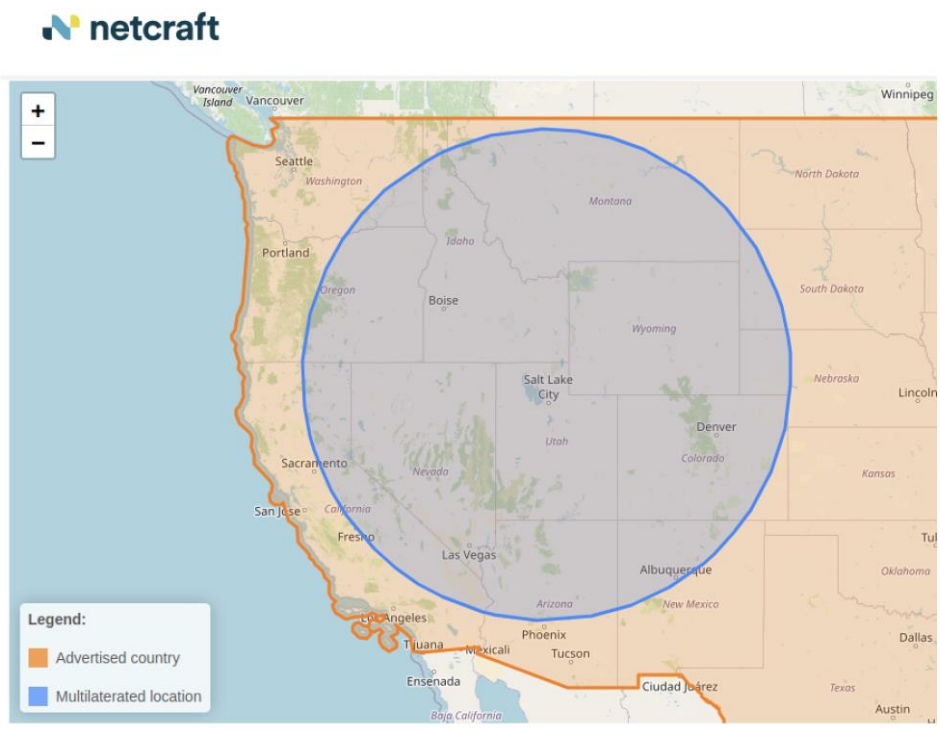
Domain	**certifiedhacker.com**
Nameserver	ns1.bluehost.com
Domain registrar	networksolutions.com
Nameserver organisation	whois.domain.com
Organisation	5335 Gate Parkway care of Network Solutions PO Box 459, Jacksonville, 32256, US
DNS admin	dnsadmin@box5331.bluehost.com
Top Level Domain	Commercial entities (.com)

We can see that the nameserver organisation is whois.domain.com. In the context of penetration testing, the nameserver is an essential target for reconnaissance and information gathering. Penetration testers may interact with the nameserver to gather valuable information about the target's infrastructure, identify potential vulnerabilities, and exploit DNS-related weaknesses.

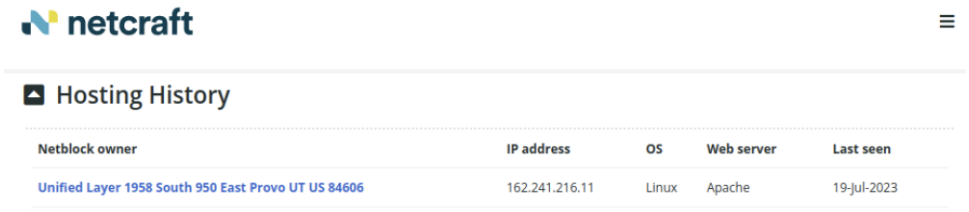
IP delegation

IPv4 address (162.241.216.11)

IP range	Country	Name	Description
::ffff:0.0.0.0/96	United States	IANA-IPV4-MAPPED-ADDRESS	Internet Assigned Numbers Authority
↳ 162.0.0.0-162.255.255.255	United States	NET162	Various Registries (Maintained by ARIN)
↳ 162.240.0.0-162.241.255.255	United States	UNIFIEDLAYER-NETWORK-16	Unified Layer
↳ 162.241.216.11	United States	UNIFIEDLAYER-NETWORK-16	Unified Layer

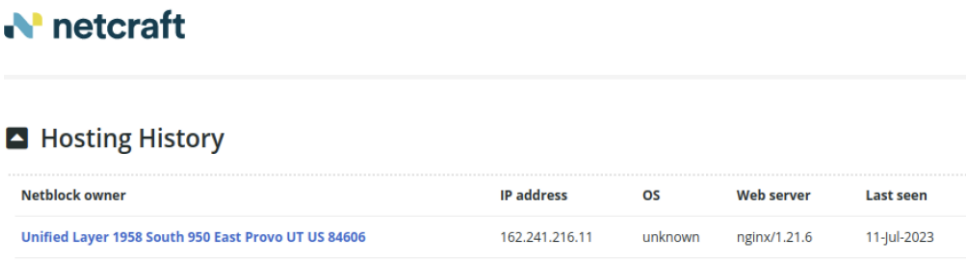


The web server is hosted at 5335 Gate Parkway care of Network Solutions PO Box 459, Jacksonville, 32256, US



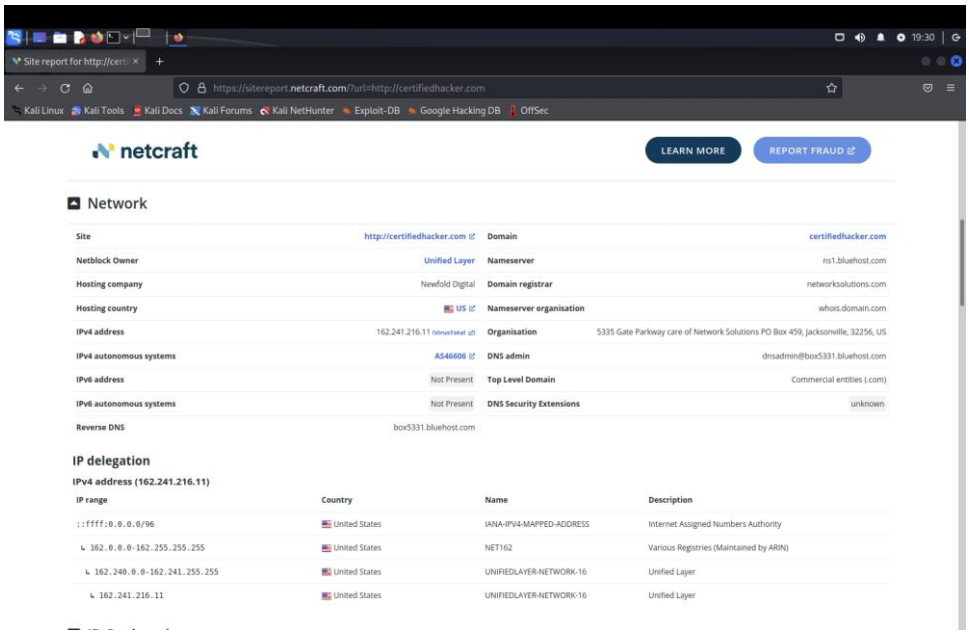
The Webserver is based on Linux OS and utilizes APACHE technology when on HTTP

Apache HTTP Server, a widely used web server technology, is a common target for assessment. The evaluation involves identifying potential vulnerabilities, misconfigurations, and exploits that could be used by attackers. Assessing Apache aims to enhance security, strengthen configurations, and protect against known risks.



The Webserver is based on Linux OS and utilizes nginx/1.21.6 technology when on HTTPS.

Nginx is an efficient and widely used web server and reverse proxy software, is commonly assessed for security weaknesses and vulnerabilities. The evaluation involves identifying misconfigurations, potential exploits, and known vulnerabilities to enhance server security.



The Web server first appeared in December 2002, but the https was only implemented in January 2018

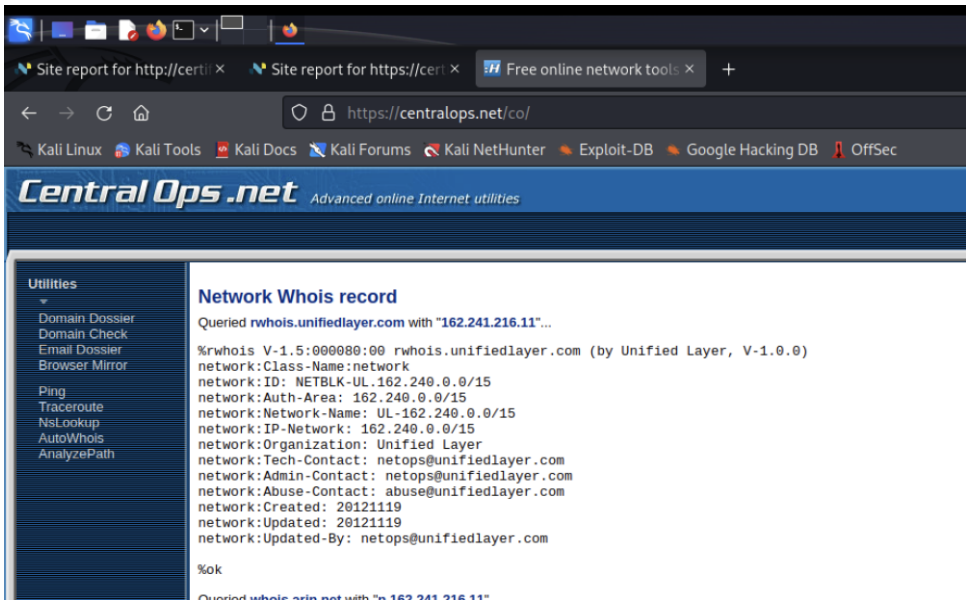
Netblock owner	IP address	OS	Web server	Last seen
Unified Layer 1958 South 950 East Provo UT US 84606	162.241.216.11	Linux	Apache	19-Jul-2023
Unified Layer 1958 South 950 East Provo UT US 84606	162.241.216.11	Linux	unknown	25-Apr-2023
Unified Layer 1958 South 950 East Provo UT US 84606	162.241.216.11	Linux	Apache	24-Apr-2023
Unified Layer 1958 South 950 East Provo UT US 84606	162.241.216.11	Linux	nginx/1.19.10	15-May-2022
Unified Layer 1958 South 950 East Provo UT US 84606	162.241.216.11	Linux	Apache	14-May-2022
Unified Layer 1958 South 950 East Provo UT US 84606	162.241.216.11	Linux	nginx/1.14.1	29-May-2019
Unified Layer 1958 South 950 East Provo UT US 84606	162.241.216.11	Linux	nginx/1.12.2	28-Nov-2018
Unified Layer 1958 South 950 East Provo UT US 84606	69.89.31.193	-	nginx/1.12.1	5-Nov-2017
Unified Layer 1958 South 950 East Provo UT US 84606	69.89.31.193	Linux	Apache	17-Oct-2017
Unified Layer 1958 South 950 East Provo UT US 84606	69.89.31.193	Linux	nginx/1.12.1	6-Oct-2017

The Webserver has always used Linux OS but has shifted across APACHE and nginx

ISP PROVIDED IP RANGE USING CENTRALOPS

CentralOps.net is a web-based platform that offers a variety of internet-related tools and services. It is designed to assist individuals and organizations in performing various internet-related tasks, such as domain name lookup, DNS analysis, IP address tracking, email verification, and WHOIS lookup.

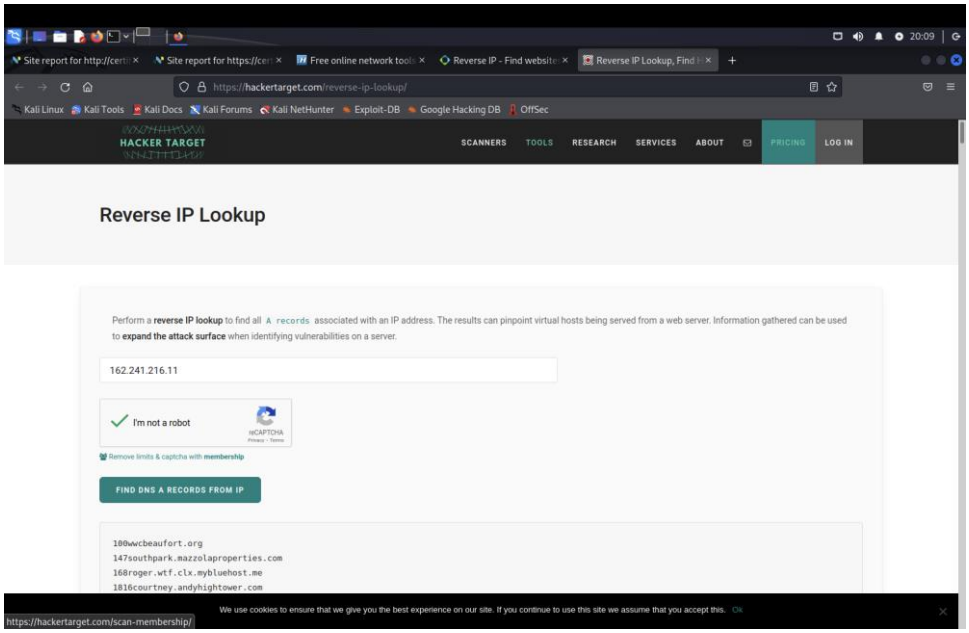
CentralOps.net is widely used by IT professionals, cybersecurity experts, webmasters, and individuals interested in obtaining information about internet resources and performing basic internet-related investigations.



The Webserver uses IP Range 162.240.0.0 - 162.241.255.255 or CIDR- 162.240.0.0/15

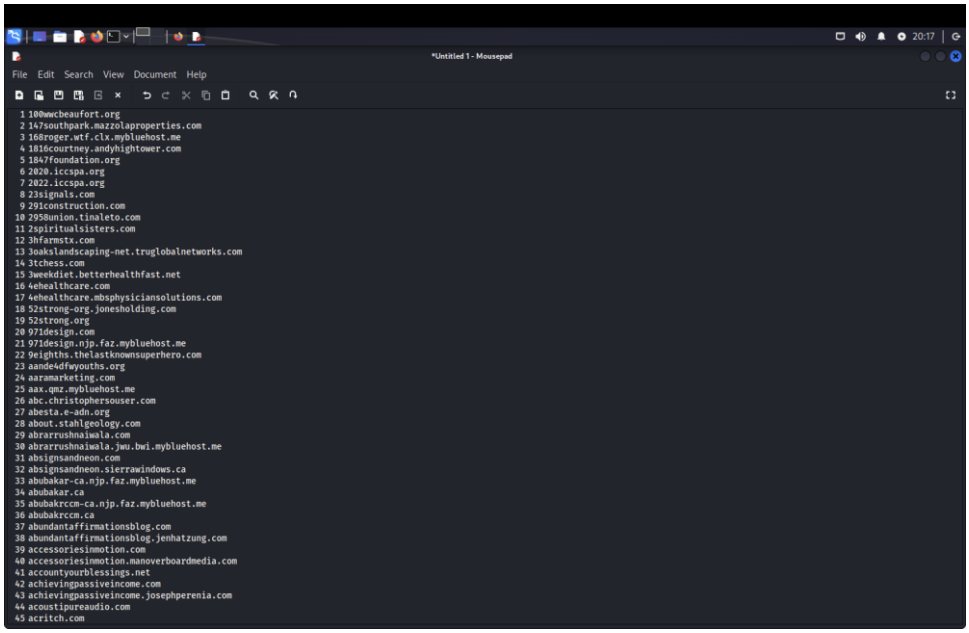
Finding the IP address Range of target system is a fundamental step in penetration testing, and it holds significant importance throughout the entire testing process.

REVERSE IP LOOKUP USING HACKER TARGET



We use Reverse IP Lookup from tools extracted from OSINT FRAMEWORK. The technique known as Reverse IP Lookup is a way to identify hostnames that have DNS (A) records associated with an IP address.

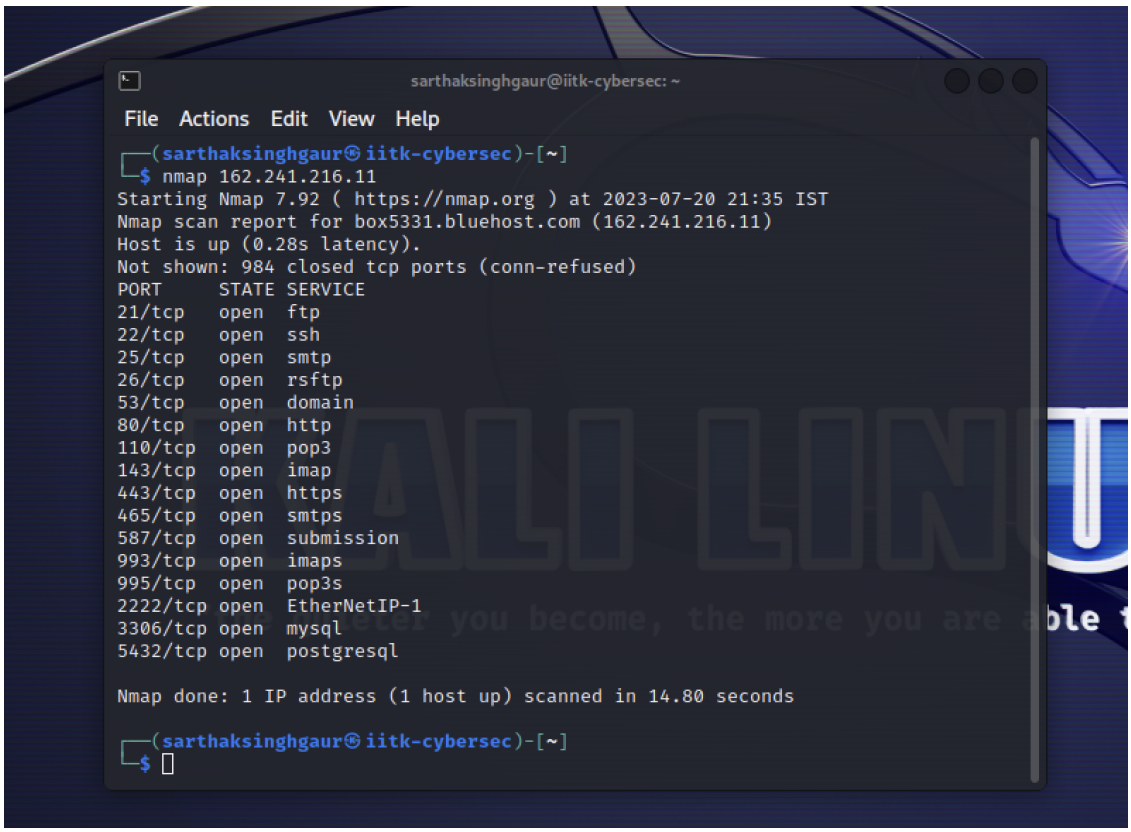
A web server can be configured to serve multiple virtual hosts from a single IP address. This is a common technique in shared hosting environments. It is also common in many organizations and can be an excellent way to expand the attack surface during reconnaissance of a web server.



We find that more than 500 domains are hosted on nameserver ns1.bluehost.com. Some of them are shown in the screenshot attached above.

OPEN PORTS USING NMAP

Nmap (Network Mapper) is a valuable tool in penetration testing for network discovery, port scanning, vulnerability scanning, and service enumeration. It helps identify potential weaknesses and assists in creating a comprehensive map of the target network. Its scripting engine allows for automation, and the tool provides detailed reports for communicating findings to stakeholders.



We can see that ports 21, 22, 25, 53, 80, 110, 143, 443, 587, 993, 995, 3306 and 5432 are open.

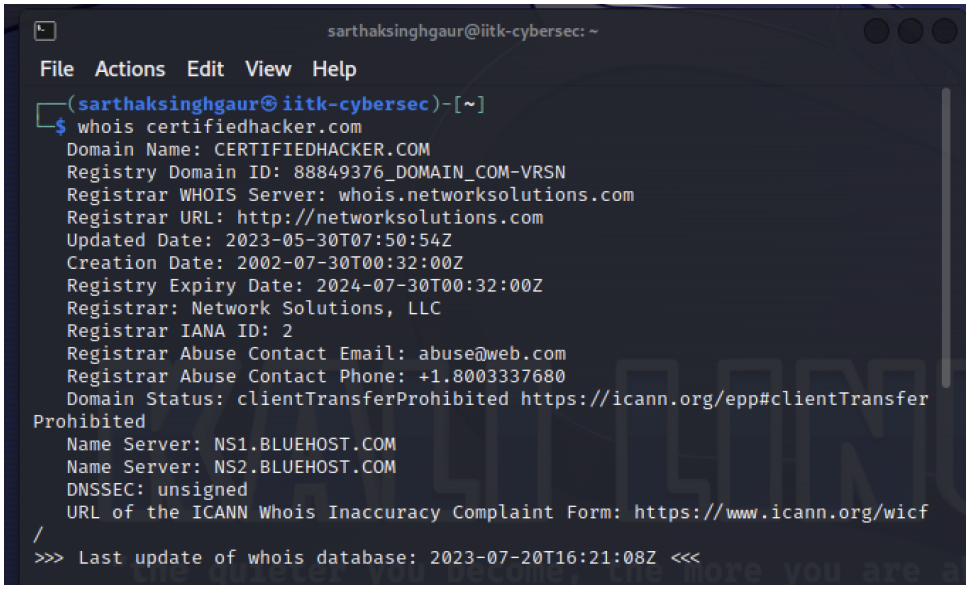
By leveraging Nmap's capabilities in a penetration testing engagement, testers can gain valuable insights into the target network, identify potential vulnerabilities, and make informed decisions on strengthening security defenses and mitigating risks. However, we use another tool from OSINT Framework to double check the open ports on the web server. The results are following.

Title	Port	Result
FTP	21	Open
SSH	22	Open
Telnet	23	Timed-Out
Mail [SMTP]	25	Open
DNS	53	Open
Web Server [HTTP]	80	Open
Mail [POP]	110	Open
netbios	137	Timed-Out
netbios	138	Timed-Out
netbios	139	Timed-Out
Mail [IMAP]	143	Open
Web Server [HTTPS]	443	Open
Microsoft-DS Service	445	Timed-Out
Apple Filesharing Protocol	548	Timed-Out
Mail [SMTP Submission]	587	Open
Mail [IMAP SSL]	993	Open
Mail [POP SSL]	995	Open
Database [MSSQL]	1433	Timed-Out
VPN [L2TP]	1701	Timed-Out
VPN [PPTP]	1723	Timed-Out
Database [MySQL]	3306	Open
Database [PgSQL]	5432	Open
Calendar Server [CalDAV]	8008	Timed-Out
Calendar Server [CalDAV SSL]	8443	Timed-Out

We can see that ports 21, 22, 25, 53, 80, 110, 143, 443, 587, 993, 995, 3306 and 5432 are open.

REGISTRAR DETAILS USING WHOIS PACKAGE IN KALI

In Linux, the whois package is a command-line utility used to query the WHOIS database, which contains registration information about domain names and IP addresses. The WHOIS database holds essential details such as the domain's registrar, registration and expiration dates, contact information for the domain owner, and the domain's name servers.



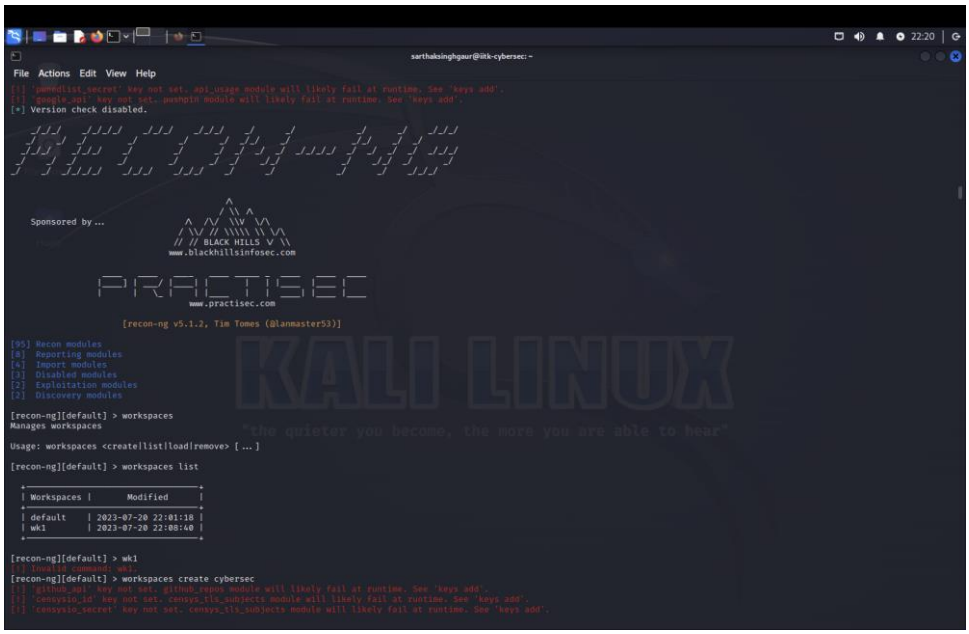
Using the whois package in kali linux, we see that Registrar is Network Soluntions ,LLC with URL http://networksolutions.com

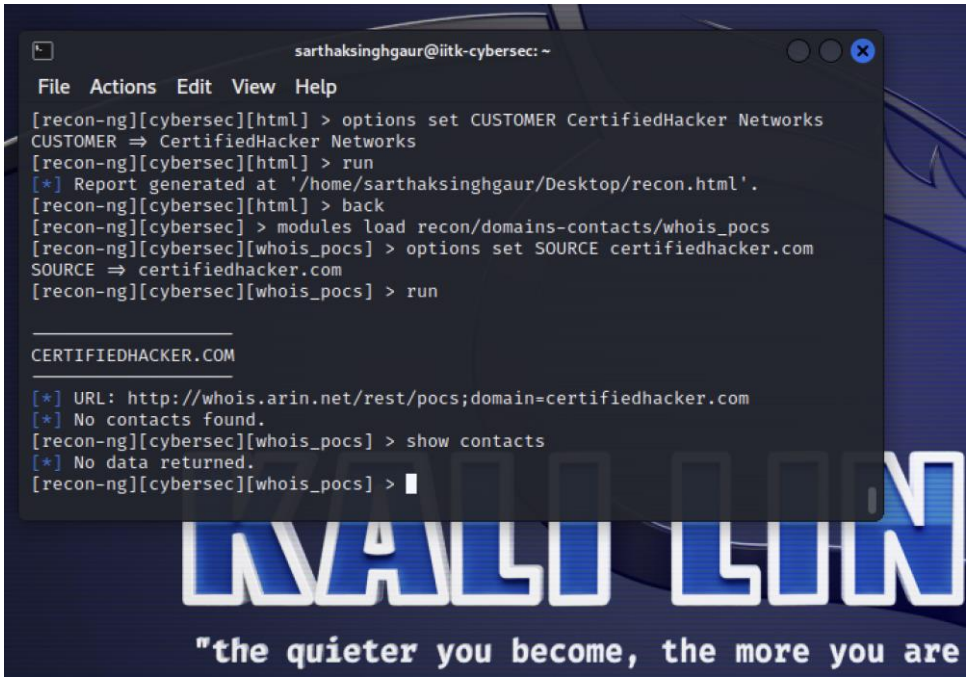
The whois command in Linux allows users to obtain this registration information by querying the WHOIS servers of various domain registries. It is a helpful tool for network administrators, webmasters, and security professionals who need to gather information about domain names or IP addresses for investigative or troubleshooting purposes.

The command will send a request to the appropriate WHOIS server, and the server will respond with the relevant registration details for that domain or IP address.

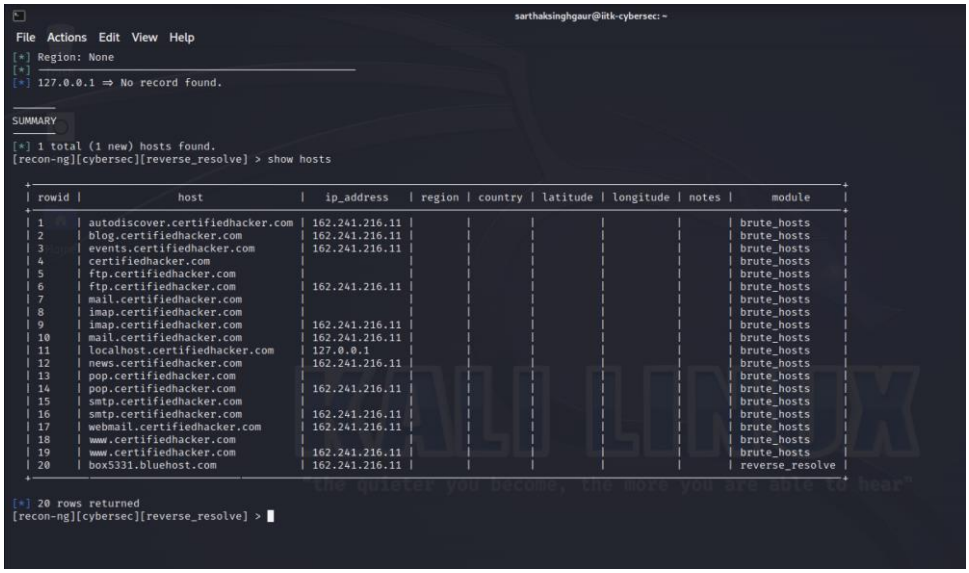
SUBDOMAINS SHARING SAME IP USING RECON-NG

Recon-ng is an open-source and powerful web reconnaissance framework designed for information gathering and OSINT (Open Source Intelligence) in penetration testing and ethical hacking. It is included in Kali Linux, a popular Linux distribution used by cybersecurity professionals and penetration testers, and can be installed on other Linux distributions as well.

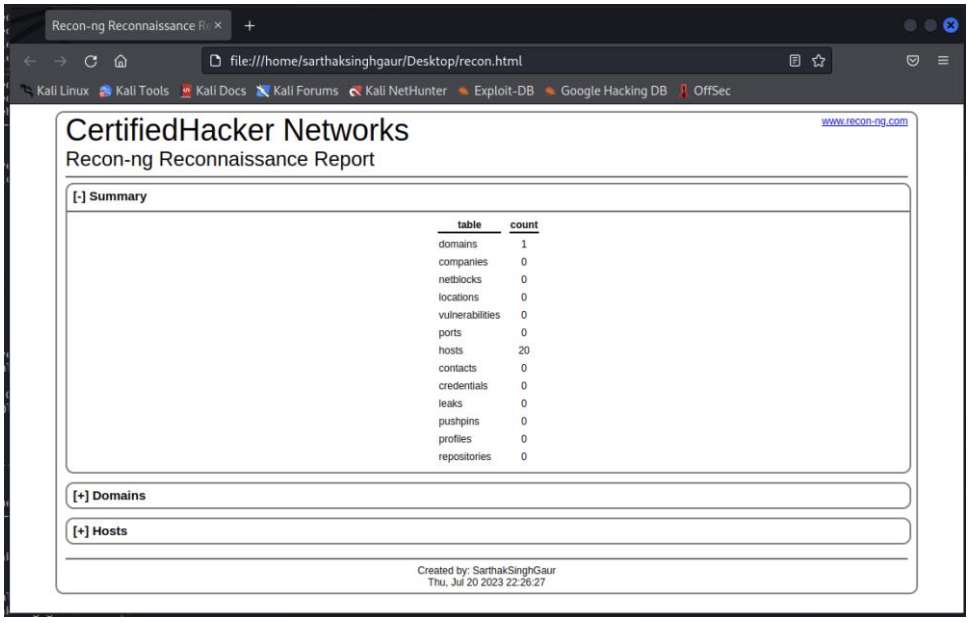




On using whois_pocs module, certifiedhacker doesn't reveal any contact information of its employees



We find that these many subdomains are hosted on the same IP address 162.241.216.11



We generate a report from recon-ng package for the same

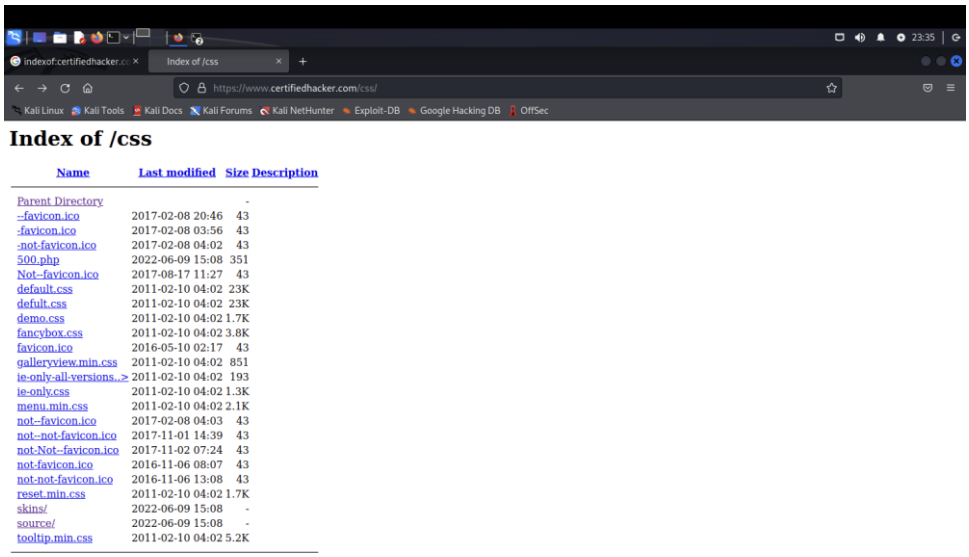
Recon-ng provides a wide range of modules for performing tasks like DNS enumeration, email gathering, social media profiling, and more. Its extensible nature allows users to add custom modules to suit their specific reconnaissance needs.

DIRECTORY LISTING USING GOOGLE DORKS

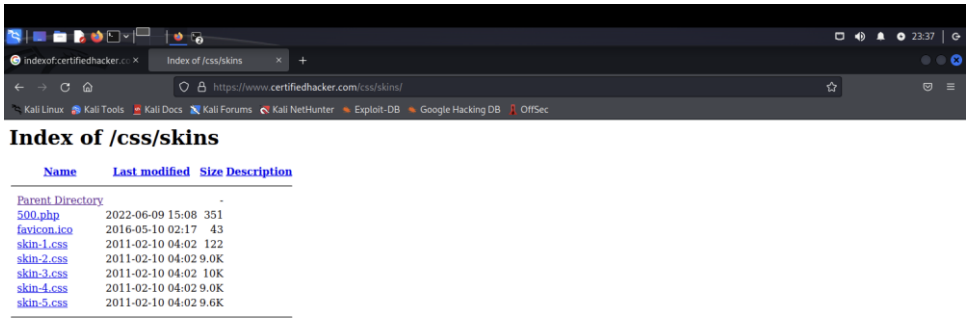
Google dorks are specialized search terms that can be used to identify sensitive information or vulnerabilities on websites and web applications. These search queries leverage the advanced search capabilities of Google to find specific information that may not be easily accessible through regular search queries.

"Google index of" is a specific Google dork that is used to find directory listings of websites containing various files and documents. When users enter "Google index of" followed by a directory name or file type, Google may return search results that reveal publicly accessible files and directories hosted on websites.

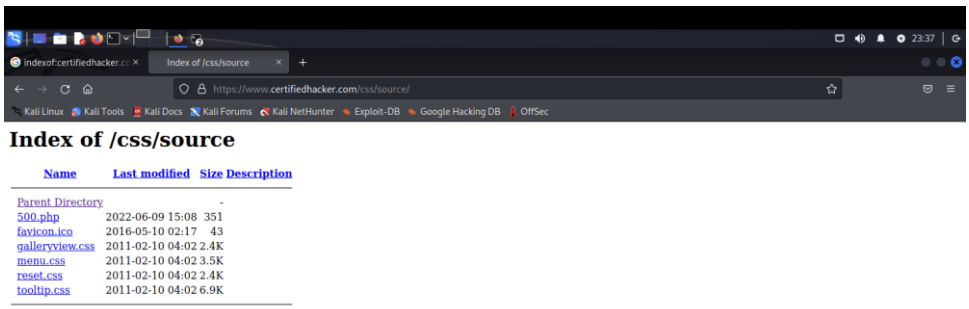
On Checking for Directory Listing, we could find that multiple directories are enabled:



https://www.certifiedhacker.com/css/



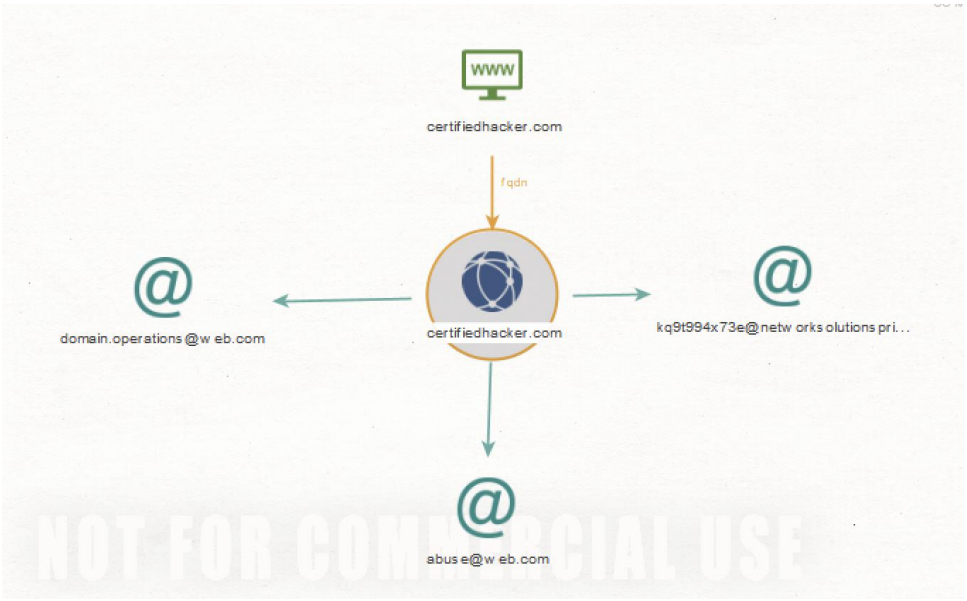
https://www.certifiedhacker.com/css/skins/



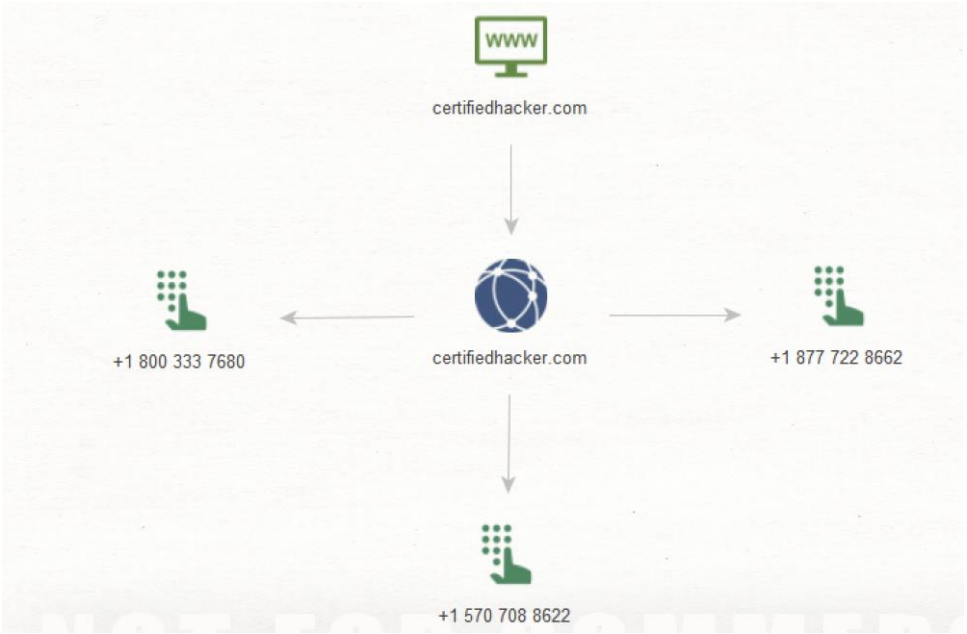
https://www.certifiedhacker.com/css/source/

BACKEND ANALYSIS USING MALTEGO

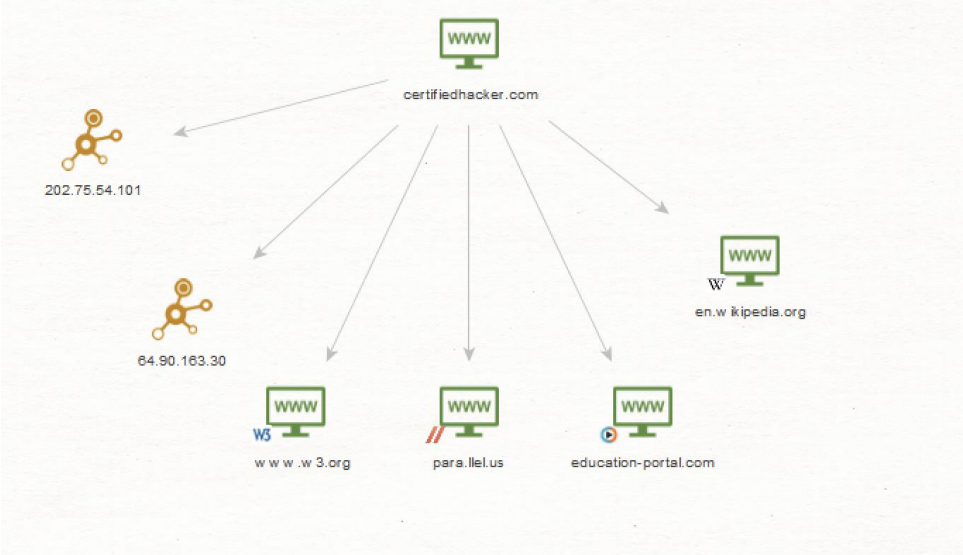
Maltego is a powerful open-source intelligence (OSINT) and data visualization tool used for information gathering and link analysis. It is widely used by cybersecurity professionals, intelligence analysts, and investigators to explore relationships between different entities and uncover connections in complex datasets.



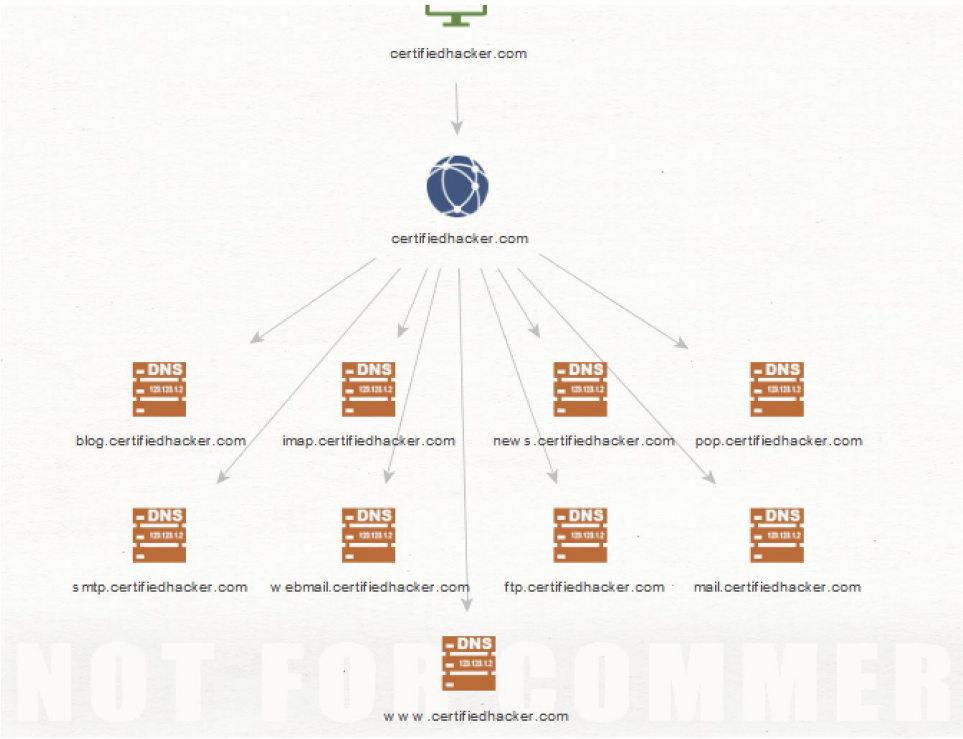
Email Addresses using admins



Phone Numbers of admins



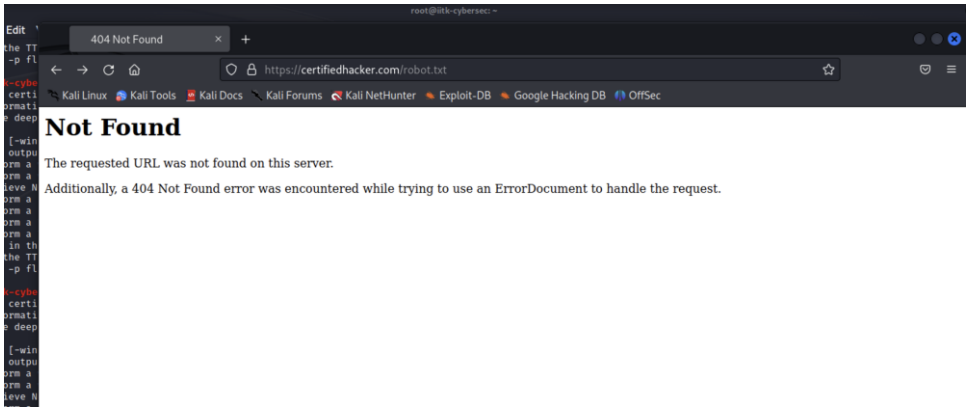
Connected external IP addresses



Sub domain entities using DNS Search

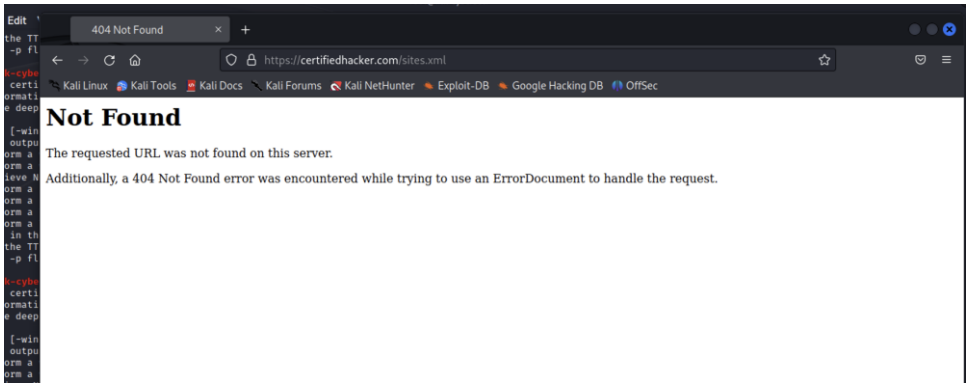
ROBOT.TXT AND SITES.XML

'robots.txt' is a plain text file that is often placed on the root directory of a website. It serves as a protocol for web crawlers or search engine bots, providing guidelines on which parts of the website should be crawled and indexed and which parts should not.



Robot.txt file is inaccessible

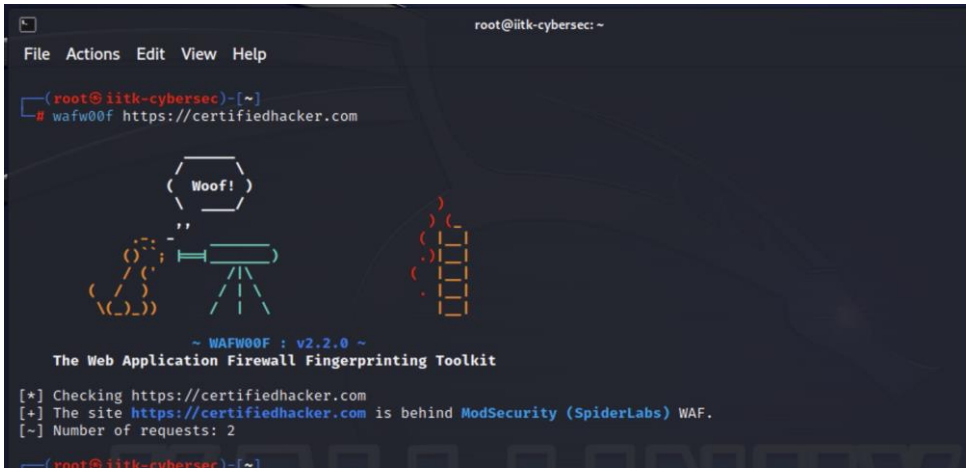
The use of another file called sites.xml is also used sometimes. 'sites.xml' is not a common practice, and it is not recognized as a standard protocol for controlling web crawler behavior or indexing by search engines.



Sites.xml file is inaccessible

FIREWALL STATUS USING WAFW00F

WAFW00F is a Python-based open-source tool used for fingerprinting and identifying web application firewalls (WAFs) and security appliances. It stands for "Web Application Firewall Detection Tool." WAFs are designed to protect web applications from various types of attacks, and WAFW00F helps in detecting the presence of these security measures on a website. Some WAFs may not be easily detectable, and false negatives or positives can occur.

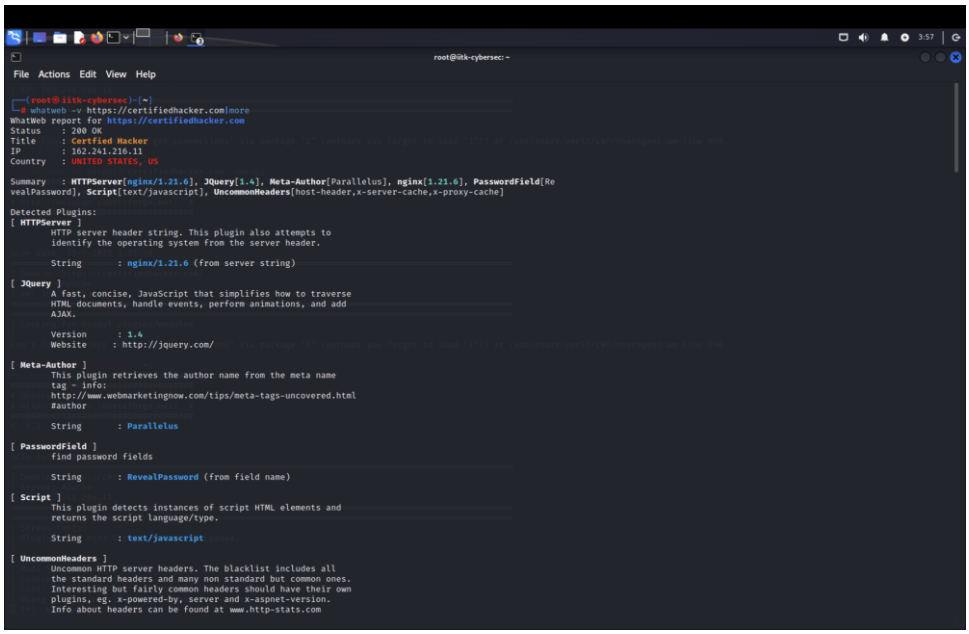


Firewall is up and the name of firewall is ModSecurity (SpiderLabs)

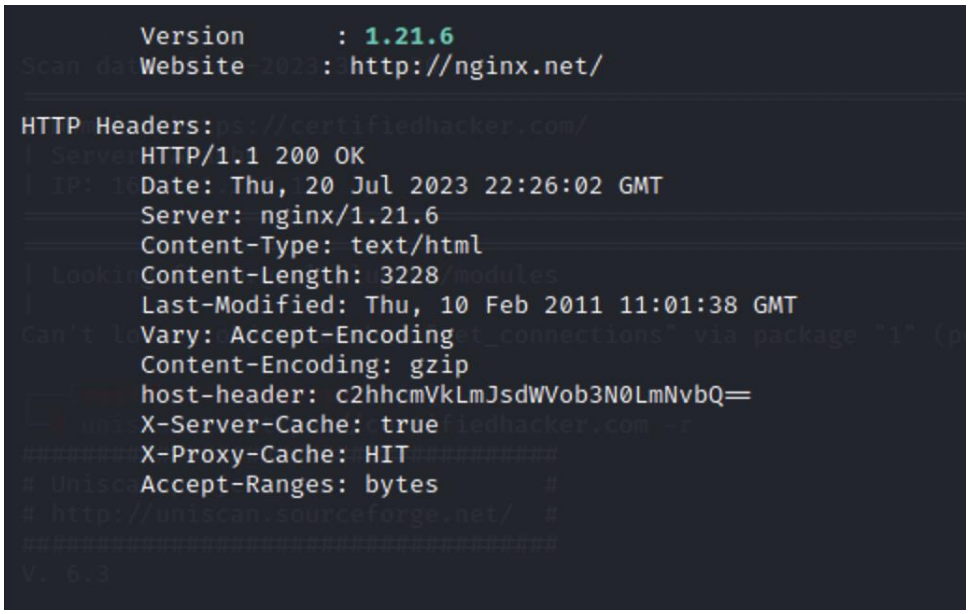
WAFW00F will send requests to the target URL and analyze the responses to determine if a web application firewall is in place. It will then display the results, indicating the identified WAF and its version (if available).

SERVER HEARDER AND SCRIPTS USING WHATWEB

In Kali Linux, WhatWeb is a powerful web application scanner and fingerprinting tool used for web reconnaissance and information gathering. It is designed to identify technologies, frameworks, and components used in web applications and provides valuable insights into the target's web stack.



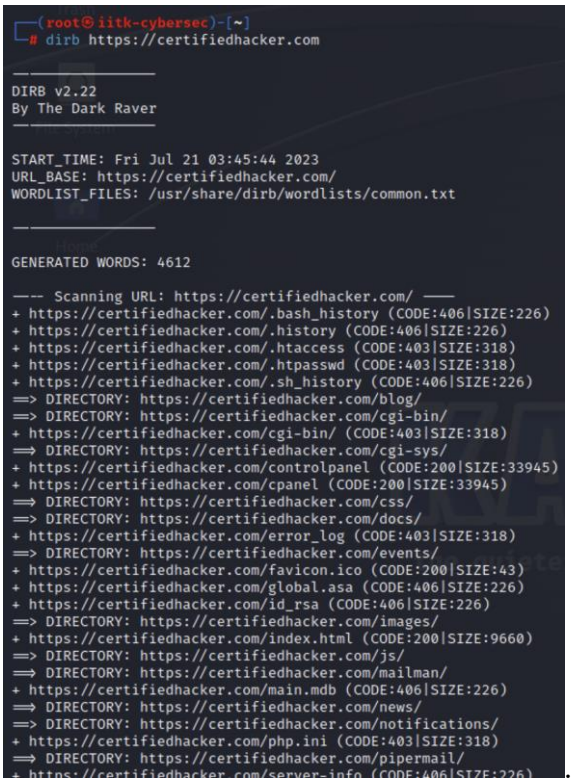
Using WhatWeb to reveal more information regarding server technologies and scripts used



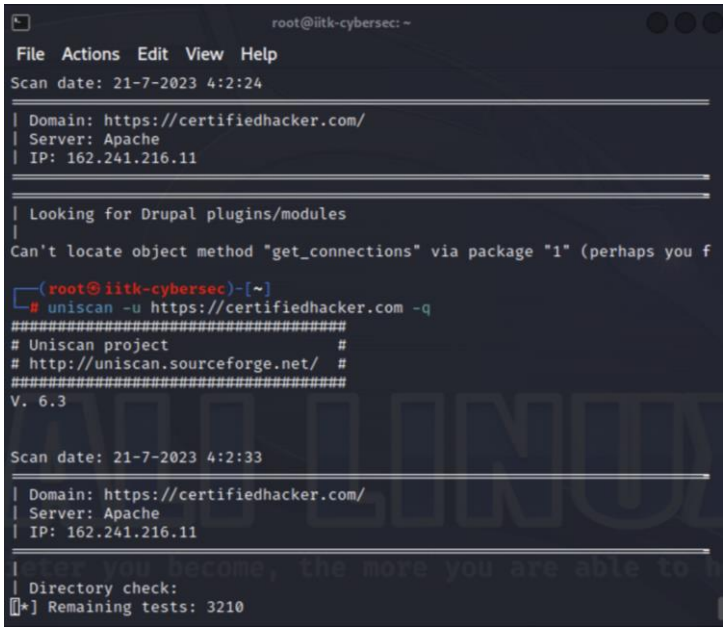
HTTP header details for the web server

DIRECTORY BRUTE-FORCING USING DIRB AND UNISCAN

In Kali Linux, DIRB and UNISCAN are popular command-line tools used for directory brute-forcing on web servers. It is designed to help penetration testers and security professionals discover hidden directories and files on a web server by performing dictionary-based attacks.



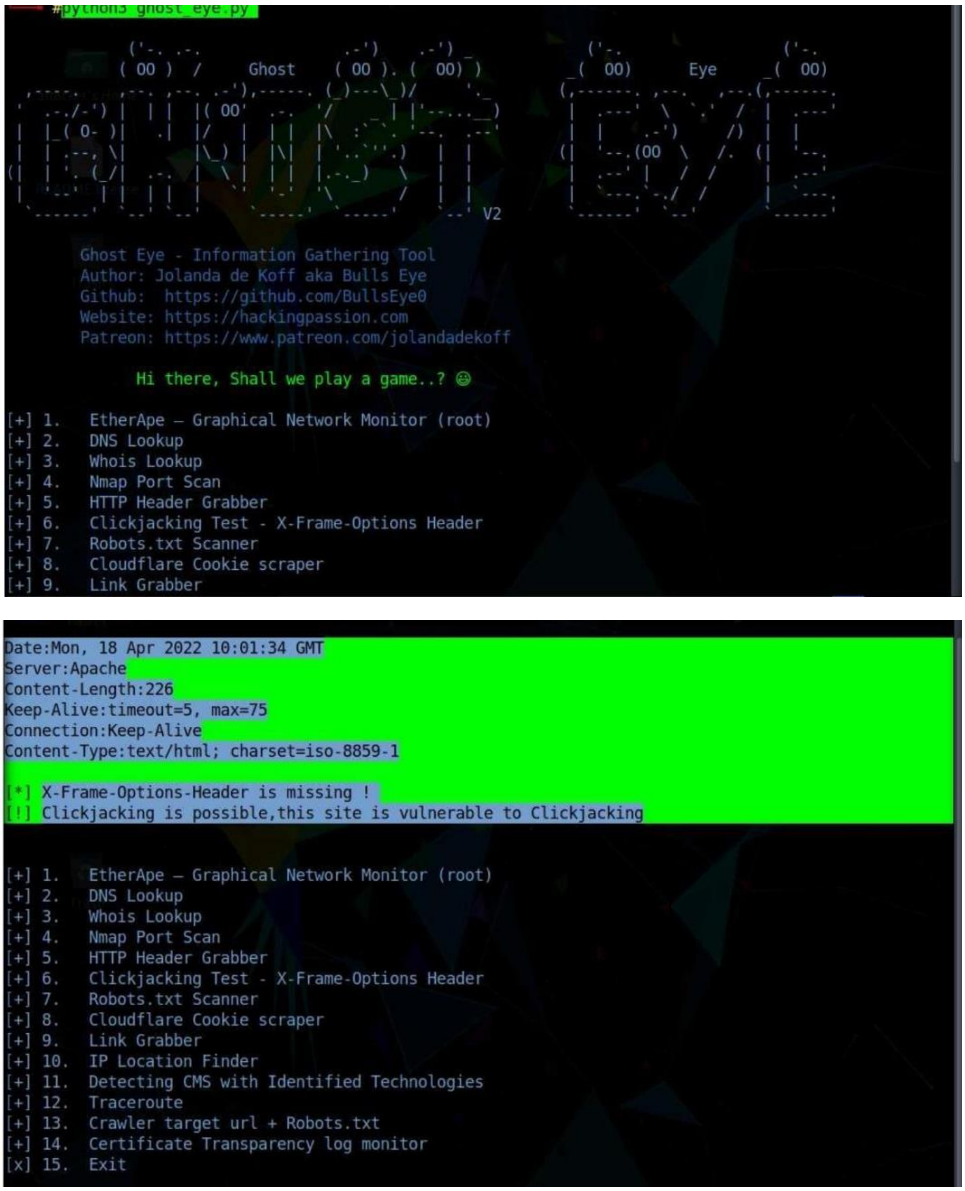
Using dirb to find open directories using a wordlist



Using uniscan to check directories in the webserver

VULNEARABILITY ANALYSIS USING GHOSTEYE

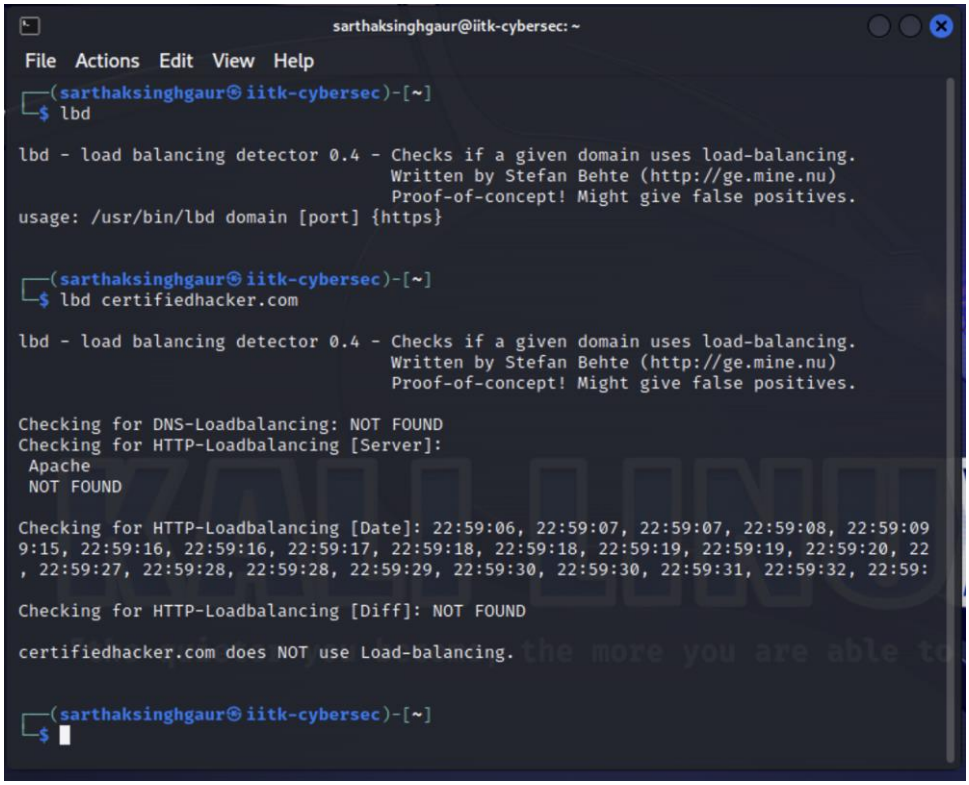
GhostEye is an information gathering, footprinting, scanner, and Reconnaissance tool built with Python 3. It captures information about the target and gives us detailed information about our objectives.



Using ghost_eye to check for clickjacking vulnerability

LOAD BALANCING INQUIRY USING LBD

Load balancers can be used to distribute traffic between multiple backend servers, making them an essential component of modern web applications and infrastructure. Identifying the presence of load balancers is crucial during security assessments and penetration testing, as they can introduce complexities in security configurations and require additional testing to ensure comprehensive coverage.

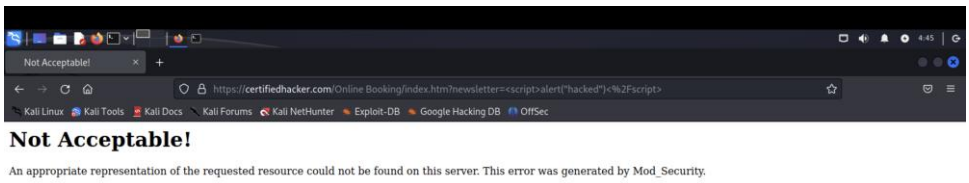


Checking for Load Balancing on web server using LBD

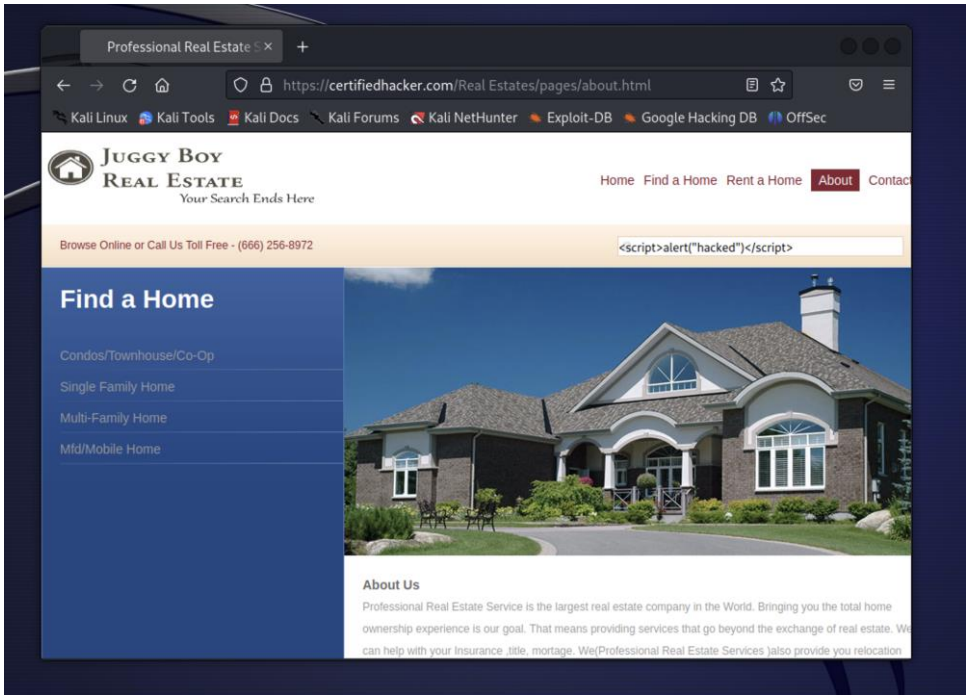
LBD will perform various tests to identify load balancing configurations and provide a report of its findings. LBD is a useful tool for identifying load balancers in web applications, which can be beneficial during security assessments and testing. LBD is a useful tool for identifying load balancers in web applications, which can be beneficial during security assessments and testing.

PARAMETER TAMPERING AND XSS VUNERABILITY SCANNING

Parameter tampering is a type of web application attack in penetration testing that involves modifying or manipulating parameters in web requests to gain unauthorized access, bypass security controls, or alter application behavior. It is a common attack vector used by malicious actors to exploit vulnerabilities in web applications and compromise their functionality.



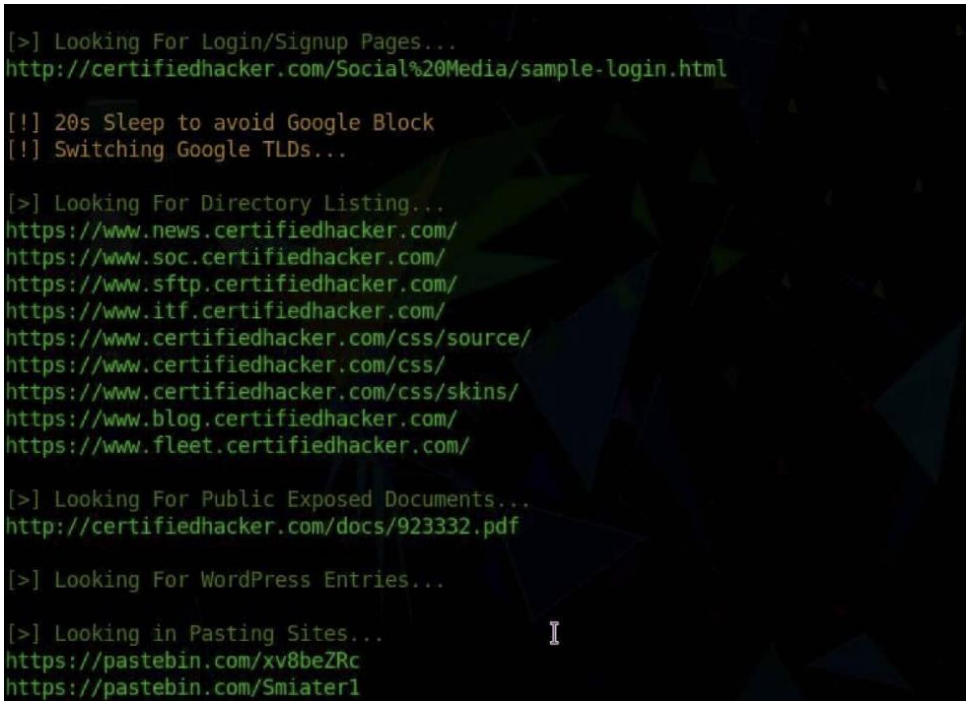
Firewall blocking Parameter Tampering Attack



Checking for XSS Vulnearabilities, which gets blocked at this page

PUBLICLY EXPOSED DATA USING GRECON

Google Dorking is the process of searching publicly available information on the internet in a more innovative way. The advanced query is fired to get the exact results of our doubt. GRecon tool is an automated script used in the process of Enumeration and Information Gathering. We can automate this task using the GRecon tool.

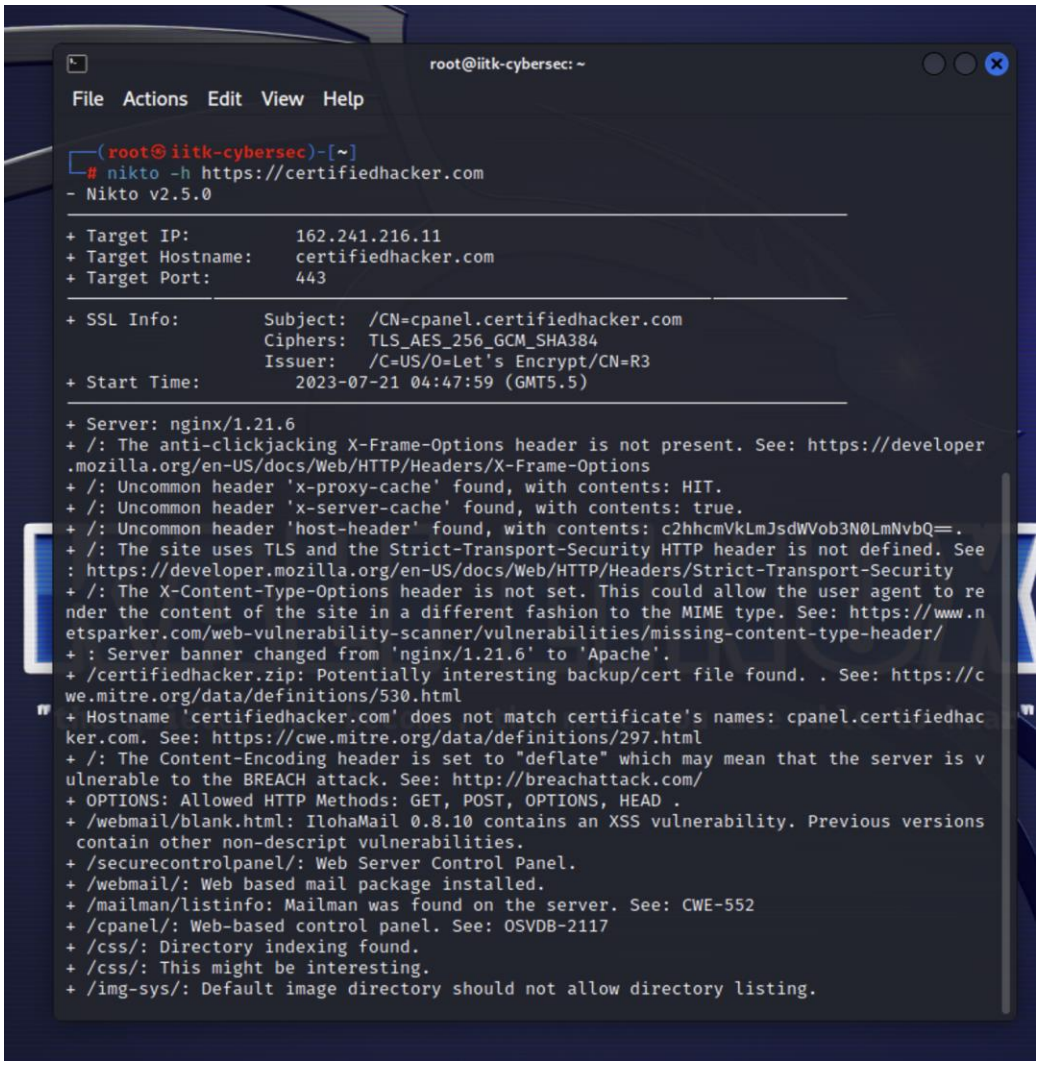


Finding external links hosted on the server

The method of Subdomain, Sub-Subdomain Enumeration can be done with the help of this tool; only you need to run the script and provide the target URL. GRecon is an open-source and free to use tool.

VULNEARABILITY ASSESSMENT USING NIKTO

Nikto is an open-source web server scanner and web application vulnerability scanner widely used in penetration testing and security assessments. It helps identify potential security issues, misconfigurations, and vulnerabilities in web servers and web applications.

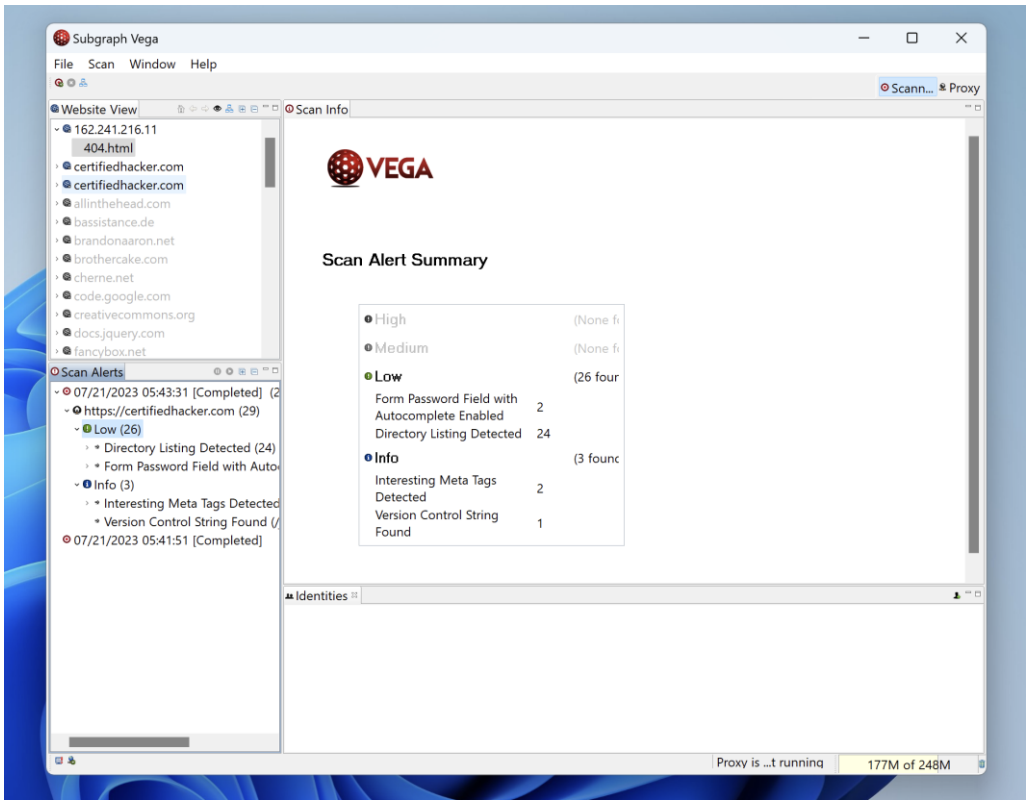


ClickJacking attack using nikto

Nikto uses a combination of methods, including signature-based scanning, static file analysis, and heuristic detection, to thoroughly assess the target web server and application. After scanning, Nikto generates a detailed report outlining the identified vulnerabilities and potential security risks. This report helps penetration testers and security professionals prioritize remediation efforts.

VULNEARABILITY ASSESSMENT USING VEGA

Vega is an open-source web application vulnerability scanner and testing platform used for identifying security issues and vulnerabilities in web applications. It is designed to be an extensible and user-friendly tool, making it popular among penetration testers, security professionals, and developers.

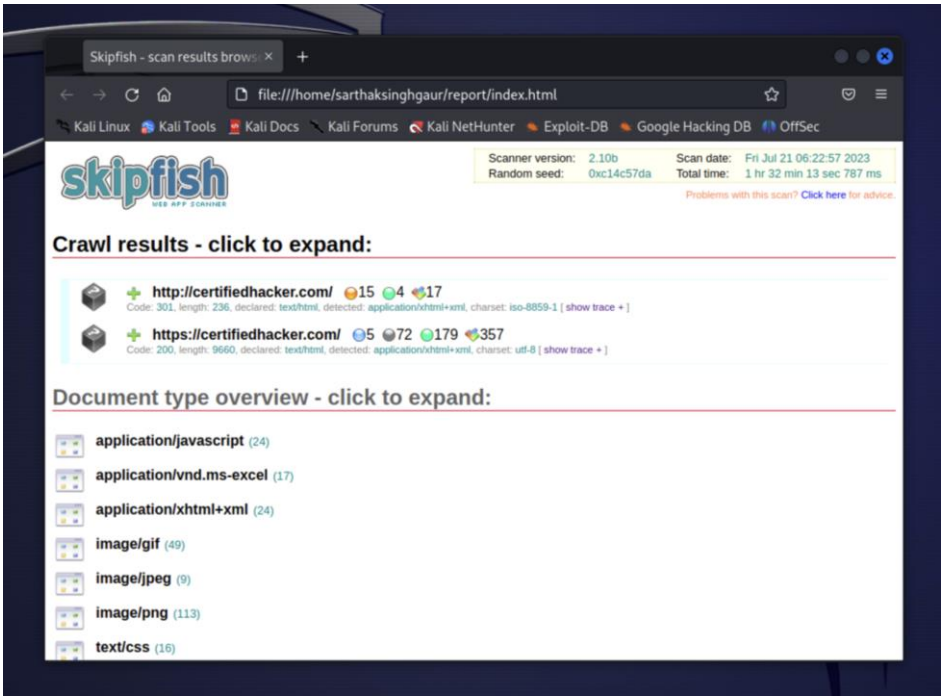


Directory Listing detection with Vega Vulnearability Scanner

Vega is built on top of the OWASP ZAP platform, leveraging its powerful scanning and testing capabilities. Vega's extensibility allows users to create custom modules and scripts to perform additional tests and extend the tool's functionality. Vega automatically crawls the web application to build a complete map of its structure, which helps identify hidden and interconnected pages.

VULNEARABILITY ASSESSMENT USING SKIPFISH

Skipfish is an open-source web application security scanner developed by Google. It is designed to perform fast and comprehensive security assessments of web applications by identifying potential vulnerabilities and security flaws.



Skipfish report

Skipfish uses a combination of crawling and analysis techniques to explore the target web application and identify potential security issues. The tool aims to provide broad coverage of potential security issues, including SQL injection, cross-site scripting (XSS), directory traversal, and more.

Skipfish supports incremental scanning, allowing testers to resume interrupted scans or focus on specific parts of the application. After completing the scan, Skipfish generates detailed reports with information about the identified vulnerabilities and their severity levels. The tool provides output in multiple formats, making it easier for users to analyze and share the results.

XSS vector in document body (15)

1. http://certifiedhacker.com/ [show trace +]
   Memo: injected '<sfi...>' tag seen in HTML
2. http://certifiedhacker.com/corporate-learning-website [show trace +]
   Memo: injected '<sfi...>' tag seen in HTML
3. http://certifiedhacker.com/css/500.php [show trace +]
   Memo: injected '<sfi...>' tag seen in HTML
4. http://certifiedhacker.com/css/skins [show trace +]
   Memo: injected '<sfi...>' tag seen in HTML
5. http://certifiedhacker.com/images [show trace +]
   Memo: injected '<sfi...>' tag seen in HTML
6. http://certifiedhacker.com/images/content [show trace +]
   Memo: injected '<sfi...>' tag seen in HTML
7. http://certifiedhacker.com/images/content/demo-only [show trace +]
   Memo: injected '<sfi...>' tag seen in HTML
8. http://certifiedhacker.com/images/icons [show trace +]
   Memo: injected '<sfi...>' tag seen in HTML
9. http://certifiedhacker.com/images/skins [show trace +]
   Memo: injected '<sfi...>' tag seen in HTML
10. http://certifiedhacker.com/images/skins/skin-2 [show trace +]
    Memo: injected '<sfi...>' tag seen in HTML
11. http://certifiedhacker.com/images/slideshow [show trace +]
    Memo: injected '<sfi...>' tag seen in HTML
12. http://certifiedhacker.com/js/galleryview [show trace +]
    Memo: injected '<sfi...>' tag seen in HTML
13. http://certifiedhacker.com/js/galleryview/themes [show trace +]
    Memo: injected '<sfi...>' tag seen in HTML
14. http://certifiedhacker.com/Online%20Booking [show trace +]
    Memo: injected '<sfi...>' tag seen in HTML
15. http://certifiedhacker.com/P-folio [show trace +]
    Memo: injected '<sfi...>' tag seen in HTML

XSS vector detected in skipfish

- **HTML form with no apparent XSRF protection** (1)
- **SSL certificate host name mismatch** (4)
- **Response varies randomly, skipping checks** (6)
- **IPS filtering enabled** (39)
- **Limits exceeded, fetch suppressed** (10)
- **Resource fetch failed** (17)
- **Numerical filename - consider enumerating** (18)
- **Incorrect or missing charset (low risk)** (46)
- **Incorrect or missing MIME type (low risk)** (20)
- **Password entry form - consider brute-force** (1)
- **HTML form (not classified otherwise)** (1)
- **Hidden files / directories** (2)
- **Directory listing enabled** (90)
- **New 404 signature seen** (2)
- **New 'X-*' header value seen** (2)
- **New 'Server' header value seen** (2)
- **SSL certificate issuer information** (1)

Vunnerability overview in skipfish report

RECOMMENDATIONS

The penetration test conducted on the "CertifiedHacker Networks" web server and e-commerce application revealed critical vulnerabilities, including an exploitable Cross-Site Scripting (XSS) flaw. Below are the expanded recommendations to address these vulnerabilities and enhance the overall security posture of the system:

CROSS-SITE SCRIPTING (XSS)

FINDING:

Multiple XSS vulnerabilities were discovered in the application's input fields, potentially enabling attackers to execute malicious scripts within users' browsers.

RECOMMENDATIONS:

Input Validation and Output Encoding: Implement strict input validation and output encoding for all user-supplied data to prevent the injection of malicious scripts.

Sanitization of User-Generated Content: Sanitize and filter user-generated content to remove or escape any potentially malicious scripts before displaying them on web pages.

Content Security Policy (CSP): Implement a Content Security Policy (CSP) that restricts the sources from which resources can be loaded, reducing the risk of XSS attacks.

HTTP-Only and Secure Flags: Set the "HttpOnly" and "Secure" flags on cookies to prevent client-side script access and ensure cookies are only transmitted over secure HTTPS connections.

X-XSS-Protection Header: Enable the X-XSS-Protection header in HTTP responses to instruct modern browsers to block or sanitize suspicious requests.

Security Awareness Training: Conduct security awareness training for developers and other personnel to educate them about the risks of XSS and safe coding practices.

MISSING SECURITY HEADERS

FINDING:

The web server lacks crucial security headers, increasing the risk of various attacks, such as XSS and Clickjacking.

RECOMMENDATIONS:

HTTP Strict Transport Security (HSTS): Enable HSTS to enforce secure HTTPS connections, reducing the risk of man-in-the-middle attacks.

X-Content-Type-Options: Set the X-Content-Type-Options header to prevent MIME-type sniffing, reducing the risk of content-based attacks.

Frame Protection Headers: Implement X-Frame-Options or Content-Security-Policy frame-ancestors to prevent Clickjacking attacks.

DIRECTORY TRAVERSAL

FINDING:

A directory traversal vulnerability was identified in the e-commerce application, allowing attackers to access files and directories outside the intended scope.

RECOMMENDATIONS:

Input Validation and Sanitization: Implement strict input validation and sanitization for all user-supplied data, particularly those used to construct file paths or URLs.

File Path Whitelisting: Create a whitelist of permissible file paths and validate user input against this list to prevent unauthorized access to sensitive directories.

Use Safe File Access Methods: Instead of relying on user-supplied input to construct file paths, use safe file access methods provided by the programming language or framework being used.

Restrict Access Permissions: Ensure that the web server process has minimal permissions necessary to access only required files and directories.

CONCLUSION

Participating in this hypothetical penetration testing project has been extremely beneficial to me as it has provided valuable hands-on experience and exposure to real-world cybersecurity challenges. The project allowed me to:

Develop Practical Skills: Through conducting vulnerability assessments and penetration tests, I gained practical skills in using various security tools and methodologies to identify and exploit vulnerabilities in web servers and applications.

Enhance Technical Knowledge: I deepened my understanding of web application security, network scanning, and the intricacies of common vulnerabilities such as SQL injection, XSS, and authentication weaknesses.

Problem-Solving and Critical Thinking: The project required me to think critically and creatively to identify potential attack vectors and devise appropriate mitigation strategies, thus honing my problem-solving skills.

Communication and Reporting: Preparing the project report allowed me to refine my communication skills by effectively documenting technical findings, vulnerabilities, and recommended remediation steps.

Professional Development: Engaging in this project showcases my commitment to cybersecurity and willingness to contribute to improving the security posture of organizations, which can be advantageous for future career opportunities in the field.

Understanding Business Impact: By identifying critical vulnerabilities, I gained insight into the potential risks posed to an organization's reputation, finances, and customer trust. This understanding will help me prioritize security efforts based on the potential impact.

Ethical Hacking Experience: As an ethical hacker, I learned the importance of responsible disclosure and ethical conduct while simulating attacks to identify vulnerabilities. This experience reinforced the significance of adhering to ethical principles in the field of cybersecurity.

Overall, this project has been a valuable learning experience that has enriched my skill set and knowledge in cybersecurity. The practical exposure gained during the penetration testing process has prepared me to tackle real-world security challenges and contribute to creating a safer digital environment for individuals and organizations alike.