

Task-1

TOP 10 HACKERS AND DESCRIPTION ABOUT THEM :

1. Kevin Mitnick

- Status: Former black hat hacker turned white hat hacker.
- Actions: Infamous for hacking into various computer systems, including IBM and Nokia. He was later arrested, served time in prison, and became a cybersecurity consultant. Mitnick now helps organizations strengthen their security.

2. Julian Assange

- Status: Founder of WikiLeaks.
- Actions: Assange is known for publishing classified government documents on WikiLeaks. His actions have been a subject of debate, with some considering him a grey hat figure who promotes transparency, while others view him as a threat to national security.

3. Edward Snowden

- Status: Former CIA employee and NSA contractor.
- Actions: Snowden leaked a trove of classified NSA documents to journalists, exposing extensive government surveillance programs. He is widely regarded as a whistleblower who ignited a global debate on privacy and government overreach.

4. Kevin Poulsen

- Status: Former black hat hacker turned white hat hacker.
- Actions: Poulsen, known as the "Dark Dante," engaged in various hacking activities, including taking over phone lines for radio contests. After serving time in prison, he became a journalist and cybersecurity expert, focusing on online security issues.

5. Adrian Lamo

- Status: Grey hat hacker and journalist.

- Actions: Lamo reported Chelsea Manning's leak of classified documents to authorities, leading to Manning's arrest. His actions were controversial, and he later worked as a journalist, covering technology and security topics.

6. Chelsea Manning

- Status: Former U.S. Army intelligence analyst.
- Actions: Manning leaked a vast amount of classified documents to WikiLeaks, exposing government misconduct and sparking debates on transparency and national security. Manning is considered a whistleblower.

7. Gary McKinnon

- Status: Former black hat hacker.
- Actions: McKinnon hacked into U.S. government computers, searching for evidence of UFOs and government cover-ups. His actions led to extradition attempts by the U.S. but were ultimately blocked by the UK government.

8. Robert Tappan Morris

- Status: Computer scientist and entrepreneur.
- Actions: Morris created the Morris Worm, one of the first computer worms to spread across the internet. After facing legal consequences, he went on to contribute positively to the tech industry as an entrepreneur and professor.

9. Marcus Hutchins (MalwareTech)

- Status: Security researcher and white hat hacker.
- Actions: Hutchins gained recognition for his role in stopping the WannaCry ransomware attack. He is dedicated to cybersecurity and works to protect organizations from cyber threats.

10. Jonathan James

- Status: Former black hat hacker.
- Actions: James committed various cybercrimes, including hacking into NASA and stealing sensitive data. Tragically, he took his own life at a young age after facing legal consequences for his actions.

Please note that these labels are based on historical actions and may not fully capture the complexity of each individual's motivations or contributions to the field of computer security.

Task-2

FIND THE VULNERABILTIES PERFORMED ON THE DIFFERENT PORTS :

1.Port no. (20,21)

port number 20 ,21 is commonly associated with ftp(file transfer protocol).

FTP stands for File Transfer Protocol. Port 20 and 21 are solely TCP ports used to allow users to send and to receive files from a server to their personal computers.

The FTP port is insecure and outdated and can be exploited using:

- Anonymous authentication. You can log into the FTP port with both username and password set to "anonymous".
- Cross-Site Scripting.
- Brute-forcing passwords.
- Directory traversal attacks.

2.(port no.22) SSH (22)

SSH stands for Secure Shell. It is a TCP port used to ensure secure remote access to servers. You can exploit the SSH port by brute-forcing SSH credentials or using a private key to gain access to the target system.

3.port no.23[Telnet (23)]

The Telnet protocol is a TCP protocol that enables a user to connect to remote computers over the internet. The Telnet port has long been replaced by SSH, but it is still used by some websites today. It is outdated, insecure, and vulnerable to malware. Telnet is vulnerable to spoofing, credential sniffing, and credential brute-forcing.

4. SMTP (25)

SMTP stands for Simple Mail Transfer Protocol. It is a TCP port used for sending and receiving mails. It can be vulnerable to mail spamming and spoofing if not well-secured

5. DNS (53)

DNS stands for Domain Name System. It is both a TCP and UDP port used for transfers and queries respectively. One common exploit on the DNS ports is the Distributed Denial of Service (DDoS) attack.

6. TFTP (69)

TFTP stands for Trivial File Transfer Protocol. It's a UDP port used to send and receive files between a user and a server over a network. TFTP is a simplified version of the file transfer protocol. Because it is a UDP port, it does not require authentication, which makes it faster yet less secure.

7.port no.80 HTTP / HTTPS (443, 80, 8080, 8443)

HTTP stands for HyperText Transfer Protocol, while HTTPS stands for HyperText Transfer Protocol Secure (secure than HTTP). These are the most popular and widely used protocols on the internet, and as such are prone to many vulnerabilities. They are vulnerable to SQL injections, cross-site scripting, cross-site request forgery, etc

8.Port no.110 [post office protocol]:-

Port 110 is associated with the Post Office Protocol version 3 (POP3), which is used for receiving email messages from a mail server to a client device. This protocol is quite old and lacks encryption by default, making it vulnerable to several security issues. Here are a couple of vulnerabilities associated with port 110:

1. **Plain Text Transmission:** POP3 was designed without encryption in mind, meaning that usernames and passwords are sent in plain text. This makes it susceptible to eavesdropping attacks, where malicious actors can intercept the data being transmitted and gain access to login credentials.
2. **Brute Force Attacks:** Since the authentication process in POP3 involves sending the username and password in clear text, attackers can attempt to brute force login credentials with relative ease. Without proper security measures, this can lead to unauthorized access.
3. **Credential Harvesting:** Attackers can exploit vulnerabilities in the mail client or server software to harvest usernames and passwords, potentially leading to unauthorized access not only to email accounts but also to other services if the same credentials are reused.
4. **Lack of Encryption:** The lack of encryption means that messages and attachments transferred between the server and the client can be intercepted and read by anyone with access to the network traffic.

To mitigate these vulnerabilities, it's recommended to:

- Use secure variants of email protocols such as POP3 over TLS (POP3S) or Internet Message Access Protocol (IMAPS) over TLS, which encrypt the communication between the client and the server.
- Implement strong, unique passwords for email accounts.
- Regularly update email client software to ensure security patches are applied.
- Employ network security measures like firewalls and intrusion detection systems to detect and prevent unauthorized access.

- Educate users about the risks of plain text authentication and encourage them to use secure alternatives.

9.Port 123 [Network time protocol] :

Port 123 is used for the Network Time Protocol (NTP), which is used for synchronizing clocks on computer systems. Some vulnerabilities associated with NTP include:

1. Amplification Attacks: NTP servers can be exploited in amplification attacks where attackers send small requests with a forged source IP address to NTP servers. The servers then respond with larger amounts of data to the spoofed IP address, potentially causing DDoS attacks.

2. Reflection Attacks: Similar to amplification attacks, reflection attacks involve sending requests to NTP servers, which then reply to the target IP address with a larger response. This can be used to overload target systems.

3. Denial of Service (DoS): NTP vulnerabilities can be exploited to launch DoS attacks against servers by sending specially crafted packets that consume resources and disrupt normal functionality.

To mitigate these vulnerabilities, network administrators should:

- Implement access control and filtering mechanisms to prevent unauthorized access to NTP servers.
- Regularly update NTP software to patch known vulnerabilities.
- Use rate limiting and other techniques to prevent amplification and reflection attacks.
- Monitor NTP server traffic for unusual patterns.

9.Port 143[internet message protocol]:

Port 143 is associated with the Internet Message Access Protocol (IMAP), used for receiving email messages from a mail server to a client device. Some vulnerabilities associated with IMAP include:

1. **Brute Force Attacks:** Attackers can attempt to brute force login credentials due to the lack of encryption during the authentication process in standard IMAP.

2. **Data Exposure:** Similar to POP3, without encryption, messages and attachments transferred between the server and client can be intercepted and read by malicious actors.

3. **Email Spoofing and Phishing:** Vulnerabilities in email clients or servers can lead to email spoofing, where attackers send emails that appear to be from a legitimate source to deceive recipients into taking harmful actions.

To enhance security:

- Use IMAPS (IMAP over TLS) to encrypt communication between clients and servers.
- Ensure email clients and servers are updated with the latest security patches.
- Educate users about email security, phishing, and the importance of strong passwords.

11.Port 443:

Port 443 is used for secure communications over the HTTPS protocol, often associated with web browsing and encrypted data transfer. While this port is generally considered secure due to the use of SSL/TLS encryption, vulnerabilities can still arise:

1. **TLS Vulnerabilities:** Weaknesses in the SSL/TLS protocols can lead to vulnerabilities such as Heartbleed, POODLE, and others that compromise the encryption and expose sensitive data.

2. Certificate Vulnerabilities: Expired, misconfigured, or compromised SSL certificates can lead to security risks.

3. Man-in-the-Middle Attacks: While SSL/TLS is designed to prevent this, misconfigurations or compromised certificates can enable attackers to intercept and modify data.

To ensure security:

- Use strong encryption protocols and keep them updated.
- Regularly renew and verify SSL certificates.
- Stay informed about emerging SSL/TLS vulnerabilities and apply patches promptly.

Task-3

TOP 5 OWASP CWE DESCRIPTION WITH BUSINESS IMPACT

1. CWE-20: Improper Input Validation

Description:

CWE-20, known as "Improper Input Validation," occurs when an application fails to properly validate user-supplied input. This weakness can lead to various security issues, including SQL injection, cross-site scripting (XSS), and remote code execution.

Business Impact:

For businesses, CWE-20 poses significant risks. An attacker can exploit this weakness to steal sensitive customer data, manipulate application functionality, and even take control of the application. This can result in data breaches, loss of customer trust, financial penalties, and damage to the brand's reputation.

2. CWE-79: Improper Neutralization of Input During Web Page Generation ('Cross-site Scripting')

Description:

CWE-79, or "Cross-Site Scripting (XSS)," occurs when an application includes untrusted data in a web page without proper validation or escaping. Attackers can then inject malicious scripts into the web pages viewed by other users, compromising their accounts and data.

Business Impact:

XSS vulnerabilities can lead to severe business consequences. Exploited XSS flaws can allow attackers to steal sensitive user information, such as login credentials and financial data. Additionally, they can deface websites, distribute malware, and tarnish a company's image. Customers may lose trust in the organization's online services.

3. CWE-306: Missing Authentication for Critical Function

Description:

CWE-306, "Missing Authentication for Critical Function," occurs when an application fails to require authentication for critical operations or functionalities. This oversight enables unauthorized users to access sensitive functions, posing significant security risks.

Business Impact:

The business impact of CWE-306 can be devastating. Unauthorized access to critical functions can result in data breaches, financial losses, and legal consequences. Customers' confidential information may be exposed, leading to a loss of trust and reputation damage.

4. CWE-89: SQL Injection

Description:

CWE-89, known as "SQL Injection," arises when an application accepts untrusted data and directly incorporates it into SQL queries without proper validation. Attackers can exploit this weakness to manipulate the database, extract sensitive data, or even delete records.

Business Impact:

SQL Injection can have severe business ramifications. An attacker with successful SQL injection can access, modify, or delete critical data, leading to data breaches, financial losses, and potential legal actions. Additionally, it can harm the organization's reputation and customer trust.

5. CWE-312: Cleartext Storage of Sensitive Information

Description:

CWE-312, "Cleartext Storage of Sensitive Information," occurs when an application stores sensitive data, such as passwords or encryption keys, in an insecure manner, such as in plaintext. This makes it easier for attackers to obtain this information.

Business Impact:

Storing sensitive data in cleartext can result in data exposure and breaches. Attackers can exploit this weakness to gain unauthorized access to sensitive accounts or information, leading to financial losses, regulatory penalties, and damage to the company's reputation.

Task-4

In the intricate world of web servers, vulnerabilities can be exploited by various attacks, each with its unique modus operandi and potentially catastrophic consequences. Let's navigate through ten such assailants that cunningly prey on the cracks in web server defenses.

1. SQL Injection (SQLi):

Picture this: a seemingly innocent search bar on a website. Unbeknownst to users, hackers deftly slip malicious SQL queries into it. When inadequately validated by the server, these queries can compromise databases, granting unauthorized access to sensitive information.

2. Cross-Site Scripting (XSS):

Imagine a cyber-illusionist conjuring a malicious script within a comment box. As unsuspecting users read the comments, this script springs to life within their browsers, stealing credentials or even redirecting them to a fake website.

3. Cross-Site Request Forgery (CSRF):

Visualize a situation where an authenticated user, logged into a shopping site, unknowingly clicks a link that triggers a purchase on their behalf. Crafty attackers manipulate this active session to perform actions without the user's knowledge.

4. Distributed Denial of Service (DDoS):

Envision an army of virtual zombies relentlessly bombarding a web server with an overwhelming torrent of requests. This digital onslaught incapacitates the server, rendering it inaccessible to genuine users.

5. Server-Side Request Forgery (SSRF):

Picture an attacker exploiting vulnerable server-side code, tricking the server into making unauthorized requests to other systems. This can lead to breaches of sensitive data or even launching attacks on internal networks.

6. XML External Entity (XXE) Attack:

Imagine a scenario where a misconfigured XML parser unknowingly gobbles up malicious input. The attacker cleverly injects external entities, opening pathways for information leakage or crippling the server.

7. Remote File Inclusion (RFI) and Local File Inclusion (LFI):

Visualize hackers manipulating file paths in a web application. By forcing the server to include external files, they can gain unauthorized access, execute arbitrary code, or even compromise the entire system.

8. Directory Traversal:

Imagine an attacker exploiting lax input validation to navigate directories beyond intended boundaries. By ascending paths, they can gain unauthorized access to files, unearthing sensitive information.

9. HTTP Response Splitting:

Envision a stealthy attacker injecting malicious input containing newlines. When the server responds, these newlines divide responses into two parts. This manipulation can lead to browser-based attacks or cache poisoning.

10. Brute Force Attacks: Picture an assailant relentlessly trying various username and password combinations, like a persistent locksmith attempting every key. This method preys on weak credentials or defaults, ultimately breaching server defenses.

To fortify against these crafty adversaries:

- Regularly update web server software and applications to patch vulnerabilities.
- Implement rigorous input validation and output encoding to thwart injection attacks.
- Enforce robust authentication mechanisms and stringent password policies.
- Configure vigilant firewalls and intrusion detection systems to detect and repel attacks.
- Safeguard data in transit with SSL/TLS encryption.
- Employ the guardianship of web application firewalls (WAFs) to filter and scrutinize incoming traffic.
- Foster a culture of secure coding practices to weave a formidable web of defense.

Remember, the digital realm is an ever-evolving landscape, demanding unwavering vigilance. Regular assessments, continuous learning, and swift adaptation are the sentinels of a resilient web server fortress.

Task-5

EXPLAIN ANY 10 WEB SERVER ATTACKS

1. SQL Injection (SQLi):

- Description: SQL Injection, often referred to as SQLi, is an attack where malicious SQL queries are injected into input fields or URL parameters. This exploits vulnerabilities in insufficient input validation, potentially leading to unauthorized access, data theft, or database manipulation.

2. Cross-Site Scripting (XSS):

- Description: Cross-Site Scripting (XSS) attacks involve the injection of malicious scripts into web pages. When other users view these pages, their browsers execute the injected scripts, enabling attackers to steal cookies, hijack sessions, or deface websites.

3. Cross-Site Request Forgery (CSRF):

- Description: CSRF attacks deceive authenticated users into unknowingly executing unwanted actions on a different site. Exploiting trust between users and vulnerable websites, attackers can manipulate settings, make unauthorized transactions, or perform actions on behalf of victims.

4. Distributed Denial of Service (DDoS):

- Description: Distributed Denial of Service (DDoS) attacks overwhelm web servers with a massive volume of traffic from various sources, rendering them unresponsive. This results in downtime, financial losses, and a poor user experience.

5. Server-Side Request Forgery (SSRF):

- Description: Server-Side Request Forgery (SSRF) attacks trick web applications into making requests to internal or external resources. Attackers can gain unauthorized access, retrieve sensitive data, or bypass firewalls using SSRF.

6. Directory Traversal:

- Description: Directory Traversal, also known as Path Traversal, permits attackers to access files and directories outside a web server's root directory by manipulating input. This can lead to unauthorized access to sensitive files.

7. Remote File Inclusion (RFI):

- Description: Remote File Inclusion (RFI) attacks involve including external files through a web server, potentially leading to arbitrary code execution. Attackers can compromise servers and execute malicious code.

8. Brute Force Attack:

- Description: Brute Force Attacks involve repeated attempts to guess usernames and passwords until the correct combination is found. Successful attacks can result in unauthorized access to user accounts or the web server itself.

9. Zero-Day Exploits:

- Description: Zero-Day Exploits target vulnerabilities in web server software or plugins that are unknown to the software vendor, allowing attackers to exploit these vulnerabilities before patches become available.

10. HTTP Response Splitting:

- Description: HTTP Response Splitting attacks manipulate the content of HTTP responses, enabling attackers to inject malicious content into web pages. This can lead to arbitrary code execution, session hijacking, or other security breaches.

Task-6

UNDERSTANDING CIS POLICY VERSION 7

Introduction:

The Center for Internet Security (CIS) is a globally recognized authority in the field of cybersecurity. Its CIS Controls and CIS Benchmarks provide essential guidelines and best practices for organizations to secure their systems and data. In this assignment, we will delve into the key aspects of CIS Policy Version 7, highlighting its importance and relevance in today's rapidly evolving cybersecurity landscape.

"CIS Policy" and "Version 7" are different components of the same concept. The "CIS Policy" refers to the cybersecurity policy framework developed by the Center for Internet Security (CIS), while "Version 7" indicates the specific version or iteration of that policy framework. In this context, "Version 7" represents the seventh revision or update of the CIS Policy. Each version typically includes refinements, updates, and adjustments to adapt to the evolving cybersecurity landscape and emerging threats. So, they are related but represent different aspects of the same cybersecurity framework.

Overview of CIS Policy Version 7

CIS Policy Version 7 represents a comprehensive framework for enhancing an organization's cybersecurity posture. It builds upon the lessons learned from previous versions, taking into account emerging threats and technological advancements. The policy emphasizes the critical need for a proactive and adaptive approach to security.

Key Features and Components

- **CIS Controls:** The policy incorporates the 20 CIS Controls, which are a set of prioritized actions designed to mitigate the most prevalent cyber threats. These controls cover areas such as asset management, continuous monitoring, and incident response.
- **CIS Benchmarks:** CIS Benchmarks provide specific configuration guidelines for various software and hardware components. These benchmarks are continuously updated to address vulnerabilities and maintain alignment with industry standards.
- **Risk Management:** CIS Policy Version 7 places a strong emphasis on risk assessment and management. It encourages organizations to identify, assess, and prioritize risks to make informed decisions about security measures.

Importance of CIS Policy Version 7

- **Adaptability:** In the face of evolving cyber threats, CIS Policy Version 7 offers a flexible framework that can be tailored to suit the unique needs of different organizations and industries.

- **Proactive Security:** By implementing the CIS Controls and Benchmarks, organizations can proactively identify and address vulnerabilities before they can be exploited by malicious actors.
- **Compliance:** Many regulatory authorities and industry standards refer to CIS Controls and Benchmarks, making adherence to CIS Policy Version 7 instrumental in achieving compliance.

Benefits of CIS Policy Version 7

- **Enhanced Security:** Implementing CIS Policy Version 7 enhances an organization's overall security posture, reducing the risk of data breaches and cyberattacks.
- **Cost-Efficiency:** By following established best practices and configurations, organizations can optimize their security investments and reduce the cost of incident response.
- **Reputation Protection:** Strong cybersecurity measures as per CIS Policy Version 7 can safeguard an organization's reputation and build trust with customers, partners, and stakeholders.

CIS Policy Version 7 serves as a vital resource for organizations seeking to strengthen their cybersecurity defenses. Its comprehensive approach, adaptability, and focus on risk management make it a valuable tool in the ongoing battle against cyber threats. By embracing the principles and guidelines outlined in this policy, organizations can better protect their assets and maintain a secure digital environment.

Task-7

SELECT A WEBSITE DO FOOTPRINTING AND RECONNAISSANCE LIKE COLLECT INFORMATION ABOUT WEBSITE USING NSLOOKUP OSINT FRAMEWORK

The website choosen is <http://testfire.net/> . Wensite and tools used to collect information.

1.<https://www.nslookup.io/domains/testfire.net/dns-records/> (ns lookup)

2.<https://osintframework.com/> (osint framework)

3.nmap

Footprinting using nmap:-

```
niraianbu@localhost: ~  
(niraianbu@localhost)-[~]  
$ nmap testfire.net  
Starting Nmap 7.94 ( https://nmap.org ) at 2023-10-07 12:55 IST  
Nmap scan report for testfire.net (65.61.137.117)  
Host is up (0.27s latency).  
Not shown: 996 filtered tcp ports (no-response)  
PORT      STATE SERVICE  
80/tcp    open  http  
443/tcp    open  https  
8080/tcp   open  http-proxy  
8443/tcp   closed https-alt  
  
Nmap done: 1 IP address (1 host up) scanned in 20.81 seconds  
  
(niraianbu@localhost)-[~]  
$
```

1. Port 80/tcp: This is the HTTP port, commonly used for serving web pages over the standard HTTP protocol. It suggests that a web server is running on this port, and you can access web content hosted on this server by opening a web browser and navigating to the IP address or domain name.

2. Port 443/tcp: This is the HTTPS port, used for secure web communication over the HTTPS protocol. Similar to port 80, it indicates the presence of a web server that offers secure browsing capabilities.
3. Port 8080/tcp: This is typically associated with an HTTP proxy server. An HTTP proxy server can be used to forward web requests from clients to other servers while providing various functions like caching, filtering, or anonymizing. It might be used for load balancing or filtering web traffic.
4. Port 8443/tcp: This port is closed, meaning there is no active service listening on it. Port 8443 is often associated with HTTPS services running on an alternative port. The fact that it's closed suggests that there might not be a service configured to listen on this port.

DNS records for **testfire.net**

Cloudflare Google DNS OpenDNS Authoritative Local DNS ▾



The Cloudflare DNS server responded with these DNS records. Cloudflare will serve these records for as long as the time to live (TTL) has not expired. After this period, Cloudflare will update its cache by querying one of the authoritative name servers.

A records

IPv4 address	Revalidate in
>  65.61.137.117	24h

AAAA records

No AAAA records found.

CNAME record

No CNAME record found.

TXT records

SPF record

This record is valid for 30m.

Pass if the email sender's IP is in the MX records (with CIDR /24 for IPv4) of testfire.net.

mx/24

Or else, mark the email as fail.

-all

By Nslookup.io

DNS for Developers

Never be confused
about DNS again.

NS records

Name server	Revalidate in
usc3.akam.net.	24h
ns1-206.akam.net.	24h
ns1-99.akam.net.	24h
eur5.akam.net.	24h
usw2.akam.net.	24h
usc2.akam.net.	24h
eur2.akam.net.	24h
asia3.akam.net.	24h

MX records

No mail servers found.

Other records

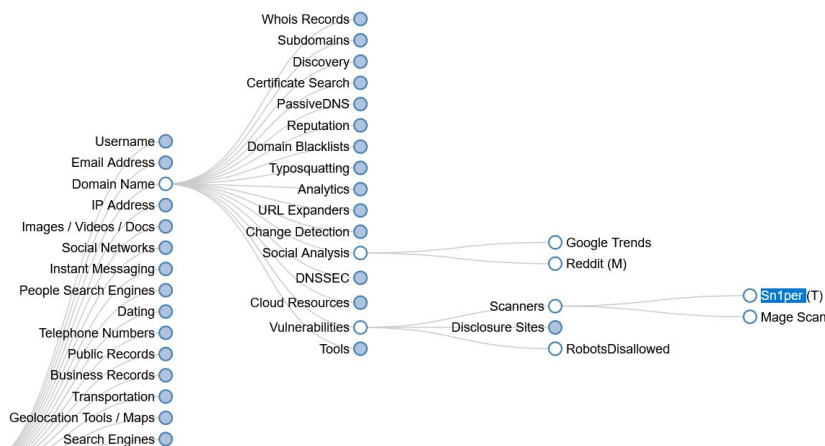
SOA


SOA data	Revalidate in
Start of authority	asia3.akam.net. 24h
Email	hostmaster@akamai.com
Serial	1366025607
Refresh	12h
Retry	2h
Expire	168h
Negative cache TTL	24h

Onsit frame work:

OSINT Framework

(T) - Indicates a link to a tool that must be installed and run l
(D) - Google Dork, for more information: [Google Hacking](#)
(R) - Requires registration
(M) - Indicates a URL that contains the search term and the
itself must be edited manually




 **Sn1per** Public

Watch 328

master 1 branch 53 tags

Go to file Add file Code

 **1N3 Delete bin/inurlbr.php**

✓ d51dece last week 🕒 593 commits

bin	Delete bin/inurlbr.php	last week
conf	Merge pull request #422 from benemohamed/master	3 months ago
loot	Delete nmap-10.0.0.1.xml	7 years ago
modes	Merge pull request #422 from benemohamed/master	3 months ago
pro	Sn1per by @sn1persecurity - https://sn1persecurity.com	2 years ago
templates	Sn1per by @sn1persecurity - https://sn1persecurity.com	10 months ago
wordlists	Sn1per by @sn1persecurity - https://sn1persecurity.com	2 years ago