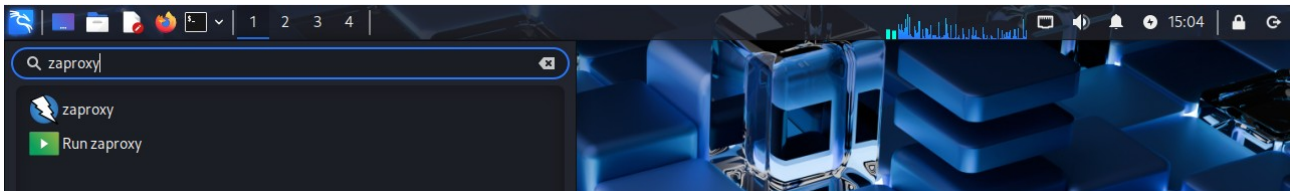


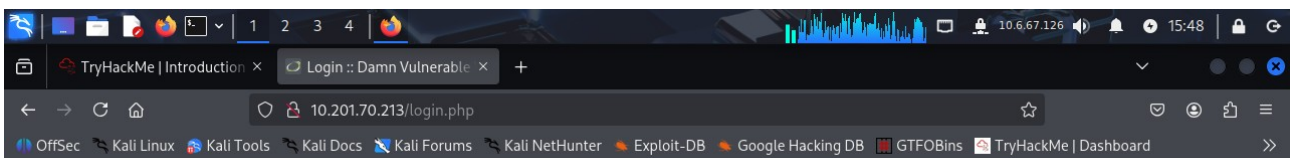
Introdução ao ZAP Zed Proxy Attack

1º Instalação no Kali Linux

```
$ sudo apt install zaproxy
$ owasp-zap
$ zaproxy
```



Zaproxy pode ser inicializado via shell ou menu iniciar do Kali Linux



Username

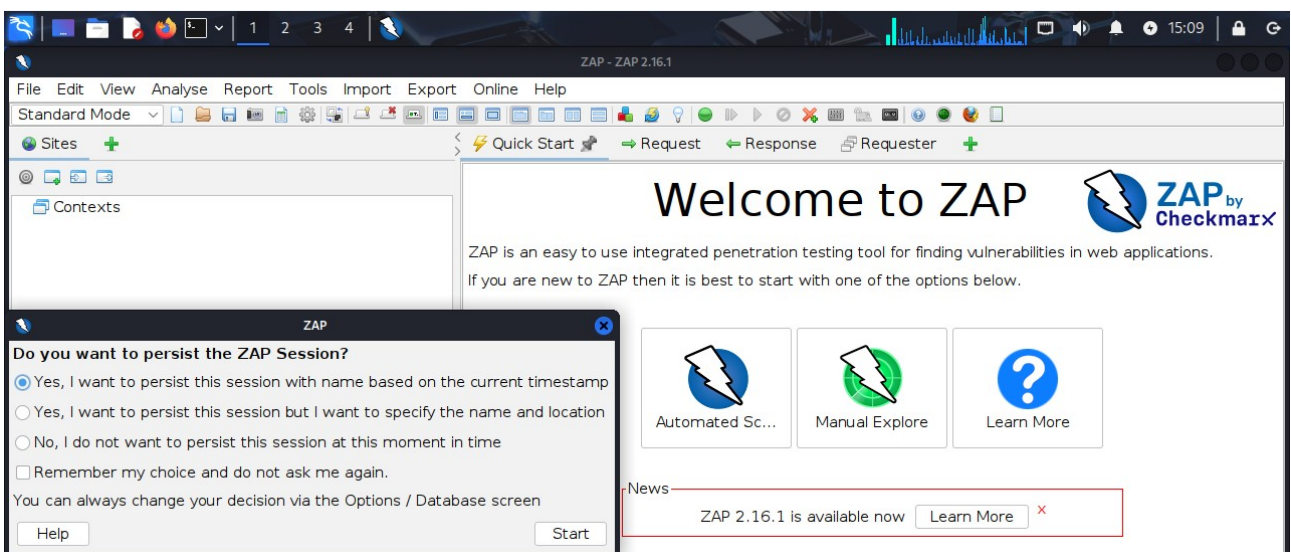
Password

Login

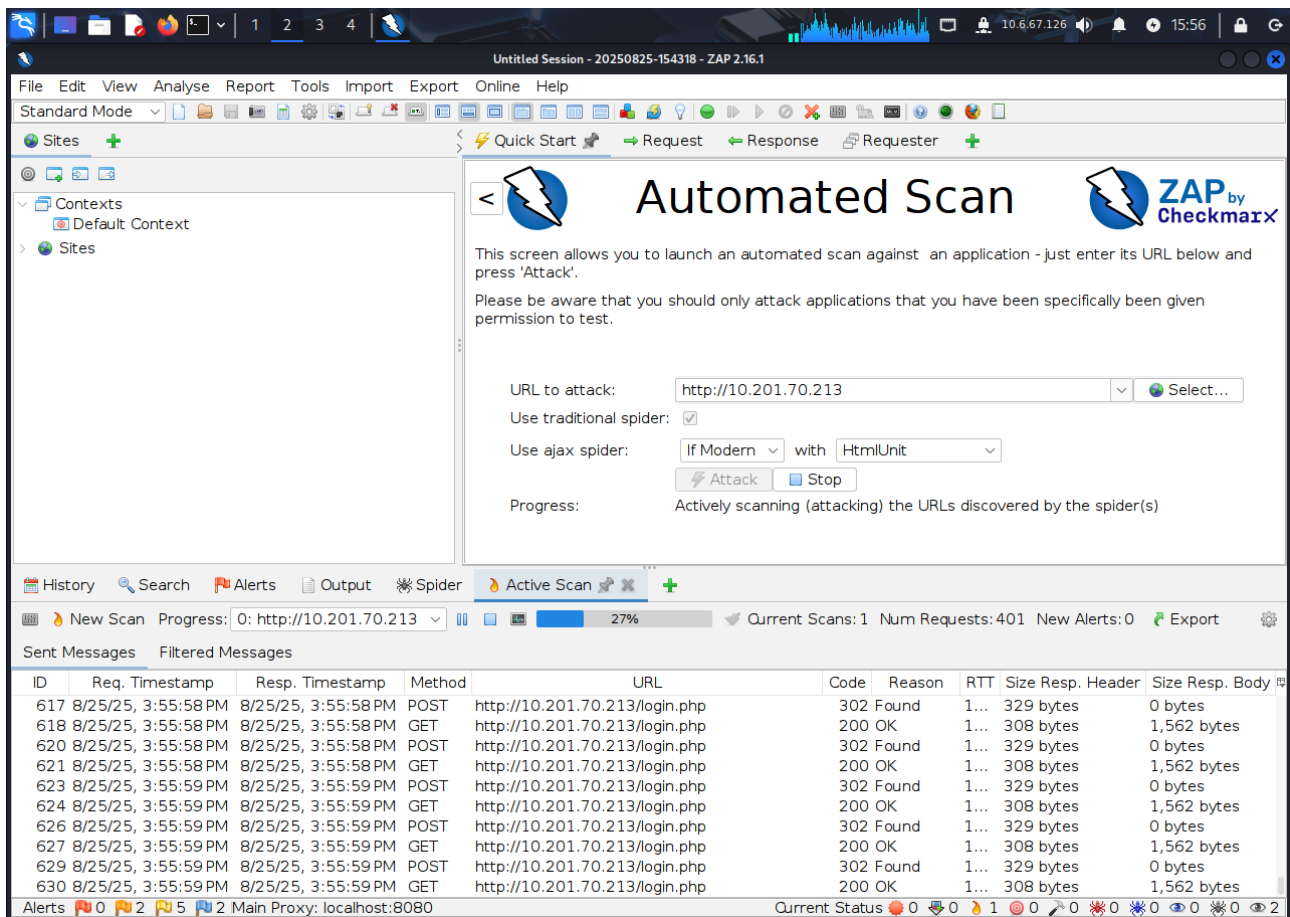
Inicialize uma máquina virtual ou contêiner “docker” com a aplicação DVWA

```
$sudo docker pull vulnerables/web-dvwa,  
$sudo docker run --rm -it -p 80:80 vulnerables/web-dvwa
```

2º Zaproxy modos de Scan Automated/Manual



Clique “Start” no botão grande “Automated Scan” e insira seu alvo.



Quick Start > Preencher os campos

➔ URL to attack: http://10.201.70.213

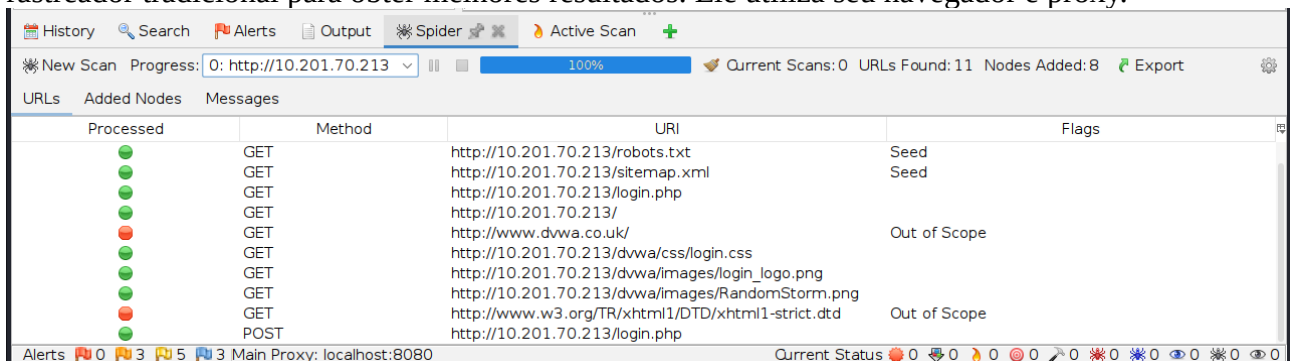
Opções para selecionar entre "spider tradicional" ou "spider Ajax".

➔ Usar tradicional spider

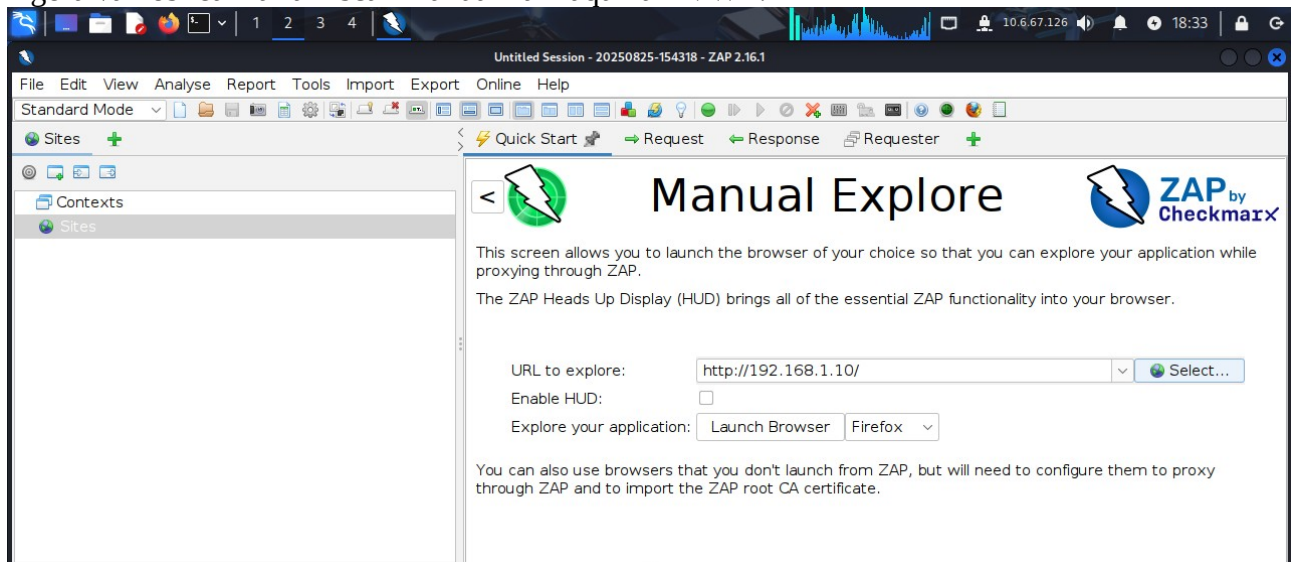
➔ Usar Ajax spider: Se moderno com: HtmlUnit

Clique no botão **“Attack”** para iniciar o Scan

Uma varredura de spider tradicional é uma varredura passiva que enumera links e diretórios do site. Ela cria um índice do site sem força bruta. Isso é muito mais silencioso do que um ataque de força bruta e ainda pode revelar uma página de login ou outros detalhes interessantes, mas não é tão abrangente quanto um ataque de força bruta. O Ajax Spider é um complemento que integra ao ZAP um rastreador de sites ricos em AJAX chamado “Crawljax”. Você pode usá-lo em conjunto com o rastreador tradicional para obter melhores resultados. Ele utiliza seu navegador e proxy.



Agora vamos realizar um scan manual na máquina DVWA.

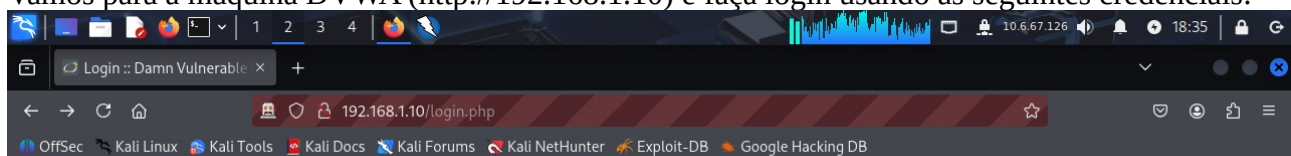


Quick Start > Manual Explore

- ➔ URL para ataque: `http://192.168.1.10`
- ➔ Clique no botão **“Launch Browser”** para iniciar o navegador Firefox com o proxy localhost pré-configurado.

* Varredura Autenticada

Vamos para a máquina DVWA (`http://192.168.1.10`) e faça login usando as seguintes credenciais:



Username
admin

Password
.....

Login

Nome de usuário: admin, senha: password

Para este exercício, após fazer login, clique em DVWA security, defina o nível de segurança como **“Low”** e clique em **“Submit”**.

Vamos passar nosso token de autenticação para o ZAP para que possamos usar a ferramenta para escanear páginas da web autenticadas.

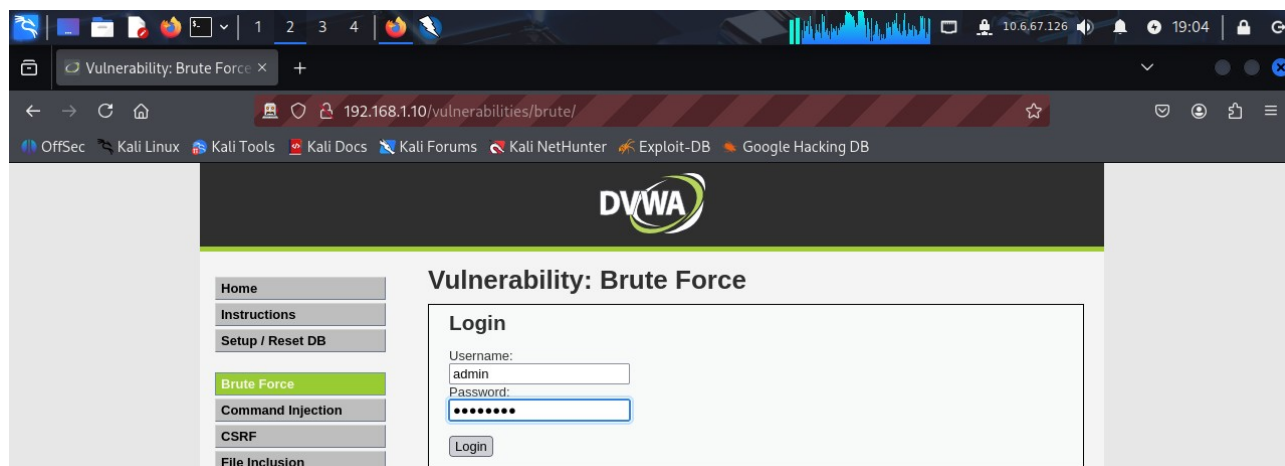
Primeiro precisamos verificar e anotar o cookie PHPSESSION.

No browser clique com o botão direito em Inspect > Storage > Cookies

3º Login aplicando Força Bruta

Vamos usar a força bruta em um formulário para obter credenciais. Embora já saibamos as credenciais, vamos ver se podemos usar o Zap para obtê-las por meio de um ataque de Força Bruta. Se você quisesse fazer isso com o BurpSuite, precisaria interceptar a solicitação e passá-la para o Hydra. No entanto, esse processo é muito mais fácil com o ZAP!

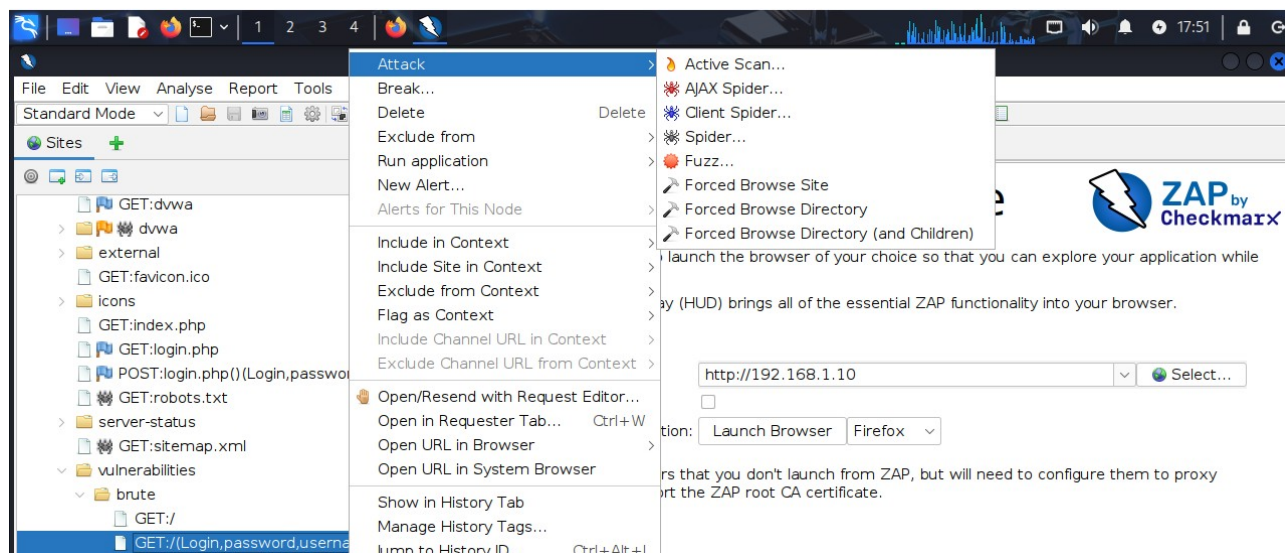
No ZAP em **“Explore your application:”** clique em **“Launch Browser”** e clique no botão de Brute Force no DVWA e tente fazer login com "admin" e senha "passwd123".



<https://192.168.1.10/vulnerabilities/brute/>

Em seguida, localize em Sites a solicitação GET e abra o menu Fuzz.

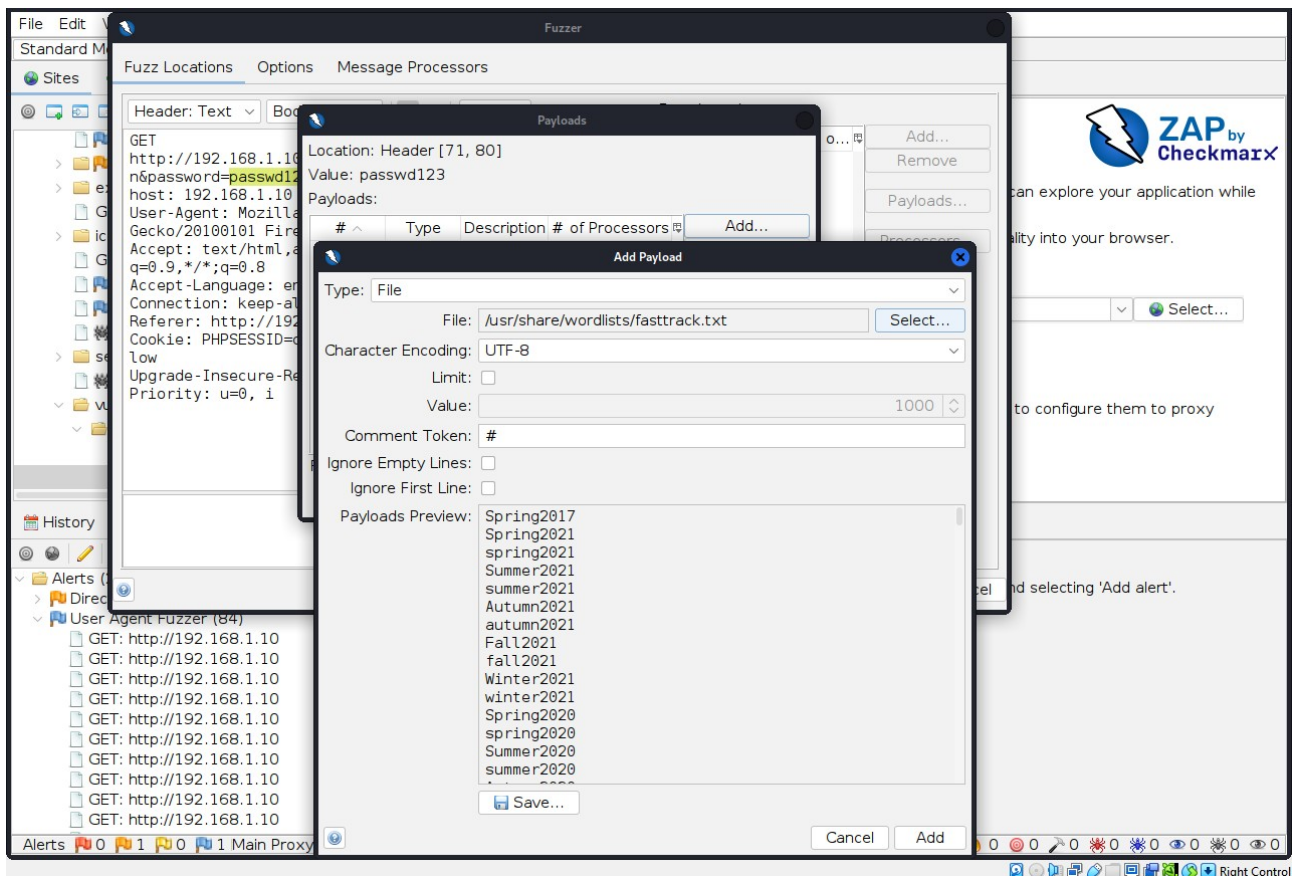
Sites > vulnerabilities > GET:/login,password,username > Attack > Fuzz



Em seguida, destaque a senha que você tentou e adicione uma lista de palavras. Isso selecionando a área da solicitação que você deseja substituir por outros dados.

Add > File: Selecione o caminho do arquivo > fasttrack.txt > OK > Start Fuzzer

Para maior velocidade, podemos usar fasttrack.txt, que está localizado em /usr/share/wordlists se você estiver usando o Kali Linux.



Após executar o fuzzer, classifique a aba de estado para mostrar os resultados refletidos primeiro. Às vezes, você receberá falsos positivos, mas pode ignorar senhas com menos de 8 caracteres.

History

Search

Alerts

Output

Spider

HTTP Sessions

Fuzzer

New Fuzzer

Progress: 3: HTTP - http://192.168.1.123&Login=Login

100%

Current fuzzers: 0

Messages Sent: 262

Errors: 0

Show Errors

Export

Task ID	Message Type	Code	Reason	RTT	Size Resp. Header	Size Resp. Body	Highest Alert	State	Payloads
262 Fuzzed		200 OK		4 ms	327 bytes	4,375 bytes			starwars
169 Fuzzed		200 OK		8 ms	327 bytes	4,375 bytes	Reflected		admin
230 Fuzzed		200 OK		6 ms	327 bytes	4,375 bytes	Reflected		nt
147 Fuzzed		200 OK		8 ms	327 bytes	4,375 bytes	Reflected		pass
106 Fuzzed		200 OK		23 ms	327 bytes	4,413 bytes	Reflected		password
145 Fuzzed		200 OK		12 ms	327 bytes	4,375 bytes	Reflected		sa
151 Fuzzed		200 OK		10 ms	327 bytes	4,375 bytes	Reflected		sa
112 Fuzzed		200 OK		6 ms	327 bytes	4,375 bytes	Reflected		security
120 Fuzzed		200 OK		57 ms	327 bytes	4,375 bytes	Reflected		sql
148 Fuzzed		200 OK		17 ms	327 bytes	4,375 bytes	Reflected		sql

Alerts

0

8

10

6 Main Proxy: localhost:8080

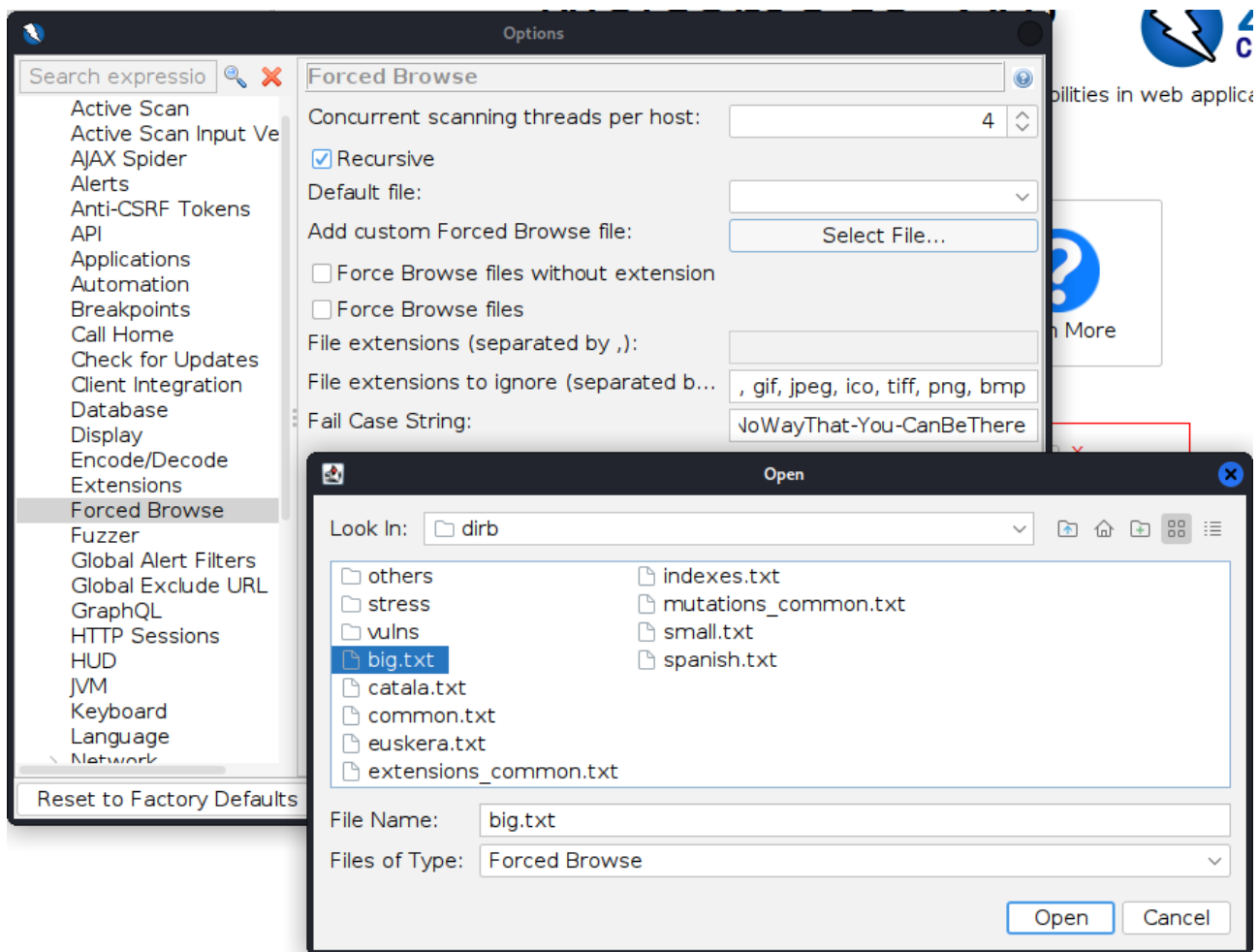
Current Status

4º Varredura de Diretórios

Se as varreduras passivas não forem suficientes, você pode usar um ataque de lista de palavras e força bruta de diretório através do ZAP, assim como faria com o Dirb ou Gobuster. Isso detectará páginas que não estão indexadas.

Options > Forced Browse > Add custom forced browse file > Select File > Open

Primeiro. Acesse as Opções do ZAP (no painel de navegação inferior, com o botão de adição na tela), navegue até Navegação Forçada e adicione a Lista de Palavras Personalizada. Você também pode adicionar mais threads e desativar a força bruta recursiva.

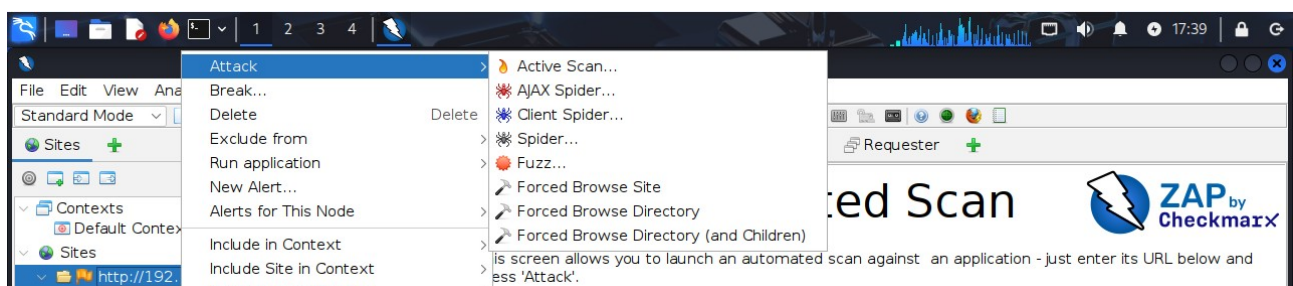


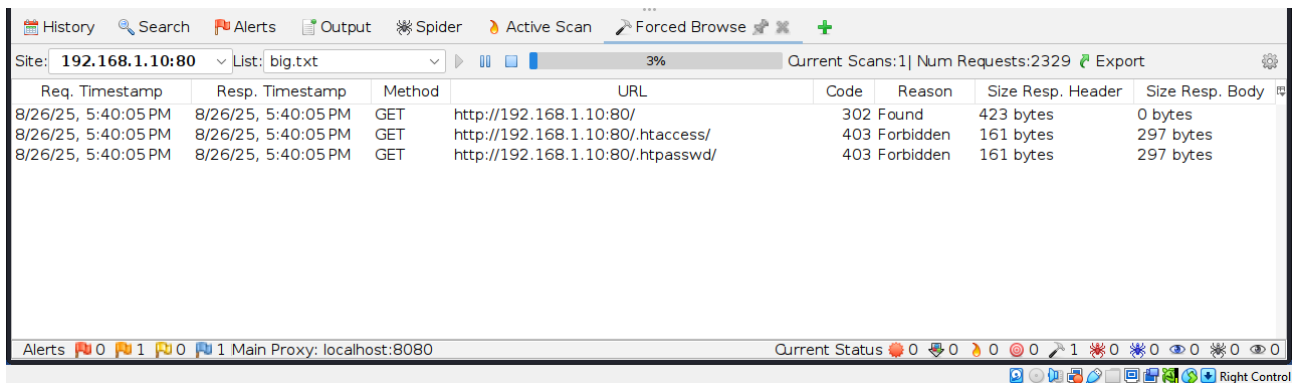
Agora é clicar com o botão direito no alvo e selecione a opção:

Sites > <http://192.168.1.10> > Attack > Forced Browse Site > List: custom-list.txt

Sites > <http://192.168.1.10> > Attack > Forced Browse Directory > List: custom-list.txt

Selecione a lista de palavras importada no menu de listas e clique no botão de reprodução!
Recomendamos usar esta lista de palavras para este exercício.





O ZAP agora aplicará força bruta em todo o site com a sua lista de palavras.

5º Instalar Extensões do ZAP

Quer aprimorar ainda mais os recursos do ZAP? Confira algumas de suas extensões para download!

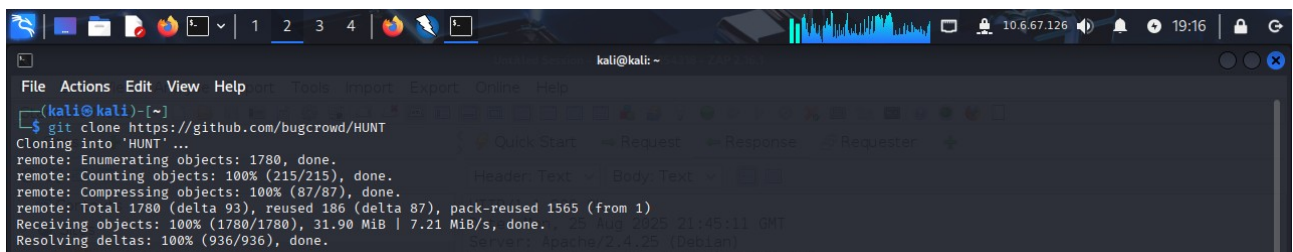
<https://github.com/zaproxy/zap-extensions>

<https://github.com/bugcrowd/HUNT>

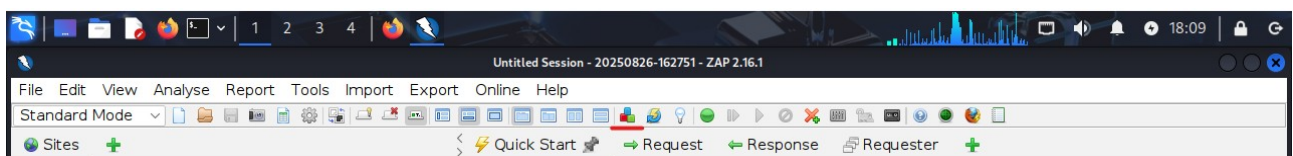
Vamos instalar as extensões HUNT do bugcrowd para o OWASP ZAP. Isso fará uma varredura passiva em busca de vulnerabilidades conhecidas em aplicações web.

Primeiro, navegue em seu terminal até o local onde você deseja armazenar os scripts.

\$ git clone https://github.com/bugcrowd/HUNT



Em seguida, no ZAP, clique no ícone “**Manage Add-Ons**”.

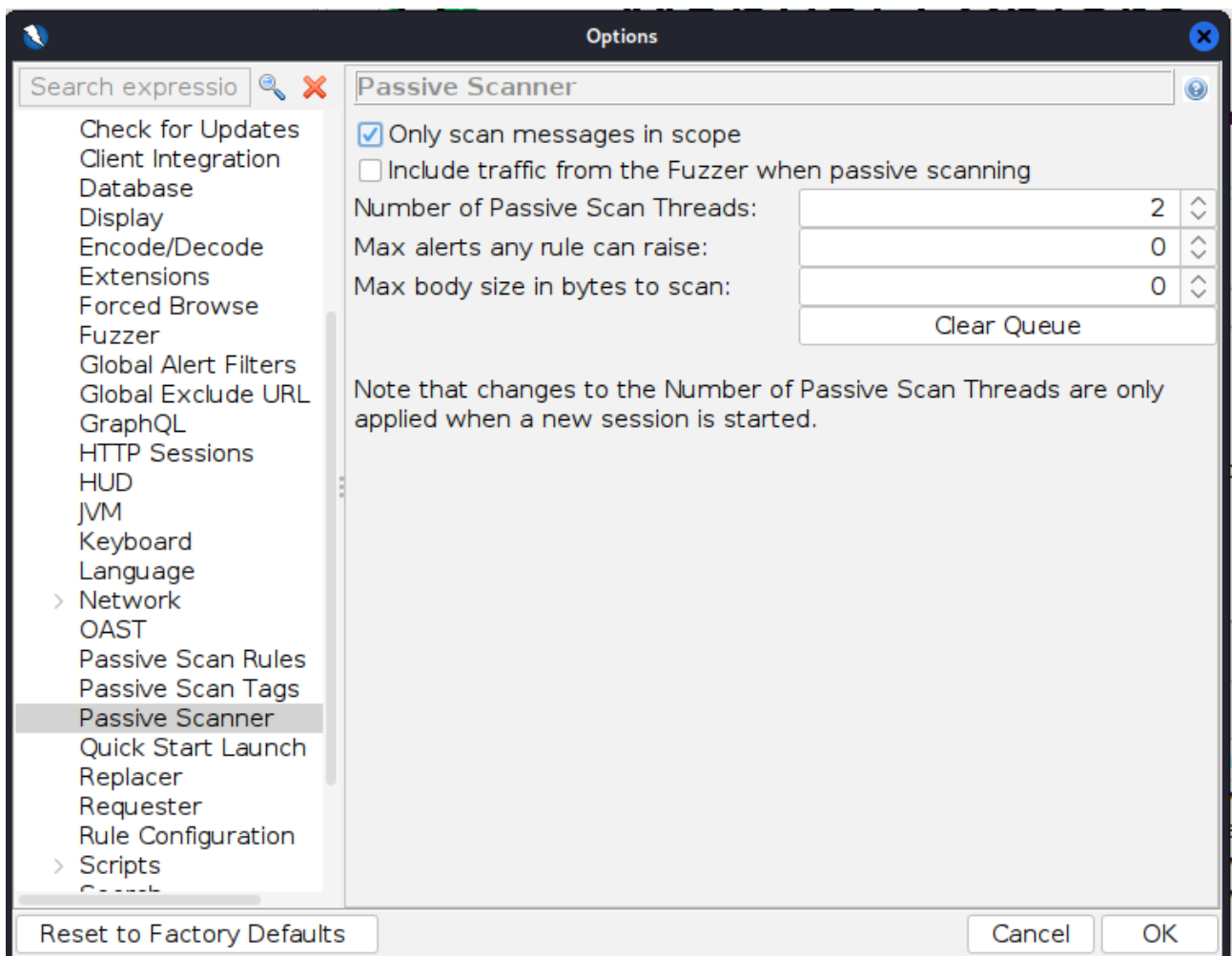
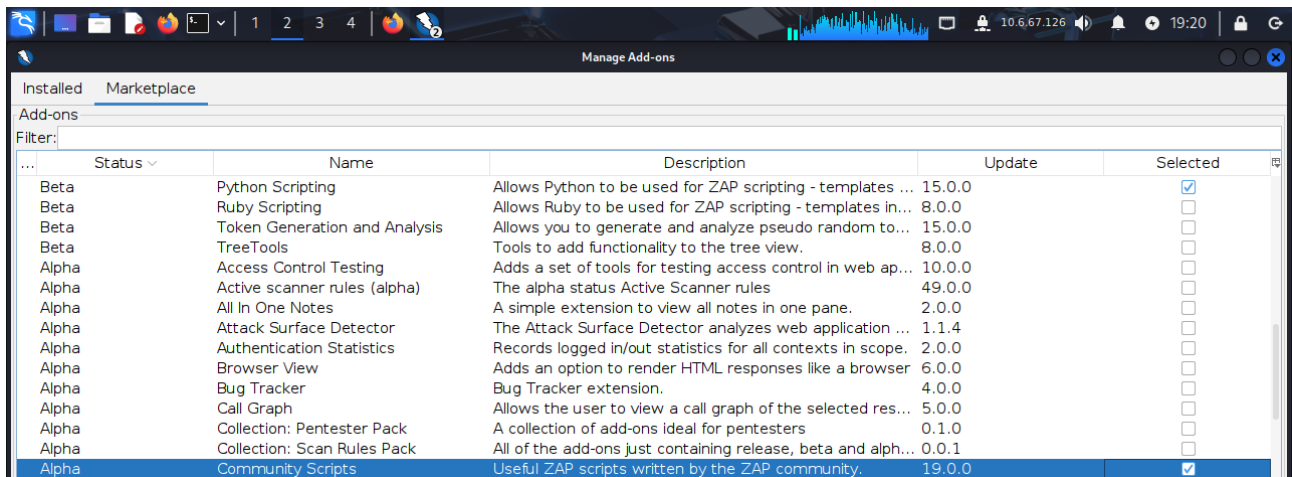


No Marketplace, procure por:

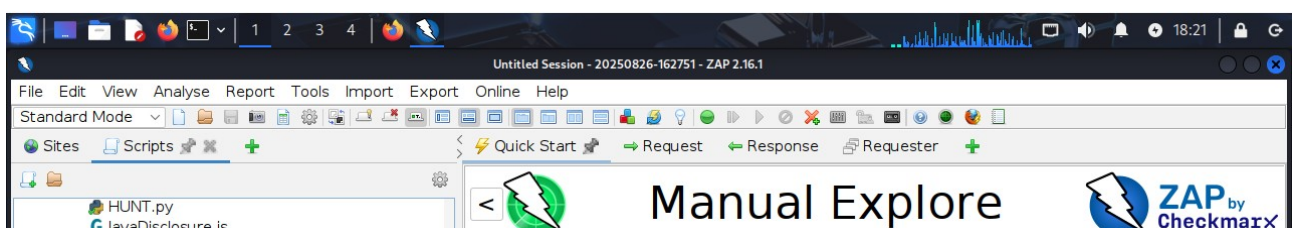
“Python Scripting” and **“Community Scripts”**

Selecione e instale os Add-ons.

Em Options do ZAP, em Passive Scanner, certifique-se de que **“Only scan message in scope”** esteja habilitado. Em seguida, clique em OK.



No ZAP, em Sites abra uma aba + Scripts.



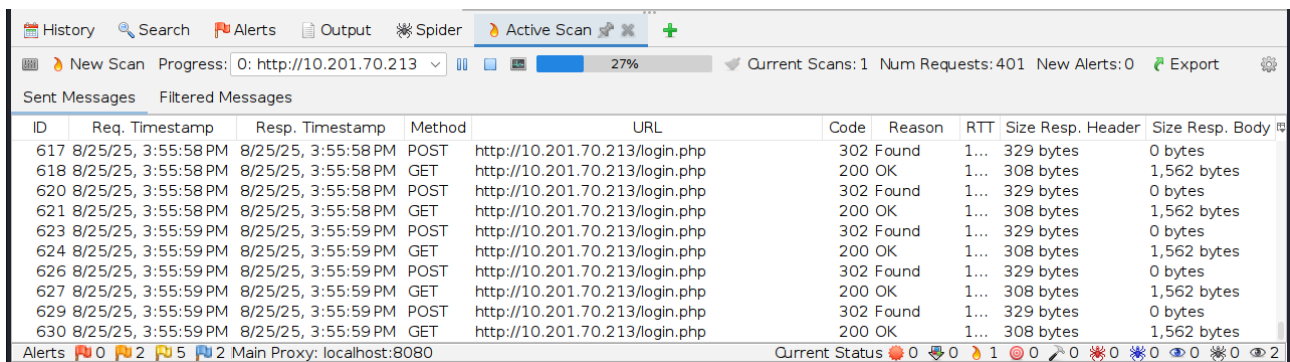
Em Passive Rules, localize e ative o script HUNT.py.

Agora, ao navegar em sites, o HUNT fará a varredura passiva de SQLi, LFI, RFI, SSRF e outros.

Configure o HUNT no seu aplicativo Zap para executar varreduras passivas automaticamente nos sites que você visita!

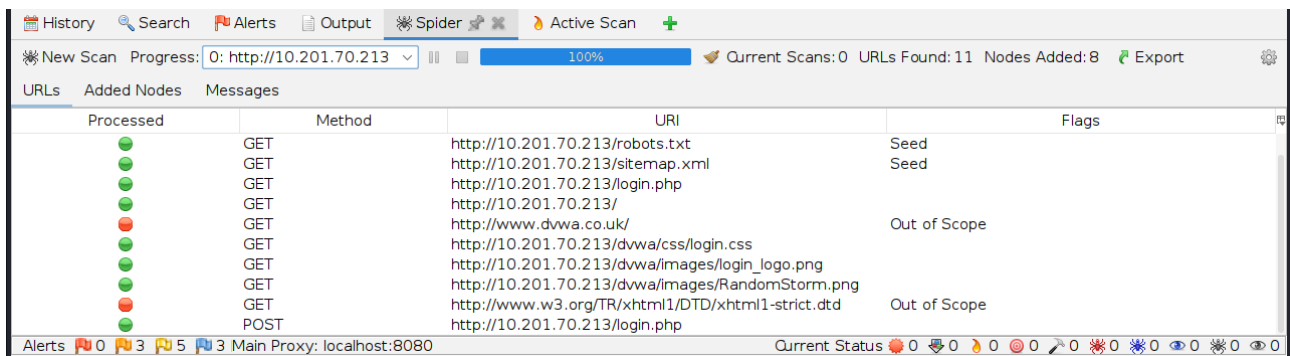
6º Painel de Progresso e Resultados dos Scans

➔ Active Scan



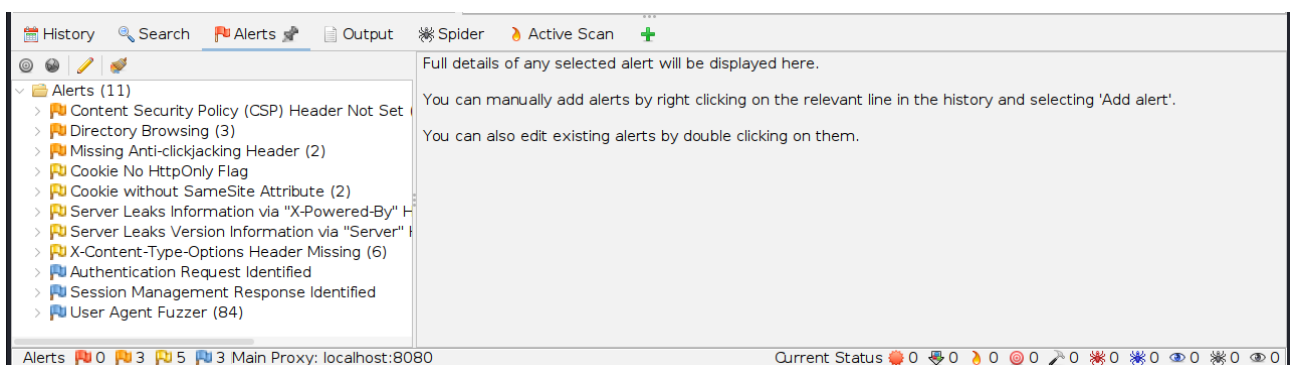
ID	Req. Timestamp	Resp. Timestamp	Method	URL	Code	Reason	RTT	Size Resp. Header	Size Resp. Body
617	8/25/25, 3:55:58 PM	8/25/25, 3:55:58 PM	POST	http://10.201.70.213/login.php	302	Found	1...	329 bytes	0 bytes
618	8/25/25, 3:55:58 PM	8/25/25, 3:55:58 PM	GET	http://10.201.70.213/login.php	200	OK	1...	308 bytes	1,562 bytes
620	8/25/25, 3:55:58 PM	8/25/25, 3:55:58 PM	POST	http://10.201.70.213/login.php	302	Found	1...	329 bytes	0 bytes
621	8/25/25, 3:55:58 PM	8/25/25, 3:55:58 PM	GET	http://10.201.70.213/login.php	200	OK	1...	308 bytes	1,562 bytes
623	8/25/25, 3:55:59 PM	8/25/25, 3:55:59 PM	POST	http://10.201.70.213/login.php	302	Found	1...	329 bytes	0 bytes
624	8/25/25, 3:55:59 PM	8/25/25, 3:55:59 PM	GET	http://10.201.70.213/login.php	200	OK	1...	308 bytes	1,562 bytes
626	8/25/25, 3:55:59 PM	8/25/25, 3:55:59 PM	POST	http://10.201.70.213/login.php	302	Found	1...	329 bytes	0 bytes
627	8/25/25, 3:55:59 PM	8/25/25, 3:55:59 PM	GET	http://10.201.70.213/login.php	200	OK	1...	308 bytes	1,562 bytes
629	8/25/25, 3:55:59 PM	8/25/25, 3:55:59 PM	POST	http://10.201.70.213/login.php	302	Found	1...	329 bytes	0 bytes
630	8/25/25, 3:55:59 PM	8/25/25, 3:55:59 PM	GET	http://10.201.70.213/login.php	200	OK	1...	308 bytes	1,562 bytes

➔ Spider



Processed	Method	URI	Flags
●	GET	http://10.201.70.213/robots.txt	Seed
●	GET	http://10.201.70.213/sitemap.xml	Seed
●	GET	http://10.201.70.213/login.php	
●	GET	http://10.201.70.213/	
●	GET	http://www.dvwa.co.uk/	Out of Scope
●	GET	http://10.201.70.213/dvwa/css/login.css	
●	GET	http://10.201.70.213/dvwa/images/login_logo.png	
●	GET	http://10.201.70.213/dvwa/images/RandomStorm.png	
●	GET	http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd	Out of Scope
●	POST	http://10.201.70.213/login.php	

➔ Alerts



Alerts (11)	Full details of any selected alert will be displayed here.
> Content Security Policy (CSP) Header Not Set	You can manually add alerts by right clicking on the relevant line in the history and selecting 'Add alert'. You can also edit existing alerts by double clicking on them.
> Directory Browsing (3)	
> Missing Anti-clickjacking Header (2)	
> Cookie No HttpOnly Flag	
> Cookie without SameSite Attribute (2)	
> Server Leaks Information via "X-Powered-By" H	
> Server Leaks Version Information via "Server" I	
> X-Content-Type-Options Header Missing (6)	
> Authentication Request Identified	
> Session Management Response Identified	
> User Agent Fuzzer (84)	