

Лабораторная работа №6.  
Утилита nmap

Евсеев Дмитрий

28 июня 2016 г.

# Оглавление

|      |  |    |
|------|--|----|
| 1    | Цель работы . . . . .  | 2  |
| 2    | Описание . . . . .   | 2  |
| 3    | Ход работы . . . . .   | 2  |
| 3.1  | Поиск активных хостов . . . . .  | 2  |
| 3.2  | Определение открытых портов . . . . .  | 3  |
| 3.3  | Определение версий сервисов . . . . .  | 4  |
| 3.4  | Исследование служебных файлов nmap-services, nmap-os-db, nmap-service-probes . . . . . | 4  |
| 3.5  | Создание собственной сигнатуры . . . . .   | 6  |
| 3.6  | Сохранение вывода в xml . . . . .  | 7  |
| 3.7  | Исследование работы Nmap с помощью Wireshark . . . . .                                 | 7  |
| 3.8  | Metasploit Framework . . . . .   | 8  |
| 3.9  | Примеры записей из nmap-service-probes . . . . .                                       | 10 |
| 3.10 | Описание скрипта finger . . . . .  | 11 |
| 4    | Вывод . . . . .  | 13 |

# 1 Цель работы

Научиться работать с Nmap

## 2 Описание

**nmap** — свободная утилита, предназначенная для разнообразного настраиваемого сканирования IP-сетей с любым количеством объектов, определения состояния объектов сканируемой сети (портов и соответствующих им служб). Изначально программа была реализована для систем UNIX, но сейчас доступны версии для множества операционных систем.

Nmap использует множество различных методов сканирования, таких как UDP, TCP (connect), TCP SYN (полуоткрытое), FTP-прокси (прорыв через ftp), Reverse-ident, ICMP (ping), FIN, ACK, Xmas tree, SYN- и NULL-сканирование. Nmap также поддерживает большой набор дополнительных возможностей, а именно: определение операционной системы удалённого хоста с использованием отпечатков стека TCP/IP, «невидимое» сканирование, динамическое вычисление времени задержки и повтор передачи пакетов, параллельное сканирование, определение неактивных хостов методом параллельного ping-опроса, сканирование с использованием ложных хостов, определение наличия пакетных фильтров, прямое (без использования portmapper) RPC-сканирование, сканирование с использованием IP-фрагментации, а также произвольное указание IP-адресов и номеров портов сканируемых сетей.

Для проведения работы будем использовать две виртуальные машины:

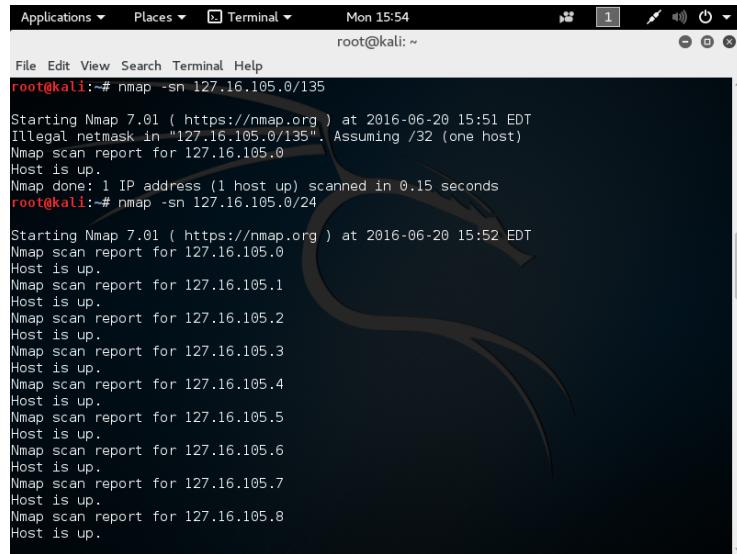
- Kali linux для поиска и эксплуатации уязвимостей. IP: 172.16.105.134
- Metasploitable2 для тестирования. IP 172.16.105.135

## 3 Ход работы

### 3.1 Поиск активных хостов

Сканируем локальную сеть и ищем активные хосты с помощью команды nmap с ключом -sn и диапазоном ip адресов (рисунок 1).

```
$ nmap -sn 127.16.105.0/24
```



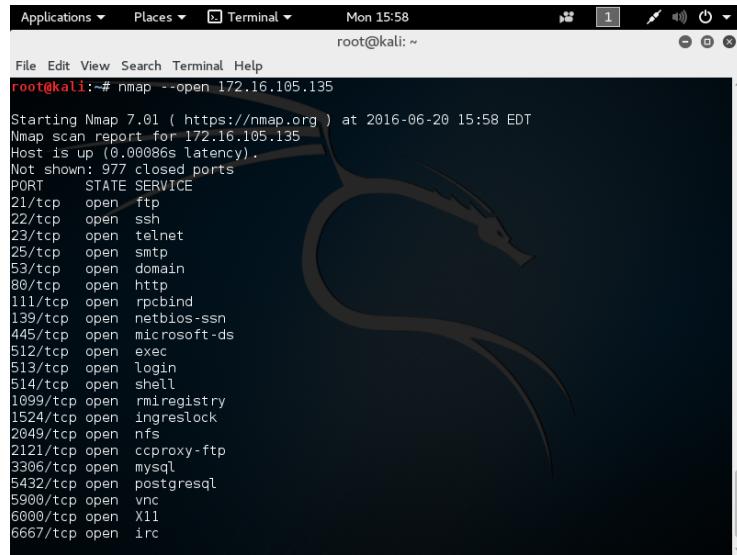
```
root@kali:~# nmap -sn 127.16.105.0/135
Starting Nmap 7.01 ( https://nmap.org ) at 2016-06-20 15:51 EDT
Illegal netmask in '127.16.105.0/135'. Assuming /32 (one host)
Nmap scan report for 127.16.105.0
Host is up.
Nmap done: 1 IP address (1 host up) scanned in 0.15 seconds
root@kali:~# nmap -sn 127.16.105.0/24
Starting Nmap 7.01 ( https://nmap.org ) at 2016-06-20 15:52 EDT
Nmap scan report for 127.16.105.0
Host is up.
Nmap scan report for 127.16.105.1
Host is up.
Nmap scan report for 127.16.105.2
Host is up.
Nmap scan report for 127.16.105.3
Host is up.
Nmap scan report for 127.16.105.4
Host is up.
Nmap scan report for 127.16.105.5
Host is up.
Nmap scan report for 127.16.105.6
Host is up.
Nmap scan report for 127.16.105.7
Host is up.
Nmap scan report for 127.16.105.8
Host is up.
```

Рис. 1: Поиск активных хостов.

### 3.2 Определение открытых портов

Для определения открытых портов необходимо воспользоваться следующей командой (рисунок 2)

```
$ nmap --open [hostIP]
```



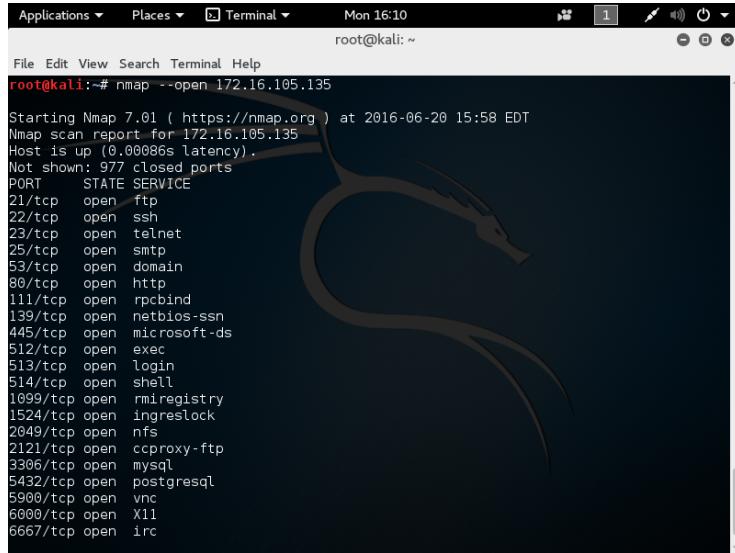
```
root@kali:~# nmap --open 172.16.105.135
Starting Nmap 7.01 ( https://nmap.org ) at 2016-06-20 15:58 EDT
Nmap scan report for 172.16.105.135
Host is up (0.00066s latency).
Not shown: 977 closed ports
PORT      STATE SERVICE
21/tcp    open  ftp
22/tcp    open  ssh
23/tcp    open  telnet
25/tcp    open  smtp
53/tcp    open  domain
80/tcp    open  http
111/tcp   open  rpcbind
139/tcp   open  netbios-ssn
445/tcp   open  microsoft-ds
512/tcp   open  exec
513/tcp   open  login
514/tcp   open  shell
1099/tcp  open  rmiregistry
1524/tcp  open  ingreslock
2049/tcp  open  nfs
2121/tcp  open  cccproxy-ftp
3306/tcp  open  mysql
5432/tcp  open  postgresql
5900/tcp  open  vnc
6000/tcp  open  X11
6667/tcp  open  irc
```

Рис. 2: Определение открытых портов.

### 3.3 Определение версий сервисов

Для определения открытых портов необходимо воспользоваться следующей командой (рисунок 2)

```
$ nmap --open [hostIP]
```



The screenshot shows a terminal window titled 'Terminal' with the command 'root@kali: ~'. The output of the 'nmap --open 172.16.105.135' command is displayed, listing various open ports and their corresponding services. The output includes:

```
Starting Nmap 7.01 ( https://nmap.org ) at 2016-06-20 15:58 EDT
Nmap scan report for 172.16.105.135
Host is up (0.00086s latency).
Not shown: 977 closed ports
PORT      STATE SERVICE
21/tcp    open  ftp
22/tcp    open  ssh
23/tcp    open  telnet
25/tcp    open  smtp
53/tcp    open  domain
80/tcp    open  http
111/tcp   open  rpcbind
139/tcp   open  netbios-ssn
445/tcp   open  microsoft-ds
512/tcp   open  exec
513/tcp   open  login
514/tcp   open  shell
1099/tcp  open  rmiregistry
1524/tcp  open  ingreslock
2049/tcp  open  nfs
2121/tcp  open  cccproxy-ftp
3306/tcp  open  mysql
5432/tcp  open  postgresql
5900/tcp  open  vnc
6000/tcp  open  X11
6667/tcp  open  irc
```

Рис. 3: Определение версий сервисов.

### 3.4 Исследование служебных файлов nmap-services, nmap-os-db, nmap-service-probes

Служебные файлы для утилиты nmap по умолчанию располагаются в директории "/usr/share/nmap".

#### nmap-services

Служебный файл nmap-services содержит в себе описание назначения стандартных портов. Сам файл имеет структуру таблицы со следующими столбцами: имя сервиса, номер порта, название протокола, вероятность того, что порт открыт, комментарии.

Для известных зарезервированных номеров портов, файл содержит подробное описание. Пример на рисунке 4.

```

Applications ▾ Places ▾ gedit ▾ Mon 16:18
nmap-services
/usr/share/nmap
Open Save
Plain Text Tab Width: 8 Ln 1, Col 1 INS
compressnet 3/udp 0.001532 # Compression Process
unknown 4/tcp 0.000477
rje 5/udp 0.000593 # Remote Job Entry
unknown 6/tcp 0.000500
echo 7/sctp 0.000000
echo 7/tcp 0.004855
echo 7/udp 0.024679
unknown 8/tcp 0.000013
discard 9/sctp 0.000000 # sink null
discard 9/tcp 0.003764 # sink null
discard 9/udp 0.015733 # sink null
unknown 10/tcp 0.000063
sysstat 11/tcp 0.000075 # Active Users
sysstat 11/udp 0.000577 # Active Users
unknown 12/tcp 0.000063
unknown 13/tcp 0.003927
daytime 13/udp 0.004827
unknown 14/tcp 0.000038
netstat 15/tcp 0.000038
unknown 16/tcp 0.000050
qotd 17/tcp 0.002346 # Quote of the Day
qotd 17/udp 0.009209 # Quote of the Day
msp 18/udp 0.000610 # Message Send Protocol
chargen 19/tcp 0.002559 # ttyst source Character Generator
chargen 19/udp 0.015865 # ttyst source Character Generator
ftp-data 20/sctp 0.000000 # File Transfer [Default Data]
ftp-data 20/tcp 0.001079 # File Transfer [Default Data]
ftp-data 20/udp 0.001878 # File Transfer [Default Data]

```

Рис. 4: Содержимое файла nmap-services.

### nmap-os-db

Данный файл необходим для определения ОС хоста. В ней содержится примеры ответов различных ОС на специальные запросы Nmap. Он разделен на блоки, так называемые отпечатки, содержащие название ОС, классификацию и данные ответа. Пример для ОС Linux на рисунке 5.

```

Applications ▾ Places ▾ gedit ▾ Mon 16:23
nmap-os-db
/usr/share/nmap
Open Save
Plain Text Tab Width: 8 Ln 17497, Col 1 INS
T4(R=Y%DF=Y%T=3B-45%TG=40%W=0%S=A%A=Z%F=R%=%RD=0%Q=)
T5(R=Y%DF=Y%T=3B-45%TG=40%W=0%S=Z%A=S+%F=AR%=%RD=0%Q=)
T6(R=Y%DF=Y%T=3B-45%TG=40%W=0%S=A%A=Z%F=R%=%RD=0%Q=)
T7(R=Y%DF=Y%T=3B-45%TG=40%W=0%S=Z%A=S+%F=AR%=%RD=0%Q=)
U1(DF=NNT=3B-45%TG=40%PL=164%UN=0%RIPL=G%RID=G%RIPCK=G%RUCK=G%RUD=G)
IE(DFI=1%T=3B-45%TG=40%CD=S)
|
# Coraid Linux NAS v. 6, debian based Storage System, made by CoRAID
Fingerprint CoRAID NAS device
Class CoRAID | embedded || storage-misc
SEQ(SP=BE-CE9GCD=1-%ISIR=C8-D2%TI=2%II=I%TS=8)
OPS(01=M1040ST11NW2%02=M1040ST11NW2%03=M1040NNT11NW2%04=M1040ST11NW2%05=M1040ST11NW2%
06=M1040ST11)
WIN(W1=309C%W2=309C%W3=309C%W4=309C%W5=309C%W6=309C)
ECN(R=Y%DF=Y%T=3B-45%TG=40%W=30C%O=0=M1040NNSNW2%CC=N%Q=)
T1(R=Y%DF=Y%T=3B-45%TG=40%S=0%A=S+%F=AS%RD=0%Q=)
T2(R=N)
T3(R=Y%DF=Y%T=3B-45%TG=40%W=309C%O=0=S+%F=AS%O=M1040ST11NW2%RD=0%Q=)
T4(R=Y%DF=Y%T=3B-45%TG=40%W=0%S=A%A=Z%F=R%=%RD=0%Q=)
T5(R=Y%DF=Y%T=3B-45%TG=40%W=0%S=Z%A=S+%F=AR%=%RD=0%Q=)
T6(R=Y%DF=Y%T=3B-45%TG=40%W=0%S=A%A=Z%F=R%O=%RD=0%Q=)
T7(R=Y%DF=Y%T=3B-45%TG=40%W=0%S=Z%A=S+%F=AR%=%RD=0%Q=)
U1(DF=NNT=3B-45%TG=40%PL=164%UN=0%RIPL=G%RID=G%RIPCK=G%RUCK=G%RUD=G)
IE(DFI=1%T=3B-45%TG=40%CD=S)
#
# CoyotePoint load balancer E250GX
Fingerprint Coyote Point E250GX Equalizer load balancer
Class Coyote Point | embedded || load balancer

```

Рис. 5: Содержимое файла nmap-os-db.

### nmap-service-probes

Этот файл содержит "пробы используемые для определения программ по

прослушиваемому порту (ключи -sV и -A). Данный о сервисах задаются при помощи нескольких директив:

- Exclude <port specification>
- Probe <protocol> <probename> <probestring>
- match <service> <pattern> [<versioninfo>]

Пример для mysql:

```

Applications ▾ Places ▾ gedit ▾ Mon 16:26
Open | Save | 1 |  |  |  |  |  |  |
nmap-service-probes
/usr/share/nmap
match medcart m|^PAR1\.75080000002B123456\?;\?\?;\?\?;\?\?;\?\?;\?\?;\\?0AC| p/Howard
Medical Med Display/ v/1.5.4.298

# https://www.kernel.org/pub/software/admin/mon/
match mon m|^520 invalid command\n$| p/mon service monitoring daemon/

match mysql m|^x10\0\0\x01\xff\x13\x04Bad handshake$| p/MySQL/ cpe:/a:mysql:mysql/
|
# Not sure if this is target MAC or scanner MAC
match ndv m|^NDV_([d.]+) ([0-9a-f][0-9a-f]:){5}[0-9a-f][0-9a-f]\n| p/Neocoretech
NDV/ v/$1/ cpe:/a:neocoretech:ndv:$1/

match netbackup m|^xeax\xdd\xbe\xef\0\0\0\x05\0\0\x000\0\0\x000\0\0..0\0\0\x08\0a\0f\0f
\0s\0p\0r\0n\0g\0\0\0\0\0\$| s/p/Veritas Netbackup Professional/

match nimp m|^V([d.]+)\r\nERROR 0\r\n$| p/Linux NetworX Network ICE Management
Protocol/ v/$1/ o/Linux/ cpe:/o:linux:linux_kernel/a

match nsi m|^%NSI\x91\xceWb\0\x08\x02\x04\x0f\x05\0\0| p/Cisco Network Spectrum
Interface/

# Alcatel Speedtouch ADSL Router
match ftp m|^220 Inactivity timer = \d+ seconds\. Use 'site idle <secs>' to change.\r
\n221 Goodbye \(\badly formatted command seen\)\. You uploaded 0 and downloaded 0 kbytes
\.\r\n221 Goodbye \(\badly formatted command seen\)\. You uploaded 0 and downloaded 0
kbytes\.\r\n\$| p/Alcatel Speedtouch ADSL router ftpd/ d/broadband router/
# bftpd 1.0.22 on Linux 2.4
match ftp m|^220 \r\n500 Unknown command: \"\"\"\\r\\n500 Unknown command: \"\"\"\r\n\$| p/
Plain Text ▾ Tab Width: 8 ▾ Ln 4966, Col 1 ▾ INS

```

Рис. 6: Содержимое файла nmap-service-probes.

### 3.5 Создание собственной сигнатуры

Попробуем добавить собственную сигнатуру в файл nmap-service-probes. Для этого напишем простейший сервер, который слушает порт 22000 и на входящее сообщение отправляет приветственное сообщение со своей версией (Исходный код находится в файле tcpserver.c).

Для определения данного сервера, добавим в файл nmap-service-probes следующие строки:

```

Probe TCP MyServer q|\x02Hi|
rarity 1
ports 22000
match testServer m|^Hello, client \((\w*) ([\d.]*))\)/ p/$1/ v/$2/

```

В данных строках мы описываем, какое сообщение будем отправлять для идентификации, на какой порт, а так же какой ответ мы будем ожидать.

Запустим сканирование интересующего нас порта. Результат показан на рисунке 7

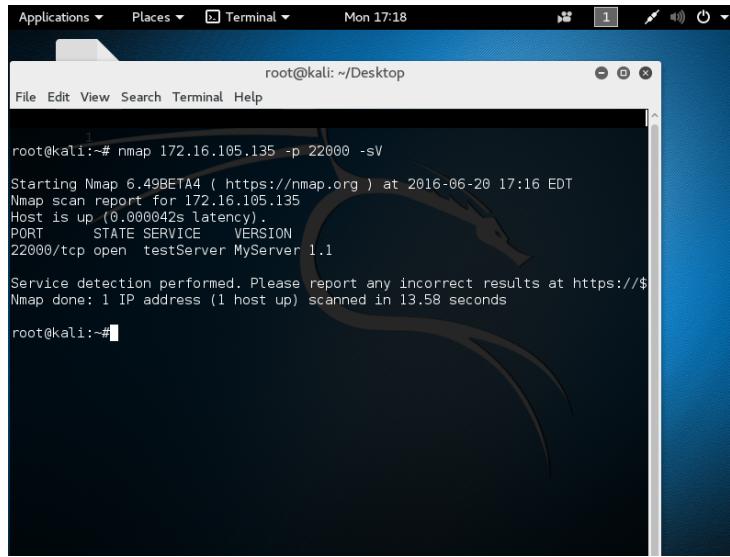


Рис. 7: Обнаружение тестовой утилиты.

### 3.6 Сохранение вывода в xml

Для сохранения вывода утилиты в файл xml необходимо воспользоваться **-oX <file>**. Например,

```
nmap -sV 127.16.105.135 -oX /home/nmap_sv_1.xml
```

### 3.7 Исследование работы Nmap с помощью Wireshark

Для исследования сетевой активности nmap воспользуемся утилитой Wireshark. Для примера отследим пакеты относящиеся к определению TCP сервиса, написанного нами ранее.

На Рисунках 8, 9 отображены TCP пакеты для определения версии сервиса. В исходящем пакете в передаваемых данных можно наблюдать текст "it\_it\_test\_tcp\_server" в ответном пакете - "yes\_it\_is".

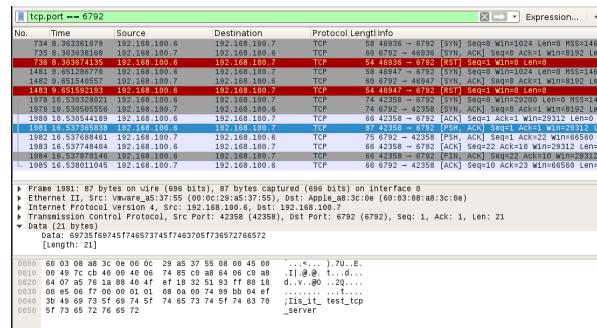


Рис. 8: Исходящий TCP пакет.

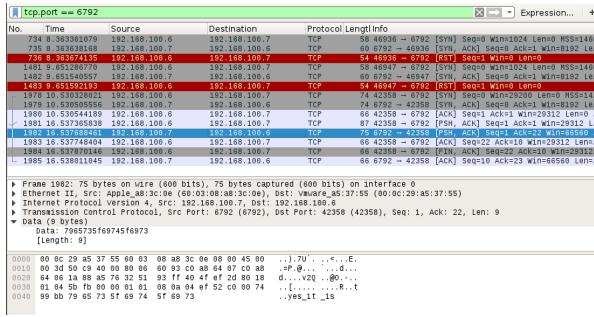


Рис. 9: Входящий TCP пакет.

### 3.8 Metasploit Framework

- инструмент для создания, тестирования и использования эксплойтов. Позволяет конструировать эксплойты с необходимой в конкретном случае «боевой нагрузкой» (payloads), которая выполняется в случае удачной атаки, например, установка shell или VNC сервера. Также фреймворк позволяет шифровать shellкод, что может скрыть факт атаки от IDS или IPS. Для проведения атаки необходима информация об установленных на удаленном сервере сервисах и их версии, то есть нужно дополнительное исследование с помощью таких инструментов, как nmap или nessus.

Проверим на уязвимости виртуальную машину Metasploitable2, используя db\_nmap (аналог nmap, сохраняющий результаты в БД) командой

```
db_nmap -A 192.168.100.9
```

Результат сканирования отображен на Рисунках 10, 11, 12

```
msf > db_nmap -A 192.168.100.9
[*] Nmap: Starting Nmap 7.01 ( https://nmap.org ) at 2016-03-27 07:14 EDT
[*] Nmap: Nmap scan report for 192.168.100.9
[*] Nmap: Host is up (0.00037s latency).
[*] Nmap: Not shown: 977 closed ports
[*] Nmap: PORT      STATE SERVICE      VERSION
[*] Nmap: 21/tcp    open  ftp          vsftpd 2.3.4
[*] Nmap: |_ftp-anon: Anonymous FTP login allowed (FTP code 230)
[*] Nmap: 22/tcp    open  ssh          OpenSSH 4.7p1 Debian 8ubuntu1 (protocol 2.0)
)
[*] Nmap: | ssh-hostkey:
[*] Nmap: |_ 1024 60:0f:cfc:e1:c0:5f:6a:74:d6:90:24:fa:c4:d5:6c:cd (DSA)
[*] Nmap: |_ 2048 56:56:24:0f:21:1d:de:a7:2b:ae:61:b1:24:3d:e8:f3 (RSA)
[*] Nmap: 23/tcp    open  telnet        Linux telnetd
[*] Nmap: 25/tcp    open  smtp         Postfix smtpd
[*] Nmap: |_smtp-commands: metasploitable.localdomain, PIPELINING, SIZE 10240000
, VRFY, ETRN, STARTTLS, ENHANCEDSTATUSCODES, 8BITMIME, DSN,
[*] Nmap: |_ssl-cert: Subject: commonName=ubuntu804-base.localdomain/organizationName=OCOSA/stateOrProvinceName=There is no such thing outside US/countryName=XX
[*] Nmap: |_Not valid before: 2010-03-17T14:07:45
[*] Nmap: |_Not valid after: 2010-04-16T14:07:45
[*] Nmap: |_ssl-date: 2016-03-26T18:57:23+00:00; -16h17m31s from scanner time.
[*] Nmap: 53/tcp    open  domain      ISC BIND 9.4.2
[*] Nmap: |_dns-nsid:
[*] Nmap: |_bind.version: 9.4.2
[*] Nmap: 80/tcp    open  http         Apache httpd 2.2.8 ((Ubuntu) DAV/2)
[*] Nmap: |_http-server-header: Apache/2.2.8 (Ubuntu) DAV/2
[*] Nmap: |_http-title: Metasploitable2 - Linux
```

Рис. 10: db\_nmap

```

[*] Nmap: 111/tcp open  rpcbind      2 (RPC #100000)
[*] Nmap: |_ rpcinfo:
[*] Nmap: |   program version  port/proto  service
[*] Nmap: |   100000  2          111/tcp  rpcbind
[*] Nmap: |   100000  2          111/udp  rpcbind
[*] Nmap: |   100003  2,3,4     2049/tcp  nfs
[*] Nmap: |   100003  2,3,4     2049/udp  nfs
[*] Nmap: |   100005  1,2,3     42841/tcp  mountd
[*] Nmap: |   100005  1,2,3     58287/udp  mountd
[*] Nmap: |   100021  1,3,4     44677/tcp  nlockmgr
[*] Nmap: |   100021  1,3,4     54520/udp  nlockmgr
[*] Nmap: |   100024  1          41270/tcp  status
[*] Nmap: |   100024  1          51630/udp  status
[*] Nmap: 139/tcp open  netbios-ssn  Samba smbd 3.X (workgroup: WORKGROUP)
[*] Nmap: 445/tcp open  netbios-ssn  Samba smbd 3.X (workgroup: WORKGROUP)
[*] Nmap: 512/tcp open  exec       netkit-rsh rexec
[*] Nmap: 513/tcp open  login?
[*] Nmap: 514/tcp open  tcpwrapped
[*] Nmap: 1099/tcp open  rmiregistry  GNU Classpath grmiregistry
[*] Nmap: |_ rmi-dumpregistry: Registry listing failed (No return data received from server)
[*] Nmap: 1524/tcp open  shell      Metasploitable root shell
[*] Nmap: 2049/tcp open  nfs        2-4 (RPC #100003)
[*] Nmap: 2121/tcp open  ftp        ProFTPD 1.3.1
[*] Nmap: 3306/tcp open  mysql     MySQL 5.0.51a-3ubuntu5
[*] Nmap: |_ mysql-info:
[*] Nmap: |   Protocol: 53
[*] Nmap: |   Version: .0.51a-3ubuntu5
[*] Nmap: |   Thread ID: 12
[*] Nmap: |   Capabilities flags: 43564
[*] Nmap: |   Some Capabilities: Support41Auth, SupportsCompression, Speaks41ProtocolNew, SwitchToSSLAfterHandshake, ConnectWithDatabase, SupportsTransactions, LongColumnFlag
[*] Nmap: |   Status: Autocommit
[*] Nmap: |   Salt: X^QZPc-sGFhZk*/IeBwn
[*] Nmap: 5432/tcp open  postgresql PostgreSQL DB 8.3.0 - 8.3.7

```

Рис. 11: db\_nmap

```

[*] Nmap: 5900/tcp open  vnc      VNC (protocol 3.3)
[*] Nmap: | vnc-info:
[*] Nmap: |   Protocol version: 3.3
[*] Nmap: |   Security types:
[*] Nmap: |     Unknown security type (33554432)
[*] Nmap: 6000/tcp open  X11      (access denied)
[*] Nmap: 6667/tcp open  irc      Unreal ircd
[*] Nmap: 8009/tcp open  ajp13    Apache Jserv (Protocol v1.3)
[*] Nmap: |_ajp-methods: Failed to get a valid response for the OPTION request
[*] Nmap: 8180/tcp open  http     Apache Tomcat/Coyote JSP engine 1.1
[*] Nmap: |_http-favicon: Apache Tomcat
[*] Nmap: |_http-server-header: Apache-Coyote/1.1
[*] Nmap: |_http-title: Apache Tomcat/5.5
[*] Nmap: MAC Address: 00:0C:29:77:6C:44 (VMware)
[*] Nmap: Device type: general purpose
[*] Nmap: Running: Linux 2.6.X
[*] Nmap: OS CPE: cpe:/o:linux:linux_kernel:2.6
[*] Nmap: OS details: Linux 2.6.9 - 2.6.33
[*] Nmap: Network Distance: 1 hop
[*] Nmap: Service Info: Hosts: metasploitable,localdomain,localhost, irc.Metasploitable.LAN; OSs: Unix, Linux; CPE: cpe:/o:linux:linux_kernel
[*] Nmap: Host script results:
[*] Nmap: |_nbstat: NetBIOS name: METASPOITABLE, NetBIOS user: <unknown>, NetBIOS MAC: <unknown> (unknown)
[*] Nmap: |_ smb-os-discovery:
[*] Nmap: |   OS: Unix (Samba 3.0.20-Debian)
[*] Nmap: |   NetBIOS computer name:
[*] Nmap: |   Workgroup: WORKGROUP
[*] Nmap: |_ System time: 2016-03-26T14:57:21-04:00
[*] Nmap: TRACEROUTE
[*] Nmap: HOP RTT      ADDRESS
[*] Nmap: 1  0.37 ms 192.168.100.9
[*] Nmap: OS and Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
[*] Nmap: Nmap done: 1 IP address (1 host up) scanned in 77.56 seconds

```

Рис. 12: db\_nmap

### 3.9 Примеры записей из nmap-service-probes

```
#####
# Detects TN3270 Servers which send IAC DO TTYPE on initial connection
# instead of IAC DO TN3270E
Probe TCP tn3270 q|\xff\xfb\x18\xff\xfa\x18\x00IBM-3279-4-E\xff\xf0|
rarity 8
ports 23,2323,2023,623
sslports 992
```

Согласно данной записи nmap для обнаружения сервиса использует протокол tcp для отправки пакета, содержащего

```
\xff\xfb\x18\xff\xfa\x18\x00IBM-3279-4-E\xff\xf0
```

Целевые порты 23, 2323, 2023, 623, ssl - 992. rarity - индикатор того, насколько часто возвращаемые пакеты содержат полезную информацию.

```
#####
Probe UDP AndroMouse q|AMSNIFF|
rarity 9
ports 8888

match AndroMouse m|^GOTBACK$|s p/AndroMouse Android remote mouse server/
```

Протокол - UDP, редкость полезных ответов - 9, порт - 8888. Данные для отправки

AMSNIFF

Шаблон ответа

```
m|^GOTBACK$|s
```

Дополнительная информация - "AndroMouse Android remote mouse server"

```
#####
Probe UDP AirHID q|from:airhid|
rarity 9
ports 13246
match AirHID m|^andReceiver-\d+\.\d+\.\d+$|s p/AirHID Andrioid remote mouse server/
```

Протокол - UDP, редкость полезных ответов - 9, порт - 13246. Данные для отправки

from:airhid

Шаблон ответа

```
m|^andReceiver-\d+\.\d+\.\d+$|s
```

Дополнительная информация - "AirHID Andrioid remote mouse server"

```
#####NEXT PROBE#####
# Queries z/OS Network Job Entry
# Sends an NJE Probe with the following information (text is converted to EBCDIC):
# TYPE      = OPEN
# OHOST     = FAKE
# RHOST     = FAKE
# RIP and OIP = 0.0.0.0
# R          = 0
# Based on http://www-01.ibm.com/support/knowledgecenter/SSLTBW_2.1.0/com.ibm.zos.v2r1.has
Probe TCP NJE q|\xd6\xd7\xc5\xd5@0@0\xc6\xc1\xd2\xc5@0@0\0\0\0\xc6\xc1\xd2\xc5@0@0\0\0\rarity 9
ports 175
sslports 2252
# If the port supports NJE it will respond with either a 'NAK' or 'ACK' in EBCDIC
match nje m|^xd5\xc1\xd2| p/IBM Network Job Entry (JES) /
match nje m|^xc1\xc3\xd2| p/IBM Network Job Entry (JES) /
```

Протокол - TCP, редкость полезных ответов - 9, порт - 175, ssl порт - 2252. Данные для отправки

\xd6\xd7\xc5\xd5@0@0\xc6\xc1\xd2\xc5@0@0\0\0\xc6\xc1\xd2\xc5@0@0\0\0\0\0

Шаблоны ответа

\xd5\xc1\xd2  
\xc1\xc3\xd2

Дополнительная информация для обоих шаблонов - "IBM Network Job Entry (JES)"

```
#####NEXT PROBE#####
# Sends a ServerInfo PBC request to the Basho Riak distributed database
Probe TCP riak-pbc q|\0\0\0\x01\x07|
rarity 8
ports 8087
match riak-pbc m|^....\x08..(riak@[w._-]+)...([\w._-]+)$|s p/Basho Riak/ v/$2/ h/$1/
```

Протокол - TCP, редкость полезных ответов - 8, порт - 8087. Данные для отправки

\0\0\0\x01\x07

Шаблоны ответа

^....\x08..(riak@[w.\_-]+)...([\w.\_-]+)\$

Дополнительная информация - "Basho Riak версия и имя хоста получаются из регулярного выражения.

### 3.10 Описание скрипта finger

В начале содержится описание скрипта

```

description = [
    Attempts to get a list of usernames via the finger service.
]

author = "Eddie Bell"

license = "Same as Nmap--See https://nmap.org/book/man-legal.html"

```

Категории, к которым принадлежит скрипт

```
categories = {"default", "discovery", "safe"}
```

Пример вывода

```
---
-- @output
-- PORT      STATE SERVICE
-- 79/tcp     open  finger
-- | finger:
-- | Welcome to Linux version 2.6.31.12-0.2-default at linux-pb94.site !
-- | 01:14am up 18:54, 4 users, load average: 0.14, 0.08, 0.01
-- |
-- | Login      Name          Tty      Idle  Login Time   Where
-- | Gutek      Ange Gutek    *:0       -      Wed 06:19  console
-- | Gutek      Ange Gutek    pts/1    18:54     Wed 06:20
-- | Gutek      Ange Gutek    *pts/0    -      Thu 00:41
-- | _Gutek     Ange Gutek    *pts/4    3      Thu 01:06
```

Подключение библиотек

```
require "comm"
require "shortport"
```

Проверка называется ли сервис "finger"или порт равен 79.

```
portrule = shortport.port_or_service(79, "finger")
```

nmap.new\_try создает обработчик исключений, comm.exchange - обрабатывает сетевые транзакции. В данном случае просиходит ожидание пока не получено хотя бы 100 строк, не менее 5 секунд или пока хост не закроет подключение.

```
action = function(host, port)
local try = nmap.new_try()

return try(comm.exchange(host, port, "\r\n",
{lines=100, proto=port.protocol, timeout=5000}))
end
```

## **4 Вывод**

В результате выполнения работы изучена утилита nmap. С помощью нее были просканированы хосты на уязвимости. Работа nmap была изучена с помощью утилиты Wireshark. Так же произведено знакомство с Metasploit framework.