

Санкт-Петербургский политехнический университет Петра Великого
Институт Информационных технологий и управления

Утилита для исследования сети и сканер портов Nmap

Выполнил: Сухинин А.А. гр. 53501/3 _____
Принял: Выглежанина К.Д. _____

2015 г.

1 Цель работы

Научиться сканировать хосты и порты, определять версии запущенных приложений.

2 Ход работы

Настройка

Предварительно были скачаны образы Kali Linux и Metasploitable 2. Данные образы развернуты на виртуальных машинах, которые включены в режиме "Сетевой мост". В сети также присутствуют и другие компьютеры: один рабочий, три других ПК, домашний сервер. Шлюз - это роутер по адресу 191.168.1.1

Провести поиск активных хостов

Поиск активных хостов можно произвести несколькими способами. Можно послать ICMP сообщение опрашивая все узлы либо попытаться просканировать популярные (1-1500) порты в диапазоне. Как правило, в современных сетях фильтруются ICMP пакеты, чтобы не предоставлять лишнюю информацию злоумышленнику и закрыть часть уязвимостей, таких как ping of death.

1. Стандартный ICMP ping

```
[*] exec: nmap -sn 192.168.1.*
```

```
Starting Nmap 6.47 ( http://nmap.org ) at 2015-05-24 11:58 EDT
Nmap scan report for router.asus.com (192.168.1.1)
Host is up (0.00052s latency).
MAC Address: 54:A0:50:83:A8:9C (Asustek Computer)
Nmap scan report for crazy_PC (192.168.1.25)
Host is up (0.000066s latency).
MAC Address: F4:6D:04:49:DC:FC (Asustek Computer)
Nmap scan report for 192.168.1.27
Host is up (0.044s latency).
MAC Address: 74:E5:43:65:15:F5 (Liteon Technology)
Nmap scan report for 192.168.1.35
Host is up (0.00021s latency).
MAC Address: 90:2B:34:DB:90:AD (Giga-byte Technology Co.)
Nmap scan report for crazy-mini (192.168.1.120)
Host is up (0.0046s latency).
MAC Address: C0:18:85:9E:54:0B (Hon Hai Precision Ind. Co.)
Nmap scan report for PODISH (192.168.1.132)
Host is up (0.017s latency).
MAC Address: 90:F6:52:6A:30:0D (Tp-link Technologies CO.)
Nmap scan report for kali (192.168.1.59)
Host is up.
Nmap done: 256 IP addresses (7 hosts up) scanned in 1.42 seconds
```

2. Сканирование основных портов

```
[*] exec: nmap 192.168.1.*
```

```
Starting Nmap 6.47 ( http://nmap.org ) at 2015-05-24 12:01 EDT
Nmap scan report for router.asus.com (192.168.1.1)
Host is up (0.0024s latency).
Not shown: 996 closed ports
PORT      STATE SERVICE
53/tcp    open  domain
80/tcp    open  http
```

```
1723/tcp open  pptp
9998/tcp open  distinct32
MAC Address: 54:A0:50:83:A8:9C (Asustek Computer)
```

```
Nmap scan report for crazy_PC (192.168.1.25)
Host is up (0.000058s latency).
Not shown: 986 closed ports
PORT      STATE SERVICE
135/tcp    open  msrpc
139/tcp    open  netbios-ssn
445/tcp    open  microsoft-ds
554/tcp    open  rtsp
1025/tcp   open  NFS-or-IIS
1026/tcp   open  LSA-or-nterm
1027/tcp   open  IIS
1030/tcp   open  iad1
1045/tcp   open  fpitp
1048/tcp   open  neod2
1049/tcp   open  td-postman
2869/tcp   open  icslap
5357/tcp   open  wsapi
10243/tcp  open  unknown
MAC Address: F4:6D:04:49:DC:FC (Asustek Computer)
```

...

Определить открытые порты

Для сканирования портов запустим уязвимую машину Metaspitable 2.

Можно просканировать основные открытые порты командой: `nmap 192.168.1.217`

Либо указать весь диапазон портов: `nmap 192.168.1.217 -p 1-65535`

```
[*] exec: nmap 192.168.1.217 -p 1-65535
```

```
Starting Nmap 6.47 ( http://nmap.org ) at 2015-05-24 12:08 EDT
```

```
Nmap scan report for 192.168.1.217
```

```
Host is up (0.00018s latency).
```

```
Not shown: 65505 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  ccproxy-ftp
3306/tcp   open  mysql
3632/tcp   open  distccd
5432/tcp   open  postgresql
```

```

5900/tcp open vnc
6000/tcp open X11
6667/tcp open irc
6697/tcp open unknown
8009/tcp open ajp13
8180/tcp open unknown
8787/tcp open unknown
39142/tcp open unknown
50756/tcp open unknown
55303/tcp open unknown
56967/tcp open unknown
MAC Address: 08:00:27:C0:D5:A0 (Cadmus Computer Systems)

```

Nmap done: 1 IP address (1 host up) scanned in 7.73 seconds

Определить версии сервисов

Для этого необходимо добавить ключ `-sV` к предыдущему пункту.

```
[*] exec: nmap 192.168.1.217 -p "*" -sV
```

Starting Nmap 6.47 (<http://nmap.org>) at 2015-05-24 12:15 EDT

Nmap scan report for 192.168.1.217

Host is up (0.00049s latency).

Not shown: 4219 closed ports

PORT	STATE	SERVICE	VERSION
21/tcp	open	ftp	vsftpd 2.3.4
22/tcp	open	ssh	OpenSSH 4.7p1 Debian 8ubuntu1 (protocol 2.0)
23/tcp	open	telnet	Linux telnetd
25/tcp	open	smtp	Postfix smtpd
53/tcp	open	domain	ISC BIND 9.4.2
80/tcp	open	http	Apache httpd 2.2.8 ((Ubuntu) DAV/2)
111/tcp	open	rpcbind	2 (RPC #100000)
139/tcp	open	netbios-ssn	Samba smbd 3.X (workgroup: WORKGROUP)
445/tcp	open	netbios-ssn	Samba smbd 3.X (workgroup: WORKGROUP)
512/tcp	open	exec?	
513/tcp	open	login	
514/tcp	open	shell?	
1099/tcp	open	rmiregistry	GNU Classpath grmiregistry
1524/tcp	open	shell	Metasploitable root shell
2049/tcp	open	nfs	2-4 (RPC #100003)
2121/tcp	open	ftp	ProFTPD 1.3.1
3306/tcp	open	mysql	MySQL 5.0.51a-3ubuntu5
3632/tcp	open	distccd	distccd v1 ((GNU) 4.2.4 (Ubuntu 4.2.4-1ubuntu4))
5432/tcp	open	postgresql	PostgreSQL DB 8.3.0 - 8.3.7
5900/tcp	open	vnc	VNC (protocol 3.3)
6000/tcp	open	X11	(access denied)
6667/tcp	open	irc	Unreal ircd
8009/tcp	open	ajp13	Apache Jserv (Protocol v1.3)
8180/tcp	open	http	Apache Tomcat/Coyote JSP engine 1.1

1 service unrecognized despite returning data.

If you know the service/version, please submit the following fingerprint

at <http://www.insecure.org/cgi-bin/servicefp-submit.cgi> :

SF-Port514-TCP:V=6.47%I=7%D=5/24%Time=5561F938%P=i686-pc-linux-gnu

%r(NULL,

SF:2B,"\\x01Couldn't\\x20get\\x20address\\x20for\\x20your\\x20host\\x20\\(kali\\)

SF:");

MAC Address: 08:00:27:C0:D5:A0 (Cadmus Computer Systems)
Service Info: Hosts: metasploitable.localdomain, localhost,
irc.Metasploitable.LAN; OSs: Unix, Linux; CPE: cpe:/o:linux:linux_kernel

Сохранить вывод утилиты в формате xml

Для этого необходимо добавить к команде ключ -oX имя файла.

Файл, полученный в результате исполнения команды "nmap 192.168.1.217 -p"*sV -oX /home/nmap.xml" лежит в каталоге с отчетом.

Изучить файлы nmap-services, nmap-os-db, nmap-service-probes

Для удобства файл nmap-service-probes выкачан в каталог с отчетом.

1. nmap-service-probes

Перечислим основные директивы, используемые в файле.

- (a) Probe <протокол> <имя> q"<посылаемая строка>"

Где в качестве протокола может быть указать TCP или UDP, имя - любой набор английских символов, а между "" указывается строка, посылаемая на сервер.

- (b) match <название сервиса> <шаблон> [<версия>]

Сравнивает ответ с шаблоном, в случае соответствия завершает сопоставление.

- (c) softmatch <название сервиса> <шаблон> [<версия>]

Аналогичен match, но не прекращает сопоставление в случае успеха.

- (d) totalwaitms <миллисекунды>

Время ожидания

2. nmap-os-db

Содержит набор отпечатков для каждой ОС представленных различными директивами.

Генерируются шесть пакетов специального вида, которые посылаются целевой машине с перерывом в 100 мс. Для получения результатов теста используются директивы SEQ, OPS, WIN и T1. Более подробную информацию можно получить по адресу <http://nmap.org/book/osdetect-methods.html>

- (a) SEQ - результаты последовательного анализа

- (b) OPS - флаги пакетов, полученных в ответ

- (c) WIN - размер окон

- (d) T1 - данные касательно ответа на первый пакет

Также отпечаток может содержать директивы T2-T7 посылающие пакеты различного вида. Например, без указания флагов, с указанием флагов SYN, FIN, URG, PSN; а также пакеты другого вида.

Кроме того, существует возможность тестировать указанный хост с помощью UDP пакетов (директива U1), а также множество других возможностей.

Модификация данного файла достаточно сложна и, как правило, производится крайне редко.

Пример отпечатка:

```
# BT2700HGV DSL Router version 5.29.107.19
Fingerprint 2Wire BT2700HG-V ADSL modem
Class 2Wire | embedded || broadband router
CPE cpe:/h:2wire:bt2700hg-v
SEQ(SP=6A-BE%GCD=1-6%ISR=96-A0%TI=I%CI=I%II=I%SS=S%TS=A)
OPS(O1=M5B4NNSWONNT11%O2=M578NNSWONNT11%O3=M28OWONNT11%
%O4=M218NNSWONNT11%O5=M218NNSWONNT11%O6=M109NNSWONNT11)
WIN(W1=8000%W2=8000%W3=8000%W4=8000%W5=8000%W6=8000)
ECN(R=Y%DF=Y%T=FA-104%TG=FF%W=8000%O=M5B4NNSWON%CC=N%Q=)
T1(R=Y%DF=Y%T=FA-104%TG=FF%S=0%A=S+%F=AS%RD=0%Q=)
```

```

T2(R=N)
T3(R=N)
T4(R=Y%DF=Y%T=FA-104%TG=FF%W=O%S=A%A=Z%F=R%O=%RD=E44A4E43%Q=)
T5(R=Y%DF=Y%T=FA-104%TG=FF%W=O%S=Z%A=S+%F=AR%O=%RD=1F59B3D4%Q=)
T6(R=Y%DF=Y%T=FA-104%TG=FF%W=O%S=A%A=Z%F=R%O=%RD=1F59B3D4%Q=)
T7(R=N)
U1(DF=Y%T=FA-104%TG=FF%IPL=70%UN=0%RIPL=G%RID=G%RIPCK=G%RUCK=G%RUD=G)
IE(DFI=Y%T=FA-104%TG=FF%CD=S)

```

3. nmap-services

Структура данного представлена в виде таблицы с тремя колонками.

Первая - имя сервиса.

Вторая - номер и тип порта.

Третья - как часто данный порт встречается.

Фрагмент файла:

```

sysstat 11/udp 0.000577 # Active Users
unknown 12/tcp 0.000063
daytime 13/tcp 0.003927

```

Выбрать пять записей из файла nmap-service-probes и описать их работу

Для дополнительной наглядности рассмотрим распознанные сервисы на Metasploitable 2

1. Рассмотрим распознавание сервиса Samba

```
139/tcp open netbios-ssn Samba smbd 3.X (workgroup: WORKGROUP)
```

Найдем соответствующую строку в файле

```

match netbios-ssn m=~\0\0\0.\xffSMB\0\0\0\0\0\x88..\0\0[-\w. ]*\0+@
\x06\0\0\x01\0\x11\x06\0.*(?:[^\0] | [^_A-Z0-9-]\0)((?:[-\w]\0){2,50})=s
p/Samba smbd/ v/3.X/ i/workgroup: $P(1)/

```

Как и было описано выше, строка состоит из директивы match, названия сервиса и шаблона. Шаблон состоит из регулярного выражения и строки для печати. К выражениям взятым в скобках, при печати можно обращаться как к параметрам. Данная директива сопоставляет ответ с регулярным выражением

```

~\0\0\0.\xffSMB\0\0\0\0\0\x88..\0\0[-\w. ]*\0+
@\x06\0\0\x01\0\x11\x06\0.*(?:[^\0] | [^_A-Z0-9-]\0)((?:[-\w]\0){2,50})

```

При этом, выражение подставленное вместо указанного ниже может быть использовано в качестве параметра при печати. Остальные игнорируются т.к. внутри скобок указан знак вопроса. (Прим. w - весь алфавит и цифры)

```
((?:[-\w]\0){2,50})
```

Последняя строка определяет результат при совпадении. Ключ r указывает имя продукта, ключ v - версию, а i - дополнительную информацию. При выводе дополнительной информации также используется вспомогательная функция P(), которая удаляет все непечатаемые символы из параметра.

```
p/Samba smbd/ v/3.X/ i/workgroup: $P(1)/
```

2. Probe TCP NULL q

Данная директива используется для тестирования TCP портов, ее название NULL. Видимо, это связано с тем, что она не передает никакой запрос серверу.

3. totalwaitms 6000

Данная строка означает, что максимальное время ожидания ответа равно шесть секунд.

4. Рассмотрим сопоставление для telnet

```
match telnet m|\xff\xfd\x18\xff\xfd \xff\xfd#\xff\xfd'$| p/Linux
telnetd/ o/Linux/ cpe:/o:linux:linux_kernel/a
```

Сравнивает ответ с последовательностью байт 0xff, 0xfd, 0x18, 0xff, 0xfd, 0xff, 0xfd, '#', 0xff, 0xfd, '', конец строки.

В случае успеха возвращает имя продукта Linux telnetd, ОС - Linux, cpe (Common platform enumeration) - o:linux:linux-kernel

5. Добавленные строчки:

```
Probe TCP HIYOU q|Hi, you!|
```

```
match simple tcp m|Hi!\r\nI'm Simple Server version ([0-9.]*)|
p/Simple Server/ v/$P(1)/
```

Первая строка посылает запрос на открытый TCP порт "Hi, you!".

В этом случае от сервера ожидается ответ:

```
Hi!
I'm Simple Server version X.X.X
```

Из ответа извлекается версия и возвращается в качестве ответа.

Пример использования nmap:

```
[*] exec: nmap 192.168.1.25 -p 1879 -sV
```

```
Starting Nmap 6.47 ( http://nmap.org ) at 2015-05-24 17:09 EDT
Nmap scan report for crazy_PC (192.168.1.25)
Host is up (0.00018s latency).
PORT      STATE SERVICE      VERSION
1879/tcp  open  SimpleServer Simple Server 1.0
MAC Address: F4:6D:04:49:DC:FC (Asustek Computer)
```

Service detection performed. Please report any incorrect results at <http://nmap.org/submit/> .

Nmap done: 1 IP address (1 host up) scanned in 6.23 seconds

Пример использования nmap без изменений:

```
[*] exec: nmap 192.168.1.25 -p 1879 -sV
```

```
Starting Nmap 6.47 ( http://nmap.org ) at 2015-05-24 17:19 EDT
Nmap scan report for crazy_PC (192.168.1.25)
Host is up (0.00024s latency).
PORT      STATE SERVICE      VERSION
1879/tcp  open  unknown
1 service unrecognized despite returning data. If you know the service/
version, please submit the following fingerprint at http://
www.insecure.org/cgi-bin/servicefp-submit.cgi :
SF-Port1879-TCP:V=6.47%I=7%D=5/24%Time=55624072%P=i686-pc-linux-gnu%r(Gene
SF:ricLines,5,"azaza")%r(GetRequest,5,"azaza")%r(HTTPOptions,5,"azaza")%r(
```

```
SF:RTSPRequest,5,"azaza")%r(RPCCheck,5,"azaza")%r(DNSVersionBindReq,5,"aza
SF:za")%r(DNSStatusRequest,5,"azaza")%r(Help,5,"azaza")%r(SSLSessionReq,
5,
SF:"azaza")%r(Kerberos,5,"azaza")%r(SMBProgNeg,5,"azaza")%r(X11Probe,
5,"az
SF:aza")%r(FourOhFourRequest,5,"azaza")%r(LPDString,5,"azaza")
%r(LDAPBindR
SF:eq,5,"azaza")%r(SIPOptions,5,"azaza")%r(LANDesk-RC,5,"azaza")
%r(Termina
SF:lServer,5,"azaza")%r(NCP,5,"azaza")%r(NotesRPC,5,"azaza")
%r(WMSRequest,
SF:5,"azaza")%r(oracle-tns,5,"azaza")%r(afp,5,"azaza")%r(kumo-server,
5,"az
SF:aza");
MAC Address: F4:6D:04:49:DC:FC (Asustek Computer)
```

Service detection performed. Please report any incorrect results at
 ://nmap.org/submit/ .
 Nmap done: 1 IP address (1 host up) scanned in 37.56 seconds

Сервер лежит в репозитории в каталоге programming

Выбрать один скрипт из состава Nmap и описать его работу

В качестве скрипта, для рассмотрения был выбран скрипт перебора паролей ftp. Для удобства данный скрипт помещен в каталог с отчетом.

nmap предоставляет мощный движок для написания скриптов (NSE). Языком написания скриптов является LUA. nmap предоставляет обширную коллекцию скриптов, которая находится в поддиректории scripts.

Как и большинство исходных файлов, скрипт начинается с импорта зависимостей. Затем следуют его описание и комментарии к использованию. После указания автора, лицензии и категории скрипта начинается значимый код.

Оставшийся код можно разделить на три части: Описание глобальных переменных, описание класса driver и использование движка перебора.

1. Глобальные переменные

В этой части объявляются переменные указывающие используемый порт и максимальный таймаут

2. Класс driver

Специального вида класс, с реализованным конструктором и методами connect, disconnect и login. В методах connect и disconnect производится управление сокетом - установка и закрытие соединения с хостом указанным в конструкторе. Метод login осуществляет попытку авторизации. В данном методе, по открытому соединению последовательно передаются команды USER * и PASS * и далее анализируются полученные ответы. В случае, если авторизация прошла успешно, метод возвращает true.

3. Функция action

В данной функции используется движок перебора паролей brute.Engine, которому в качестве параметров передаются имена пользователей и пароли, а также класс Driver.

Просканировать виртуальную машину Metasploitable2 используя db nmap из состава metasploit-framework

Предварительно необходимо включить postgresql и metasploit.

```
service postgresql start
service metasploit start
msfconsole
```

Затем использовать любую команду из перечисленных выше, но вместо nmap использовать db nmap. Все результаты будут занесены в базу данных. Таким образом, db nmap позволяет повторно использовать результаты и экономить большое количество времени.


```

msf > db_nmap -sn 192.168.1.*
[*] Nmap: Starting Nmap 6.47 ( http://nmap.org ) at 2015-05-24 18:30 EDT
[*] Nmap: Nmap scan report for router.asus.com (192.168.1.1)
[*] Nmap: Host is up (0.0011s latency).
[*] Nmap: MAC Address: 54:A0:50:83:A8:9C (Asustek Computer)
[*] Nmap: Nmap scan report for crazy_PC (192.168.1.25)
[*] Nmap: Host is up (0.000062s latency).
[*] Nmap: MAC Address: F4:6D:04:49:DC:FC (Asustek Computer)
[*] Nmap: Nmap scan report for 192.168.1.27
[*] Nmap: Host is up (0.22s latency).
[*] Nmap: MAC Address: 74:E5:43:65:15:F5 (Liteon Technology)
[*] Nmap: Nmap scan report for crazy_server (192.168.1.35)
[*] Nmap: Host is up (0.00039s latency).
[*] Nmap: MAC Address: 90:2B:34:DB:90:AD (Giga-byte Technology Co.)
[*] Nmap: Nmap scan report for crazy-mini (192.168.1.120)
[*] Nmap: Host is up (0.11s latency).
[*] Nmap: MAC Address: C0:18:85:9E:54:0B (Hon Hai Precision Ind. Co.)
[*] Nmap: Nmap scan report for PODISH (192.168.1.132)
[*] Nmap: Host is up (0.13s latency).
[*] Nmap: MAC Address: 90:F6:52:6A:30:0D (Tp-link Technologies CO.)
[*] Nmap: Nmap scan report for 192.168.1.217
[*] Nmap: Host is up (0.00013s latency).
[*] Nmap: MAC Address: 08:00:27:C0:D5:A0 (Cadmus Computer Systems)
[*] Nmap: Nmap scan report for kali (192.168.1.59)
[*] Nmap: Host is up.
[*] Nmap: Nmap done: 256 IP addresses (8 hosts up) scanned in 2.43 seconds

```

Исследовать различные этапы и режимы работы nmap с использованием утилиты Wireshark

При помощи утилиты wireshark были рассмотрены три случая работы nmap:

1. Сканирование локальной сети Как видно из примера ниже, сканирование локальной сети производится не при помощи ICMP пакетов, а при помощи ARP запросов.

При сканировании определенного узла wireshark отлавливает два пакета связанных с arp запросом и ответом.

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000000	CadmusCo_bf:fd:c9	Broadcast	ARP	42	Who has 10.0.0.30?
		Tell 10.0.0.20				

No.	Time	Source	Destination	Protocol	Length	Info
2	0.000256000	CadmusCo_f1:b5:af	CadmusCo_bf:fd:c9	ARP	60	10.0.0.30 is at 08:00:27:f1:b5:af

2. Успешное сканирование порта При сканировании порта nmap пытается установить соединение по указанному сокету. Установка TCP соединения, как известно, производится согласно правила трех рукопожатий. Анализируя второй шаг этого процесса можно определить состояние указанного сокета. Кроме того, следует отметить, что данному процессу предшествует обыкновенное сканирование, указанное в предыдущем пункте.

No.	Time	Source	Destination	Protocol	Length	Info
-----	------	--------	-------------	----------	--------	------

```

1 0.000000000 CadmusCo_bf:fd:c9
Broadcast      ARP      42      Who
has 10.0.0.30? Tell 10.0.0.20

```

```

No.      Time      Source
Destination      Protocol Length Info
2 0.000271000 CadmusCo_f1:b5:af
CadmusCo_bf:fd:c9 ARP      60
10.0.0.30 is at 08:00:27:f1:b5:af

```

```

No.      Time      Source
Destination      Protocol Length Info
3 0.001097000 10.0.0.20
10.0.0.30 TCP      58      57092
> ssh [SYN] Seq=0 Win=1024 Len=0 MSS=1460

```

```

No.      Time      Source
Destination      Protocol Length Info
4 0.001340000 10.0.0.30
10.0.0.20 TCP      60      ssh >
57092 [SYN, ACK] Seq=0 Ack=1 Win=5840 Len=0
MSS=1460

```

```

No.      Time      Source
Destination      Protocol Length Info
5 0.001351000 10.0.0.20
10.0.0.30 TCP      54      57092
> ssh [RST] Seq=1 Win=0 Len=0

```

Согласно правила трех рукопожатий ожидается следующая последовательность пакетов с флагами SYN, SYN+ACK, ACK. Как видно из примера, в ответ на попытку установки соединения (SYN) хост подтверждает установку соединения (SYN+ACK). Последним шагом должна являться отправка ACK, однако так как нет необходимости устанавливать соединение, в данном случае отправляется RST.

3. Неудачное сканирование порта Данный пример иллюстрирует сканирование закрытого порта. Как и в предыдущем случае, процесс протекает по тому же алгоритму, однако на втором шаге исследуемый хост отправляет пакет с флагом RST, сообщая о том, что порт закрыт.

```

No.      Time      Source
Destination      Protocol Length Info
1 0.000000000 CadmusCo_bf:fd:c9
Broadcast      ARP      42      Who
has 10.0.0.30? Tell 10.0.0.20

```

```

No.      Time      Source
Destination      Protocol Length Info
2 0.000218000 CadmusCo_f1:b5:af
CadmusCo_bf:fd:c9 ARP      60
10.0.0.30 is at 08:00:27:f1:b5:af

```

```

No.      Time      Source
Destination      Protocol Length Info
3 0.001017000 10.0.0.20
10.0.0.30 TCP      58      63464
> vlsi-lm [SYN] Seq=0 Win=1024 Len=0
MSS=1460

```

No.	Time	Source	Destination	Protocol	Length	Info
4	0.001245000	10.0.0.30	10.0.0.20	TCP	60	vlsi-
			lm > 63464	[RST, ACK]	Seq=1	Ack=1 Win=0
					Len=0	

3 Выводы

В ходе данной работы были изучены основные возможности nmap. Определение активных хостов, сканирование портов, определение версий сервисов, дополнение определения версий сервисов, были рассмотрены основные файлы используемые для определения версий сервисов и ОС. В качестве примера - один скрипт перебора паролей. Также была рассмотрена версия db nmap сохраняющая результаты в БД для последующего применения.

Инструмент nmap является мощным и гибким инструментом для сбора информации. При этом, не стоит забывать, что именно сбор информации определяет успех предстоящей атаки.