PREPARING A NETWORK ATTACK: "ENUMERATION"

Corso di Laurea Magistrale in Ingegneria Informatica

Prof. Simon Pietro Romano

spromano@unina.it

HOW TO PREPARE A NETWORK ATTACK?

- Vital concepts for anyone looking to prepare, with knowledge, for launching an attack on a computer network:
  - Footprinting:
    - The art of gathering information on the network
    - Often referred to as "network reconnaissance"
  - Scanning:
    - a thorough inspection of the attack perimeter, looking for potential entry points
  - Enumeration:
    - probing the identified services to discover potential vulnerabilities

ENUMERATION vs SCANNING

- The main difference between enumeration techniques and scanning techniques lies in the higher level of intrusiveness of the former
- Enumeration requires:
  - creating "active" connections to the analyzed systems
  - sending explicit "queries" to the services identified during scanning
- By its nature, enumeration is:
  - ☹ more dangerous than other information-gathering techniques
    - it provides access to detailed data about the identified services
  - ☺ more "traceable" and therefore more easily detectable by the security systems that the target organization is (hopefully) equipped with

INFORMATION SOUGHT

- User account names
  - for potential password guessing attacks
- Poorly configured shared resources
  - e.g., shared and inadequately protected file system folders
- Outdated software modules, potentially vulnerable
  - e.g., web servers vulnerable to buffer overflow attacks

CHARACTERISTICS OF ENUMERATION

- Enumeration techniques are closely tied to the specific characteristics of the hardware and software platforms being analyzed
- They depend heavily on the data gathered during scanning
  - often, the scanning and enumeration phases coexist and are conducted sequentially within the same tool:
    - Phase 1: port scanning (scanning)
    - Phase 2: banner grabbing on active services to determine the type of operating system running on the target (enumeration)

SERVICE FINGERPRINTING

- The next step after port scanning
  - once the open ports are identified…
  - …a detailed analysis of the associated services starts:
    - service version, revisions, level of applied patching
- Typically, this analysis is automated, utilizing techniques provided by tools like nmap
- Attackers usually resort to "manual" techniques only when they require extreme stealthiness in their attack to avoid detection

VERSION SCANNING WITH NMAP

- Usage of the "-sV" switch
  - Involves
    - querying the open ports
    - soliciting feedback
    - comparing the responses with a database of "signatures" associated with individual services and their versions/ implementations
  - This allows, e.g., the discovery of services listening on "non-default" ports

“SCANNING” vs “ENUMERATION” WITH *nmap*

```
root@kali:~# nmap -sS 143.225.28.169 -p 81

Starting Nmap 6.49BETA4 ( https://nmap.org ) at 2015-10-05 06:45 EDT
Nmap scan report for 143.225.28.169
Host is up (0.0013s latency).
PORT   STATE SERVICE
81/tcp open  hosts2-ns
MAC Address: 40:6C:8F:3C:31:E3 (Apple)

Nmap done: 1 IP address (1 host up) scanned in 0.43 seconds
root@kali:~# nmap -sV 143.225.28.169 -p 81

Starting Nmap 6.49BETA4 ( https://nmap.org ) at 2015-10-05 06:45 EDT
Nmap scan report for 143.225.28.169
Host is up (0.00068s latency).
PORT   STATE SERVICE VERSION
81/tcp open  http    Apache httpd 2.4.4 ((Unix) PHP/5.4.16 OpenSSL/1.0.1e mod_perl/2.0.8-dev Perl/v5.16.3)
MAC Address: 40:6C:8F:3C:31:E3 (Apple)

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 7.09 seconds
```

Existence of an HTTP server listening on port 81!

BEHIND THE SCENES: “SCANNING”

No.	Time	Source	Destination	Protocol	Length	Info
138	6.332911000	143.225.28.168	143.225.28.169	TCP	58	47293→81 [SYN] Seq=0 Win=1024 Len=0 MSS=1460
139	6.333859000	143.225.28.169	143.225.28.168	TCP	60	81→47293 [SYN, ACK] Seq=0 Ack=1 Win=65535 Len=0 MSS=1460
140	6.333881000	143.225.28.168	143.225.28.169	TCP	54	47293→81 [RST] Seq=1 Win=0 Len=0

```
▸ Frame 139: 60 bytes on wire (480 bits), 60 bytes captured (480 bits) on interface 0
▸ Ethernet II, Src: Apple_3c:31:e3 (40:6c:8f:3c:31:e3), Dst: CadmusCo_bf:ed:99 (08:00:27:bf:ed:99)
▸ Internet Protocol Version 4, Src: 143.225.28.169 (143.225.28.169), Dst: 143.225.28.168 (143.225.28.168)
▾ Transmission Control Protocol, Src Port: 81 (81), Dst Port: 47293 (47293), Seq: 0, Ack: 1, Len: 0
    Source Port: 81 (81)
    Destination Port: 47293 (47293)
    [Stream index: 3]
    [TCP Segment Len: 0]
    Sequence number: 0    (relative sequence number)
    Acknowledgment number: 1    (relative ack number)
    Header Length: 24 bytes
  ▸ .... 0000 0001 0010 = Flags: 0x012 (SYN, ACK)
    Window size value: 65535
    [Calculated window size: 65535]
  ▸ Checksum: 0x5e44 [validation disabled]
    Urgent pointer: 0
  ▾ Options: (4 bytes), Maximum segment size
    ▸ Maximum segment size: 1460 bytes
  ▾ [SEQ/ACK analysis]
      [This is an ACK to the segment in frame: 138]
      [The RTT to ACK the segment was: 0.000948000 seconds]
```

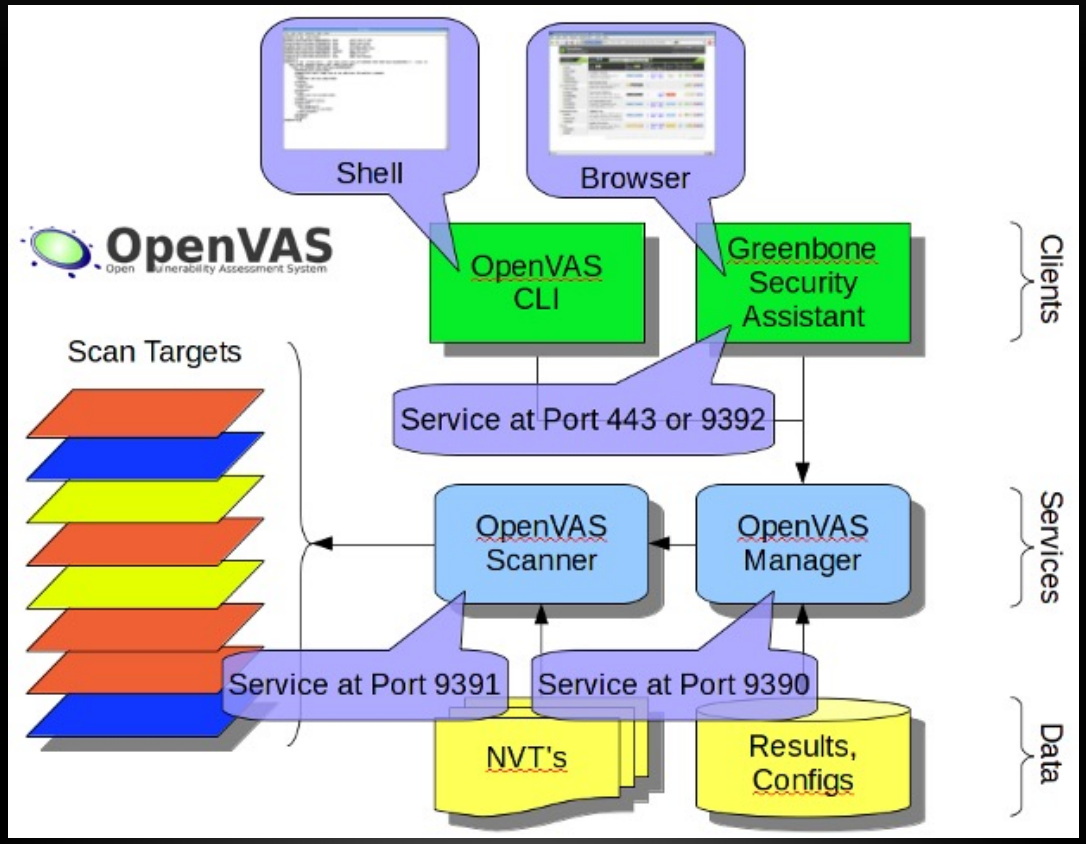
BEHIND THE SCENES : “ENUMERATION”

No.	Time	Source	Destination	Protocol	Length	Info
98	4.641790000	143.225.28.168	143.225.28.169	TCP	58	49769→81 [SYN] Seq=0 Win=1024 Len=0 MSS=1460
99	4.642583000	143.225.28.169	143.225.28.168	TCP	60	81→49769 [SYN, ACK] Seq=0 Ack=1 Win=65535 Len=0 MSS=1460
100	4.642590000	143.225.28.168	143.225.28.169	TCP	54	49769→81 [RST] Seq=1 Win=0 Len=0
104	4.742019000	143.225.28.168	143.225.28.169	TCP	58	49770→81 [SYN] Seq=0 Win=1024 Len=0 MSS=1460
105	4.742915000	143.225.28.169	143.225.28.168	TCP	60	81→49770 [SYN, ACK] Seq=0 Ack=1 Win=65535 Len=0 MSS=1460
106	4.742931000	143.225.28.168	143.225.28.169	TCP	54	49770→81 [RST] Seq=1 Win=0 Len=0
108	4.881624000	143.225.28.168	143.225.28.169	TCP	74	59451→81 [SYN] Seq=0 Win=29200 Len=0 MSS=1460 SACK_PERM=1 TSval=2772176 TSecr=0 WS=1024
109	4.882600000	143.225.28.169	143.225.28.168	TCP	78	81→59451 [SYN, ACK] Seq=0 Ack=1 Win=65535 Len=0 MSS=1460 WS=32 TSval=330830057 TSecr=2772176 SACK_PERM=1
110	4.882629000	143.225.28.168	143.225.28.169	TCP	66	59451→81 [ACK] Seq=1 Ack=1 Win=29696 Len=0 TSval=2772176 TSecr=330830057
111	4.883466000	143.225.28.169	143.225.28.168	TCP	66	[TCP Window Update] 81→59451 [ACK] Seq=1 Ack=1 Win=131744 Len=0 TSval=330830057 TSecr=2772176
244	10.88972500	143.225.28.168	143.225.28.169	HTTP	84	GET / HTTP/1.0
245	10.89056800	143.225.28.169	143.225.28.168	TCP	66	81→59451 [ACK] Seq=1 Ack=19 Win=131744 Len=0 TSval=330836055 TSecr=2773678
246	10.89234800	143.225.28.169	143.225.28.168	HTTP	460	HTTP/1.1 302 Found (text/html)
247	10.89236500	143.225.28.168	143.225.28.169	TCP	66	59451→81 [ACK] Seq=19 Ack=395 Win=30720 Len=0 TSval=2773678 TSecr=330836056
248	10.89238100	143.225.28.168	143.225.28.169	TCP	66	81→59451 [FIN, ACK] Seq=395 Ack=19 Win=131744 Len=0 TSval=330836056 TSecr=2773678
249	10.89277000	143.225.28.168	143.225.28.169	TCP	66	59451→81 [FIN, ACK] Seq=19 Ack=396 Win=30720 Len=0 TSval=2773678 TSecr=330836056
250	10.89279100	143.225.28.169	143.225.28.168	TCP	66	[TCP Out-Of-Order] 81→59451 [FIN, ACK] Seq=395 Ack=19 Win=131744 Len=0 TSval=330836056 TSecr=2773678
251	10.89279100	143.225.28.169	143.225.28.168	TCP	78	[TCP Dup ACK 249#1] 59451→81 [ACK] Seq=20 Ack=396 Win=30720 Len=0 TSval=2773679 TSecr=330836056 SLE=395 SRE=
252	10.89297400	143.225.28.169	143.225.28.168	TCP	66	81→59451 [ACK] Seq=396 Ack=20 Win=131744 Len=0 TSval=330836057 TSecr=2773678
253	10.96664200	143.225.28.168	143.225.28.169	TCP	74	59452→81 [SYN] Seq=0 Win=29200 Len=0 MSS=1460 SACK_PERM=1 TSval=2773697 TSecr=0 WS=1024
254	10.96763400	143.225.28.169	143.225.28.168	TCP	78	81→59452 [SYN, ACK] Seq=0 Ack=1 Win=65535 Len=0 MSS=1460 WS=32 TSval=330836130 TSecr=2773697 SACK_PERM=1
255	10.96764700	143.225.28.168	143.225.28.169	TCP	66	59452→81 [ACK] Seq=1 Ack=1 Win=29696 Len=0 TSval=2773697 TSecr=330836130
256	10.96825900	143.225.28.169	143.225.28.168	TCP	66	[TCP Window Update] 81→59452 [ACK] Seq=1 Ack=1 Win=131744 Len=0 TSval=330836130 TSecr=2773697
257	10.96855600	143.225.28.168	143.225.28.169	HTTP	84	GET / HTTP/1.0
258	10.96925200	143.225.28.169	143.225.28.168	TCP	66	81→59452 [ACK] Seq=1 Ack=19 Win=131744 Len=0 TSval=330836131 TSecr=2773697
259	10.97052300	143.225.28.169	143.225.28.168	HTTP	460	HTTP/1.1 302 Found (text/html)
260	10.97053900	143.225.28.168	143.225.28.169	TCP	66	59452→81 [ACK] Seq=19 Ack=395 Win=30720 Len=0 TSval=2773698 TSecr=330836132
261	10.97056600	143.225.28.169	143.225.28.168	TCP	66	81→59452 [FIN, ACK] Seq=395 Ack=19 Win=131744 Len=0 TSval=330836132 TSecr=2773697
262	10.97086000	143.225.28.168	143.225.28.169	TCP	74	59453→81 [SYN] Seq=0 Win=29200 Len=0 MSS=1460 SACK_PERM=1 TSval=2773698 TSecr=0 WS=1024
263	10.97105900	143.225.28.169	143.225.28.168	TCP	66	[TCP Out-Of-Order] 81→59452 [FIN, ACK] Seq=395 Ack=19 Win=131744 Len=0 TSval=330836132 TSecr=2773698
264	10.97106400	143.225.28.168	143.225.28.169	TCP	78	59452→81 [ACK] Seq=19 Ack=396 Win=30720 Len=0 TSval=2773698 TSecr=330836132 SLE=395 SRE=396
265	10.97138200	143.225.28.169	143.225.28.168	TCP	78	81→59453 [SYN, ACK] Seq=0 Ack=1 Win=65535 Len=0 MSS=1460 WS=32 TSval=330836133 TSecr=2773698 SACK_PERM=1
266	10.97139400	143.225.28.168	143.225.28.169	TCP	66	59453→81 [ACK] Seq=1 Ack=1 Win=29696 Len=0 TSval=2773698 TSecr=330836133
267	10.97140300	143.225.28.169	143.225.28.168	TCP	66	[TCP Dup ACK 263#1] 81→59452 [ACK] Seq=396 Ack=19 Win=131744 Len=0 TSval=330836133 TSecr=2773698
268	10.97152000	143.225.28.168	143.225.28.169	HTTP	106	GET / HTTP/1.1
269	10.97188800	143.225.28.169	143.225.28.168	TCP	66	[TCP Window Update] 81→59453 [ACK] Seq=1 Ack=1 Win=131744 Len=0 TSval=330836133 TSecr=2773698
270	10.97189300	143.225.28.169	143.225.28.168	TCP	66	81→59453 [ACK] Seq=1 Ack=41 Win=131712 Len=0 TSval=330836133 TSecr=2773698
271	10.97248600	143.225.28.169	143.225.28.168	HTTP	322	HTTP/1.1 302 Found
272	10.97249100	143.225.28.168	143.225.28.169	TCP	66	59453→81 [ACK] Seq=41 Ack=257 Win=30720 Len=0 TSval=2773698 TSecr=330836134
273	10.97273400	143.225.28.168	143.225.28.169	TCP	66	59452→81 [FIN, ACK] Seq=19 Ack=396 Win=30720 Len=0 TSval=2773698 TSecr=330836133
274	10.97274600	143.225.28.168	143.225.28.169	TCP	66	59453→81 [FIN, ACK] Seq=41 Ack=257 Win=30720 Len=0 TSval=2773699 TSecr=330836134

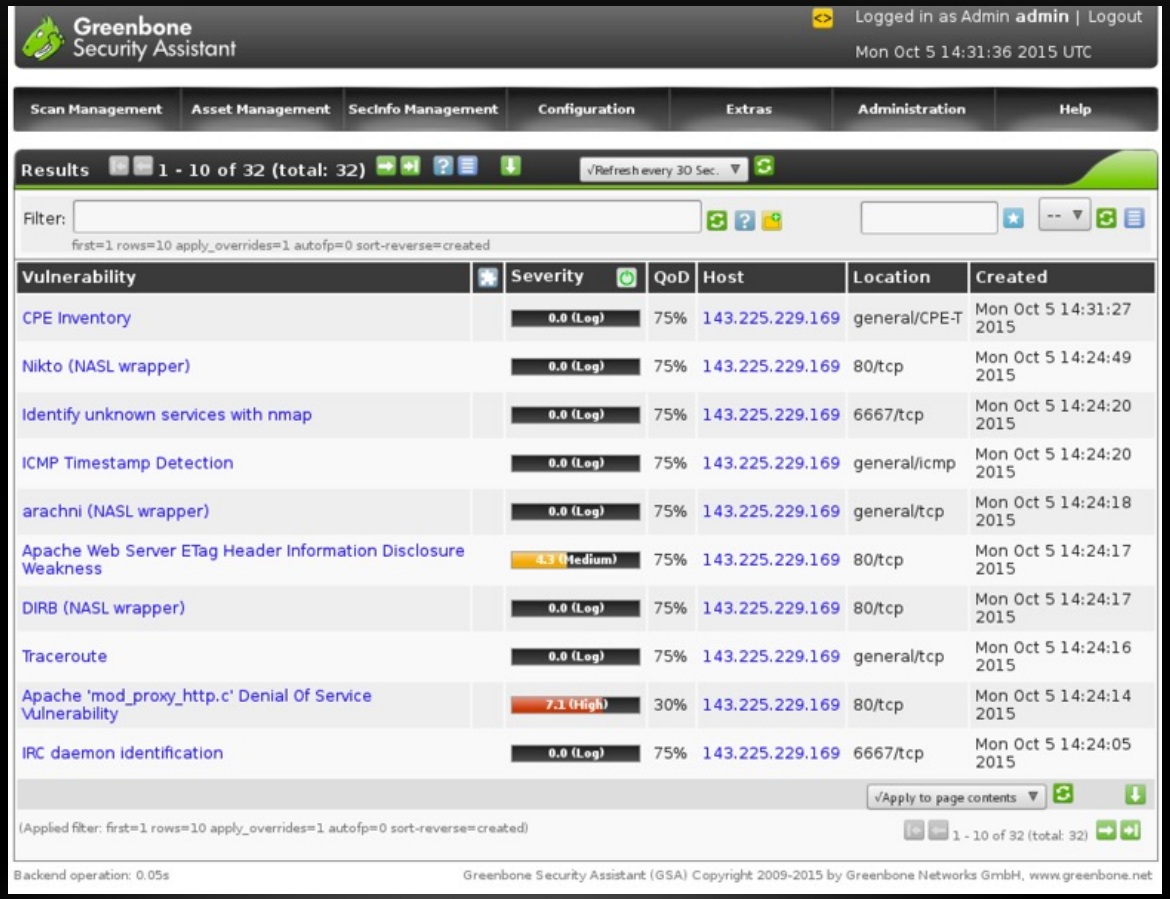
VULNERABILITY SCANNERS

- Typically used when there is little concern about covering up the traces of scanning activity
- They rely on the collection and updating of signatures of known vulnerabilities related to all types of processes potentially listening on a network port:
  - operating systems, network services, web applications, databases, etc.
- Many tools are available for this purpose, both commercially and in the open-source community:
  - OpenVAS → Open Vulnerability Assessment System

OpenVAS: GENERAL ARCHITECTURE



OpenVAS IN ACTION



OpenVAS: REPORTS

Scan Report

October 19, 2021

Summary

This document reports on the results of an automatic security scan. All dates are displayed using the timezone "Coordinated Universal Time", which is abbreviated "UTC". The task was "Immediate scan of IP 143.225.229.130". The scan started at Tue Oct 19 10:19:46 2021 UTC and ended at Tue Oct 19 11:00:15 2021 UTC. The report first summarises the results found. Then, for each host, the report describes every issue found. Please consider the advice given in each description, in order to rectify the issue.

Contents

2.1.1 High 22/tcp

High (CVSS: 7.8)
NVT: OpenSSH Denial of Service And User Enumeration Vulnerabilities (Linux)

Product detection result
`cpe:/a:openbsd:openssh:6.6p1`
`Detected by SSH Server type and version (OID: 1.3.6.1.4.1.25623.1.0.10267)`

Summary
This host is installed with openssh and is prone to denial of service and user enumeration vulnerabilities.

Vulnerability Detection Result
`Installed version: 6.6p1`
`Fixed version: 7.3`

Impact
Successfully exploiting this issue allows remote attackers to cause a denial of service (crypt CPU consumption) and to enumerate users by leveraging the timing difference between responses when a large password is provided.
Impact Level: Application

Solution
Solution type: VendorFix
Upgrade to OpenSSH version 7.3 or later. For updates refer to http://www.openssh.com

Affected Software/OS
OpenSSH versions before 7.3 on Linux

NMAP SCRIPTING ENGINE (NSE)

- An interface that allows nmap users to extend its capabilities through scripts written in the Lua programming language
- Enabling tasks such as sending and receiving data, generating reports, and more
- Nmap comes with many default scripts for various purposes:
  - network discovery, detecting the version of network services, uncovering backdoors and…
  - …exploiting vulnerabilities through attack payloads

BANNER GRABBING

- The simplest form of enumeration
- It involves connecting to a remote service and observing its output
- This activity provides valuable information:
  - the type of active service, service version, the presence of plugins or additional modules, etc.
- It can be manually executed using two useful tools:
  - telnet
  - netcat

BANNER GRABBING WITH NETCAT

```
root@kali:~# vim snippet.txt
root@kali:~# nc -nvv -o banners.txt 143.225.28.169 80 < snippet.txt
(UNKNOWN) [143.225.28.169] 80 (http) open
HTTP/1.1 302 Found
Date: Mon, 05 Oct 2015 15:30:05 GMT
Server: Apache/2.4.4 (Unix) PHP/5.4.16 OpenSSL/1.0.1e mod_perl/2.0.8-dev Perl/v5.16.3
X-Powered-By: PHP/5.4.16
Location: http:///xampp/
Content-Length: 131
Connection: close
Content-Type: text/html

<br />
<b>Notice</b>:  Undefined index: HTTP_HOST in <b>/Applicati
 sent 17, rcvd 394
```

```
root@kali:~# more snippet.txt
GET / HTTP/1.0


root@kali:~#
```

```
root@kali:~# more banners.txt
> 00000000 47 45 54 20 2f 20 48 54 54 50 2f 31 2e 30 0a 0a # GET / HTTP/1.0..
> 00000010 0a                                              # .
< 00000000 48 54 54 50 2f 31 2e 31 20 33 30 32 20 46 6f 75 # HTTP/1.1 302 Fou
< 00000010 6e 64 0d 0a 44 61 74 65 3a 20 4d 6f 6e 2c 20 30 # nd..Date: Mon, 0
< 00000020 35 20 4f 63 74 20 32 30 31 35 20 31 35 3a 33 30 # 5 Oct 2015 15:30
< 00000030 3a 30 35 20 47 4d 54 0d 0a 53 65 72 76 65 72 3a # :05 GMT..Server:
< 00000040 20 41 70 61 63 68 65 2f 32 2e 34 2e 34 20 28 55 #  Apache/2.4.4 (U
< 00000050 6e 69 78 29 20 50 48 50 2f 35 2e 34 2e 31 36 20 # nix) PHP/5.4.16
< 00000060 4f 70 65 6e 53 53 4c 2f 31 2e 30 2e 31 65 20 6d # OpenSSL/1.0.1e m
< 00000070 6f 64 5f 70 65 72 6c 2f 32 2e 30 2e 38 2d 64 65 # od_perl/2.0.8-de
< 00000080 76 20 50 65 72 6c 2f 76 35 2e 31 36 2e 33 0d 0a # v Perl/v5.16.3..
< 00000090 58 2d 50 6f 77 65 72 65 64 2d 42 79 3a 20 50 48 # X-Powered-By: PH
< 000000a0 50 2f 35 2e 34 2e 31 36 0d 0a 4c 6f 63 61 74 69 # P/5.4.16..Locati
< 000000b0 6f 6e 3a 20 68 74 74 70 3a 2f 2f 2f 78 61 6d 70 # on: http:///xamp
< 000000c0 70 2f 0d 0a 43 6f 6e 74 65 6e 74 2d 4c 65 6e 67 # p/..Content-Leng
< 000000d0 74 68 3a 20 31 33 31 0d 0a 43 6f 6e 6e 65 63 74 # th: 131..Connect
< 000000e0 69 6f 6e 3a 20 63 6c 6f 73 65 0d 0a 43 6f 6e 74 # ion: close..Cont
< 000000f0 65 6e 74 2d 54 79 70 65 3a 20 74 65 78 74 2f 68 # ent-Type: text/h
< 00000100 74 6d 6c 0d 0a 0d 0a 3c 62 72 20 2f 3e 0a 3c 62 # tml....<br />.<b
< 00000110 3e 4e 6f 74 69 63 65 3c 2f 62 3e 3a 20 20 55 6e # >Notice</b>:  Un
< 00000120 64 65 66 69 6e 65 64 20 69 6e 64 65 78 3a 20 48 # defined index: H
< 00000130 54 54 50 5f 48 4f 53 54 20 69 6e 20 3c 62 3e 2f # TTP_HOST in <b>/
< 00000140 41 70 70 6c 69 63 61 74 69 6f 6e 73 2f 58 41 4d # Applications/XAM
< 00000150 50 50 2f 78 61 6d 70 70 66 69 6c 65 73 2f 68 74 # PP/xamppfiles/ht
< 00000160 64 6f 63 73 2f 69 6e 64 65 78 2e 70 68 70 3c 2f # docs/index.php</
< 00000170 62 3e 20 6f 6e 20 6c 69 6e 65 20 3c 62 3e 37 3c # b> on line <b>7<
< 00000180 2f 62 3e 3c 62 72 20 2f 3e 0a                   # /b><br />.
root@kali:~#
```

COMMON NETWORK SERVICES: FTP (TCP PORT 21)

```
root@kali:~# ftp ftp.unina.it
Connected to ftp.unina.it.
220 ftp.unina.it NcFTPd Server (free educational license) ready.
Name (ftp.unina.it:root): anonymous
331 Guest login ok, send your complete e-mail address as password.
Password:
230-You are user #3 of 50 simultaneous users allowed.
230-
230 Logged in anonymously.
Remote system type is UNIX.
Using binary mode to transfer files.
ftp> ls
200 PORT command successful.
150 Opening ASCII mode data connection for /bin/ls.
drwxr-xr-x   2 ftpuser  ftpusers   1808 Sep  5  2011 images
-rw-r--r--   1 ftpuser  ftpusers  28319 Sep  6  2011 index.html
-rw-r--r--   1 ftpuser  ftpusers  24855 Jan 14  2011 index2.html
-rw-r--r--   1 ftpuser  ftpusers  28252 Sep  5  2011 indexprova.html
-rw-r--r--   1 ftpuser  ftpusers  28305 Sep  5  2011 indexprova.html-2
drwxr-xr-x  10 ftpuser  ftpusers    360 May  2  2011 pub
226 Listing completed.
ftp>
```

FTP: COUNTERMEASURES

- FTP (File Transfer Protocol):
  - one of those services that are considered so insecure today that the only recommended countermeasure is to discontinue their use!
- An alternative to the "plain" FTP service is Secure FTP (SFTP), which is based on SSH (Secure Shell) encryption
- In cases where you still want to offer FTP access to users, it is essential to follow some basic precautions, such as not allowing anonymous logins

COMMON NETWORK SERVICES: TELNET (TCP PORT 23)

```
root@kali:~# telnet 143.225.229.254
Trying 143.225.229.254...
Connected to 143.225.229.254.
Escape character is '^]'.

C
     ------------------------------------------------------
          Universita' degli Studi di Napoli "Federico II"
          CSI - Centro di ateneo per i Servizi Informativi
                     Facolta' di Ingegneria
                      Campus di Via Claudio

                       Cisco Catalyst 6509

          Ogni tentativo di accesso non autorizzato a
          questo sistema e' un reato perseguibile ai sensi
          dell'art. 615-ter del C.P.
     ------------------------------------------------------

Username:
```

NB: Sometimes, it's the banner itself that tells us what kind of system we are contacting!

COMMON NETWORK SERVICES: SMTP (TCP PORT 25)

```
root@kali:~# telnet mail.unina.it 25
Trying 192.132.34.73...
Connected to mail.unina.it.
Escape character is '^]'.
220 smtp1.unina.it ESMTP Sendmail 8.14.4/8.14.4; Mon, 5 Oct 2015 18:04:22 +0200
vrfy spromano@unina.it
252 2.1.5 <spromano@unina.it>
vrfy ciccio@unina.it
550 5.0.0 ciccio@unina.it... User unknown
quit
221 2.0.0 smtp1.unina.it closing connection
Connection closed by foreign host.
root@kali:~#
```

- A basic form of “account enumeration”!
  - “spromano” is a valid user…
  - …“ciccio” IS NOT!

COMMON NETWORK SERVICES: DNS (UDP/TCP, PORT 53)

```
root@kali:~# fierce -dns unina.it
DNS Servers for unina.it:
        dscna1.unina.it
        dscna2.unina.it

Trying zone transfer first...
        Testing dscna1.unina.it
                Request timed out or transfer not allowed.
        Testing dscna2.unina.it
                Request timed out or transfer not allowed.

Unsuccessful in zone transfer (it was worth a shot)
Okay, trying the good old fashioned way... brute force

Checking for wildcard DNS...
Nope. Good.
Now performing 2280 test(s)...
143.225.5.200   apps.unina.it
192.132.34.4    web.unina.it
192.132.34.5    vftp.unina.it
192.132.34.8    pmxln.unina.it
192.132.34.9    pmx3.unina.it
192.132.34.12   ssoiam.unina.it

Subnets found (may want to probe here using nmap or unicornscan):
        127.0.0.0-255 : 1 hostnames found.
        143.225.148.0-255 : 1 hostnames found.
        143.225.163.0-255 : 2 hostnames found.
        143.225.172.0-255 : 1 hostnames found.
        143.225.19.0-255 : 1 hostnames found.
        143.225.200.0-255 : 1 hostnames found.
        143.225.215.0-255 : 1 hostnames found.
        143.225.5.0-255 : 1 hostnames found.
        143.225.58.0-255 : 1 hostnames found.
        172.29.0.0-255 : 1 hostnames found.
        192.132.34.0-255 : 78 hostnames found.
        192.133.28.0-255 : 7 hostnames found.

Done with Fierce scan: http://ha.ckers.org/fierce/
Found 96 entries.

Have a nice day.
```

- Main issue:
  - Zone Transfer
    - see lecture on footprinting…
- Even when Zone Transfer is disabled:
  - reverse lookup, brute forcing, etc.
- Lots of tools for automating the overall process:
  - es: *fierce*

COMMON NETWORK SERVICES: TFTP (UDP/TCP, PORT 69)

- Trivial File Transfer Protocol (TFTP):
  - the simplest form of file transfer on a network
  - typically configured to work on UDP port 69
  - the basic assumption is that to download a file from the server:
    - you need to know its name…
    - …and there is no authentication required!
- It is rarely enabled on network nodes due to its lack of security features, but it is still widely used, even on routers and switches
- Typical configuration file names available on routers are:
  - *"running-config"*, "*startup-config*", "*config*", *"cisco-config"*, etc.

COMMON NETWORK SERVICES: Finger (UDP/TCP, PORT 79)

- The classic way to provide automated user information on the early Internet when everyone was more trusting
- Typically disabled in modern network systems

```
[root$]finger -l @target.example.com
[target.example.com]
Login: root                             Name: root
Directory: /root                        Shell: /bin/bash
On since Sun Mar 28 11:01 (PST) on tty1 11 minutes idle
     (messages off)
On since Sun Mar 28 11:01 (PST) on ttyp0 from :0.0
   3 minutes 6 seconds idle
No mail.
plan:
John Smith
Security Guru
```

```
root@kali:~# finger @143.225.28.244

Integrated port
Printer Type: Lexmark T644
Print Job Status: No Job Currently Active
Printer Status: 0 Ready
root@kali:~#
```

COMMON NETWORK SERVICES: HTTP (TCP, PORT 80)

- "manual" approach: *netcat* as the usual suspect
  - the HEAD method already provides lots of useful data…

```
root@kali:~# nc www.unina.it 80
HEAD / HTTP/1.1
Host: www.unina.it

HTTP/1.1 301 Moved Permanently
Date: Mon, 05 Oct 2015 18:05:28 GMT
Set-Cookie: JSESSIONID=9310ADFCD08E953244C280103B7A52BB.node_staging11; Path=/; HttpOnly
Set-Cookie: GUEST_LANGUAGE_ID=it_IT; Expires=Tue, 04-Oct-2016 18:05:33 GMT; Path=/
Set-Cookie: COOKIE_SUPPORT=true; Expires=Tue, 04-Oct-2016 18:05:33 GMT; Path=/
Location: http://www.unina.it/home;jsessionid=9310ADFCD08E953244C280103B7A52BB.node_staging11
Content-Type: text/html;charset=ISO-8859-1
Content-Length: 336
Connection: close
```

```
root@kali:~# nc www.unina.it 80
HEAD /home;jsessionid=9310ADFCD08E953244C280103B7A52BB.node_staging11 HTTP/1.1
Host: www.unina.it

HTTP/1.1 200 OK
Date: Mon, 05 Oct 2015 18:06:25 GMT
Set-Cookie: COOKIE_SUPPORT=true; Expires=Tue, 04-Oct-2016 18:06:31 GMT; Path=/
Liferay-Portal: Liferay Portal Enterprise Edition 6.1.20 EE (Paton / Build 6120 / July 31, 2012)
ETag: "0"
Set-Cookie: COOKIE_SUPPORT=true; Expires=Tue, 04-Oct-2016 18:06:31 GMT; Path=/
Content-Type: text/html;charset=UTF-8
Content-Length: 32
Connection: close
```

MICROSOFT RPC (MSRPC): TCP, PORT 135

- Remote Procedure Call (RPC) endpoint mapper:
  - used to provide information about the presence of services/applications on the target (Microsoft) machine

```
root@kali:~# nmap 143.225.XXX.XXX -script=msrpc-enum
```

```
Host script results:
| msrpc-enum:
|
|     uuid: d95afe70-a6d5-4259-822e-2c84da1ddb0d
|     tcp port: 49152
|     ip_addr: 0.0.0.0
|
|     uuid: 4b112204-e19-11d3-b42b-0000f81feb9f
|     ncalrpc: LRPC-51de3ed2060d22ec0d
|
|     netbios: \\GREEN-PC
|     uuid: b58aa02e-2884-4e97-8176-4ee06d794184
|     ncacn_np: \pipe\trkwks
```

NETBIOS NAME SERVICE: UDP, PORT 137

- NetBIOS Name Service (NBNS):
  - a distributed naming system for Microsoft networks no longer necessary, from Windows 2000 onwards, as it was replaced by the standard approach (DNS)...
  - ...but still enabled by default in almost all Windows distributions
- Enumeration is straightforward in this case:
  - simple UDP "poll" messages are sent to port 137 on the network

NETBIOS NAME SERVICE: TOOLS

- **“net view”**
  - identifies all Microsoft domains in a network or all computers in a domain
- **“nltest”**
  - identifies the Domain Controllers (which hold authentication information) of a specific domain
- **“nbtstat”**
  - allows connection to individual machines in a domain to retrieve the "name table", which includes:
    - system name, the domain to which the system belongs, active users on the system, active services, MAC address, etc.
- **“nbtscan”** (also available for Linux…)
  - performs the operations of nbtstat on an entire network

NBTSCAN: AN EXAMPLE

```
root@kali:~# nbtscan -r 143.225.  XXX.XXX/XX
Doing NBT name scan for addresses from 143.225.  XXX.XXX/XX

IP address       NetBIOS Name     Server    User             MAC address
------------------------------------------------------------------------------
143.225.         Sendto failed: Permission denied
143.225.         XXXXXXXXXXXXX    <server>  <unknown>        00:22:64:
143.225.         XXXXXXXXXXXXX    <server>  <unknown>        18:03:73:
143.225.         <unknown>                  <unknown>
143.225.         GREEN-PC         <server>  <unknown>        08:60:6e:
143.225.         XXXXXXXXXXXXX    <server>  <unknown>        00:19:99:
143.225.         FMREPOS          <server>  FMREPOS          00:00:00:
143.225.         POSEMBEDDED      <server>  <unknown>        00:22:64:
143.225.         POSSECLABA       <server>  <unknown>        4c:72:b9:
143.225.         TIME-CAPSULE-DI  <server>  <unknown>        20:c9:d0:
143.225.         NASD985F8        <server>  NASD985F8        00:00:00:
143.225.         Sendto failed: Permission denied
143.225.         XXXXXXXXXXXXX    <server>  <unknown>        00:15:f2:
143.225.         <unknown>        <server>  <unknown>        00:00:00:
143.225.         AIRPORT-TIME-CA  <server>  <unknown>        90:72:40:
143.225.         DAVIDE-OFFICE    <server>  <unknown>        c8:60:00:
143.225.         WIN-R3RC0L0HRDA  <server>  <unknown>        b8:ca:3a:
143.225.         XXXXXXXXXXXXX              <unknown>        c8:2a:14:
143.225.         XXXXXXXXXXXXX    <server>  <unknown>        c8:9c:dc:
143.225.         XXXXXXXXXXXXX    <server>  <unknown>        00:1f:f3:
143.225.         NPI0D69BB                  <unknown>        00:1a:4b:
143.225.         XRX9C934E5E0AE0  <server>  <unknown>        9c:93:4e:
```

NETBIOS NAME SERVICES: COUNTERMEASURES

- Restricting (or denying) access to UDP port 137
- To prevent user data from appearing in NETBIOS tables:
  - disable the "Alerter" and "Messenger" services on individual hosts in the domain
    - Services Control Panel
- To prevent access to NETBIOS services from the Internet:
  - disable the NETBIOS over TCP/IP service in the properties of the individual network adapters on your host

NETBIOS SESSION ENUMERATION: TCP 139/445

- The main Achilles' heel for Windows systems:
  - null session (or anonymous connection) attack
- It is an exploit of the SMB (Server Message Block) protocol, which forms the foundation for Microsoft's file and print sharing services:

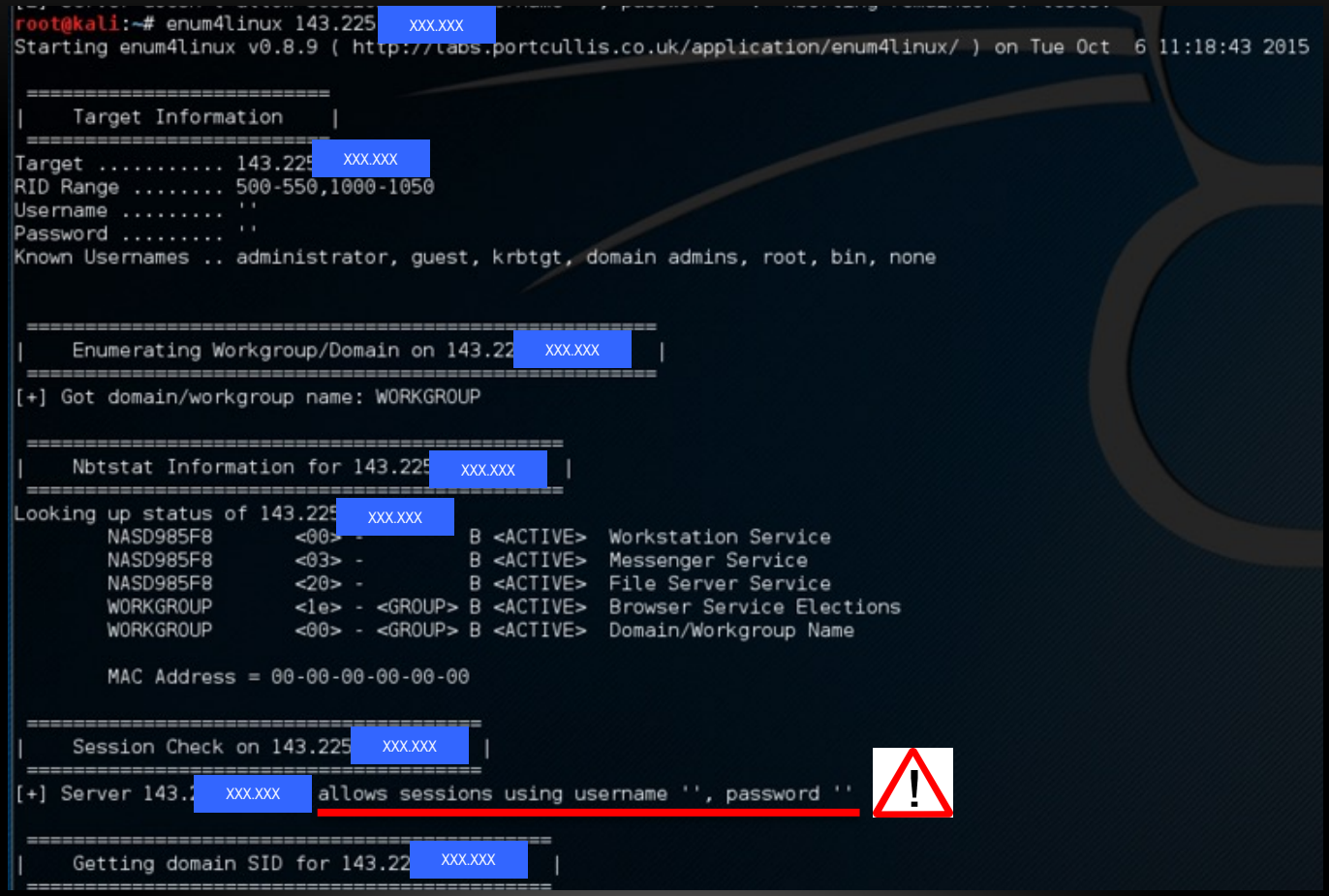
```
C:\>net use \\192.168.202.33\IPC$ "" /u:""
```

  - the syntax is similar to the command used to "mount" a network drive
  - it is used to:
    - connect to the hidden Inter Process Communication (IPC$) shared resource
    - as an anonymous user ("/u:"")
    - with a null password ("")
- If successful, the attacker gains an open channel to extract sensitive information from the target system, including:
  - network information, shared folders, users, groups, registry keys, etc.

NULL SESSION: “ALL-IN-ONE” TOOLS

- A set of pre-packaged tools for:
  - establishing a null session with the target
  - retrieving as much information as possible from the target by exploiting the established session
- Windows:
  - Winfingerprint: https://github.com/kkuehl/winfingerprint
  - NBTenum: http://nbtenum.sourceforge.net/
- Linux:
  - enum4linux: http://tools.kali.org/information-gathering/enum4linux

ENUM4LINUX IN ACTION…



SMB NULL SESSION: COUNTERMEASURES

- Port numbers TCP used: 139 and 445 (the latter from Windows 2000 onwards)
  - the most immediate solution is to filter TCP (and UDP) ports 139 and 445 on all perimeter access devices in your network
- On individual hosts, you can disable the SMB services
  - unbind the WINS client (TCP/IP) from the network interface using the "Bindings" tab in the networking control panel
  - for systems after Windows NT4 Service Pack3
    - configure the "RestrictAnonymous" flag in the system registry
      - a facility specifically designed to prevent the enumeration of sensitive information by exploiting "null sessions
    - however, please note that this solution can still be circumvented by some of the more powerful attack tools

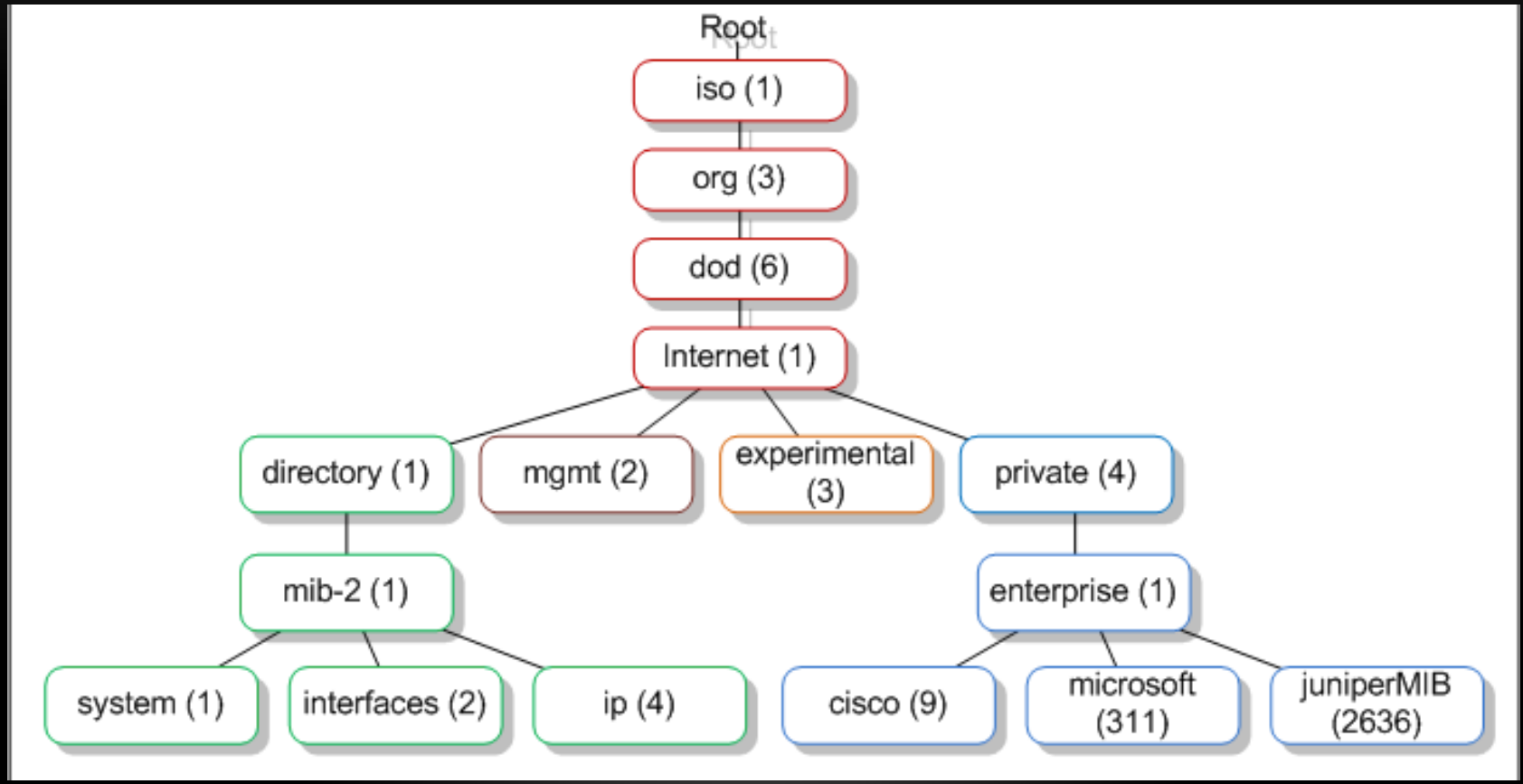
SNMP ENUMERATION: UDP, PORT 161

- Simple Network Management Protocol (SNMP)...
- ...aka "Security Not My Problem" (at least for versions 1 and 2 of the protocol)!
- A protocol designed to provide intimate information about network devices
- Equipped with a simple password-based authentication mechanism, often configured too loosely
  - Example: default password for read-only access to SNMP devices is often "public"
- Data is stored in a dedicated structure called the Management Information Base (MIB)
  - Many pieces of information are published in the proprietary part of the MIB by individual device manufacturers
    - Example: Windows NT systems provide user account names

SEARCHING FOR SNMP-ENABLED DEVICES

```
MacBookPro-spromano:logs spromano$ sudo nmap -sU -p161 --script snmp-brute --script-args snmplist=community.lst 192.168.1.0/24

Starting Nmap 6.40-2 ( http://nmap.org ) at 2015-10-07 08:52 CEST
Nmap scan report for 192.168.1.64
Host is up (0.77s latency).
PORT    STATE         SERVICE
161/udp open|filtered snmp
| snmp-brute:
|   admin - Valid credentials
|_  public - Valid credentials
MAC Address: B0:E8:92:76:34:13 (Seiko Epson)

Nmap scan report for 192.168.1.79
Host is up (0.84s latency).
PORT    STATE  SERVICE
161/udp closed snmp
MAC Address: 40:F3:08:8D:52:4A (Murata Manufactuaring Co.)

Nmap scan report for 192.168.1.253
Host is up (0.041s latency).
PORT    STATE  SERVICE
161/udp closed snmp
MAC Address: 9E:97:26:D0:5C:0E (Unknown)

Nmap scan report for 192.168.1.254
Host is up (0.0083s latency).
PORT    STATE         SERVICE
161/udp open|filtered snmp
MAC Address: 9C:97:26:D0:5C:0E (Technicolor)

Nmap scan report for 192.168.1.76
Host is up (0.000071s latency).
PORT    STATE  SERVICE
161/udp closed snmp

Nmap done: 256 IP addresses (5 hosts up) scanned in 58.27 seconds
```

SNMP MIB


Root
iso (1)
org (3)
dod (6)
Internet (1)
directory (1)
mgmt (2)
experimental (3)
private (4)
mib-2 (1)
enterprise (1)
system (1)
interfaces (2)
ip (4)
cisco (9)
microsoft (311)
juniperMIB (2636)

STROLLING THROUGH THE MIB

```
[root]# snmpwalk -c public -v 2c 192.168.1.60

system.sysDescr.0 = Linux wave 2.6.10 mdk #1 Sun Apr 15 2008 i686
system.sysObjectID.0 = OID: enterprises.ucdavis.ucdSnmpAgent.linux
system.sysUpTime.0 = Timeticks: (25701) 0:04:17.01
system.sysContact.0 = Root <root@localhost> (configure /etc/snmp/snmp.
conf)system.sysName.0 = wave
system.sysLocation.0 = Unknown (confi gure /etc/snmp/snmp.conf)system.
sysORLastChange.0 = Timeticks: (0)

[output truncated for brevity]
```

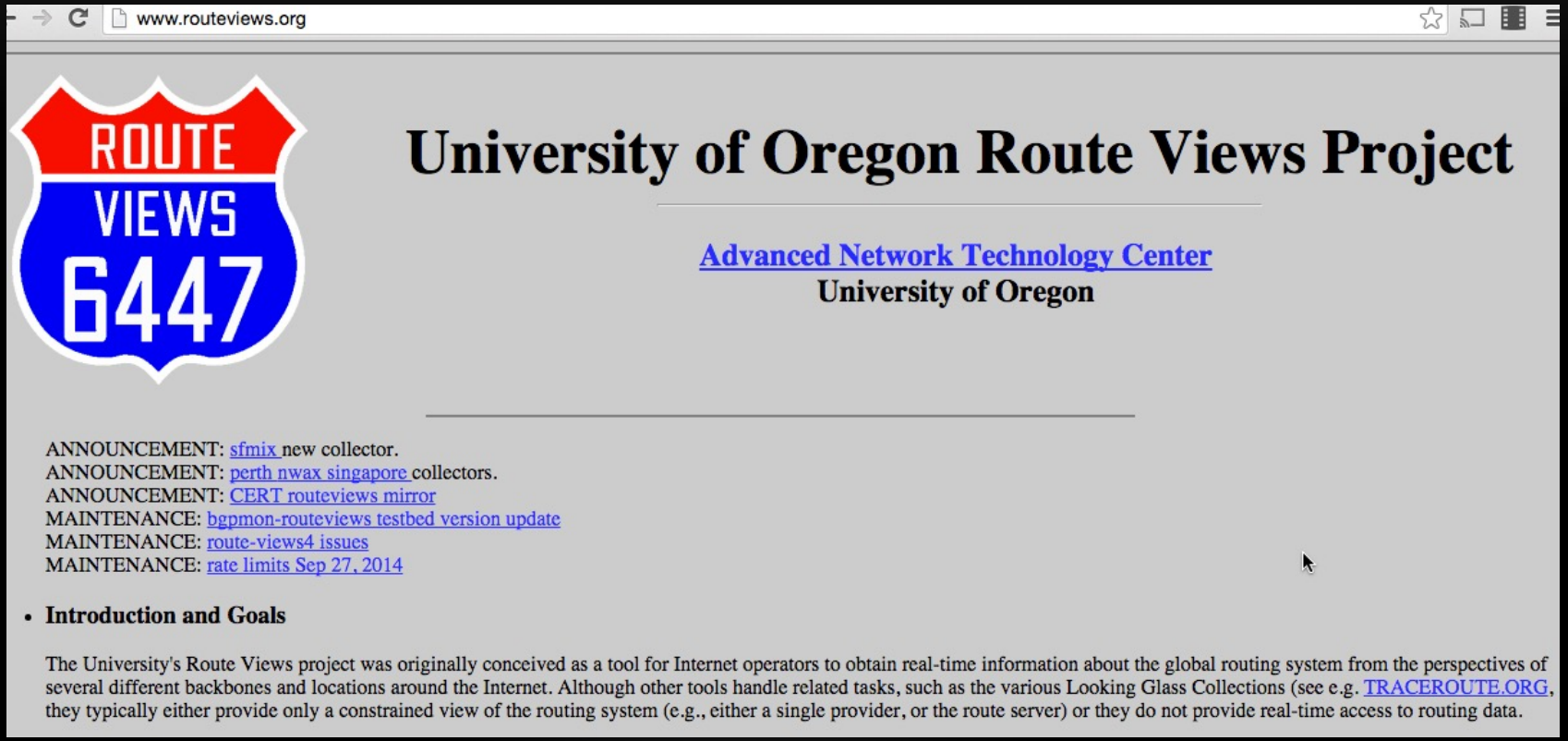
SNMP ENUMERATION: COUNTERMEASURES

- Disable SNMP agents on individual machines
- For active agents:
  - configure hard-to-guess "community" names
- On the network:
  - block access to port 161 across the entire perimeter of your infrastructure
- In general:
  - use the latest version of the protocol (SNMPv3) which offers much more advanced encryption and authentication mechanisms

BGP ENUMERATION: TCP, PORT 179

- Two steps:
  1. Determine the Autonomous System Number (ASN) of the target organization
  2. Execute a query on routers to identify all networks where the Autonomous System path (AS path) terminates with the ASN identified in the previous step
- Use publicly available services:
  - www.arin.net → query with “ASN” as a keyword
- An alternative technique, starting from a generic IP address of the target organization:
  1. query a 'public' BGP router
  2. find the ASN, identified by the "last hop" in the path vector associated with the IP in question
     - Route Views Project from the University of Oregon

ROUTE VIEWS PROJECT



USING ROUTE VIEWS

```
MacBook-Pro-di-Simon-2:DSP_Projects spromano$ telnet route-views.routeviews.org
Trying 128.223.51.103...
Connected to route-views.routeviews.org.
Escape character is '^]'.
C
**************************************************************************

                    RouteViews BGP Route Viewer
                    route-views.routeviews.org

 route views data is archived on http://archive.routeviews.org

 This hardware is part of a grant by the NSF.
 Please contact help@routeviews.org if you have questions, or
 if you wish to contribute your view.

 This router has views of full routing tables from several ASes.
 The list of peers is located at http://www.routeviews.org/peers
 in route-views.oregon-ix.net.txt

 NOTE: The hardware was upgraded in August 2014.  If you are seeing
 the error message, "no default Kerberos realm", you may want to
 in Mac OS X add "default unset autologin" to your ~/.telnetrc

 To login, use the username "rviews".

 **************************************************************************

User Access Verification

Username: rviews
```

```
route-views>show ip bgp 143.225.229.254
BGP routing table entry for 143.225.0.0/16, version 1032847344
Paths: (25 available, best #12, table default)
  Not advertised to any peer
  Refresh Epoch 1
  49788 1299 137 137 137
    91.218.184.60 from 91.218.184.60 (91.218.184.60)
      Origin IGP, localpref 100, valid, external
      Community: 1299:30000
      Extended Community: 0x43:100:1
      path 7FE0C7CD4490 RPKI State valid
      rx pathid: 0, tx pathid: 0
  Refresh Epoch 1
  3257 174 137 137 137
```

ARIN Whois/RDAP

137 Search

» Search www.arin.net instead Search Filter: **Automatic**

all requests subject to terms of use

"137"

ASN: AS137

Source Registry	RIPE NCC
Number	*not provided*
Name	ASGARR
Handle	AS137
Last Changed	Wed, 05 May 2021 08:30:07 GMT (Wed May 05 2021 local time)
Self	https://rdap.db.ripe.net/autnum/137
Copyright	http://www.ripe.net/data-tools/support/documentation/terms
Remark	Consortium GARR
Port 43 Whois	whois.ripe.net

UNIX RPC ENUMERATION: PORTS 111 AND 32771

- “portmapper” service, implemented thruogh the *rpcbind* daemon:
  - it orchestrates client requests and assigns ports to the services listening
  - similar to the "finger" service but related to the services offered by a node

THE RPCINFO TOOL

```
MacBookPro-spromano:logs spromano$ rpcinfo -p localhost
   program vers proto   port
    100000    2   udp    111  portmapper
    100000    3   udp    111  portmapper
    100000    4   udp    111  portmapper
    100000    2   tcp    111  portmapper
    100000    3   tcp    111  portmapper
    100000    4   tcp    111  portmapper
    100024    1   udp    752  status
    100024    1   tcp   1019  status
    100021    0   udp    621  nlockmgr
    100005    3   udp    853  mountd
    100005    1   tcp   1023  mountd
    100005    3   tcp   1023  mountd
    100011    1   udp    885  rquotad
    100011    2   udp    885  rquotad
    100011    1   tcp    997  rquotad
    100011    2   tcp    997  rquotad
```

IPSec/IKE ENUMERATION: UDP, PORT 500

- IPSec:
  - the layer three protocol with security features
- IKE:
  - Internet Key Exchange
    - the component of IPSec responsible for handling the key negotiation phase
    - it is crucial for discovering the presence of Virtual Private Networks (VPNs) within the target organization
- In this case, enumeration is not simply based on sending generic probe packets to port 500
  - the standard requires that improperly formatted packets be ignored by the IPSec service

THE "*IKE-SCAN*" TOOL

- It constructs packets compatible with the IPSec specification
- Once it identifies a VPN, it tries to extract useful information about its configuration:
  - type of authentication (pre-shared keys vs. certificates)
  - encryption protocols used
  - operation mode (main mode vs. aggressive mode)
- This information can be a prelude to the attack phase, which may involve tools like "psk-crack"

IKE-SCAN IN ACTION

```
# ./ike-scan 10.10.10.0/24
Starting ike-scan 1.9 with 256 hosts \
(http://www.nta-monitor.com/tools/ike-scan/)
10.10.10.1  Main Mode Handshake returned HDR=(CKY-R= 42c304f96fa8f857)
\
SA=(Enc=3DES Hash=SHA1 Auth=PSK Group=2:modp1024 \
LifeType=Seconds LifeDuration(4)=0x00007080) VID= f4ed19e0cc114eb-
516faaac0ee37daf2807b4381f00000001
0000138d4925b9df0000000018000000
(Firewall-1 NGX)

Ending ike-scan 1.9: 1 hosts scanned in 0.087 seconds \
(11.47 hosts/sec).  1 returned handshake; 0 returned notify
```

IPSec/IKE ENUMERATION: COUNTERMEASURES

- Implementing source IP address filtering policies is a usable solution in sufficiently static scenarios, such as site-to-site VPNs with commercial partners
- It is advisable to use the "Main Mode" operation mode over "Aggressive Mode" for several reasons:
  - it provides a higher level of security as it doesn't expose sensitive information like pre-shared keys and device data during negotiation
  - it exchanges data with peers more securely
  - it is less susceptible to Denial of Service (DoS) attacks
- It is important to note that Aggressive Mode becomes the only choice when:
  - you must rely on pre-shared key (PSK) authentication
  - you need to establish dynamic connections with clients whose IP addresses are not known beforehand

QUESTIONS?