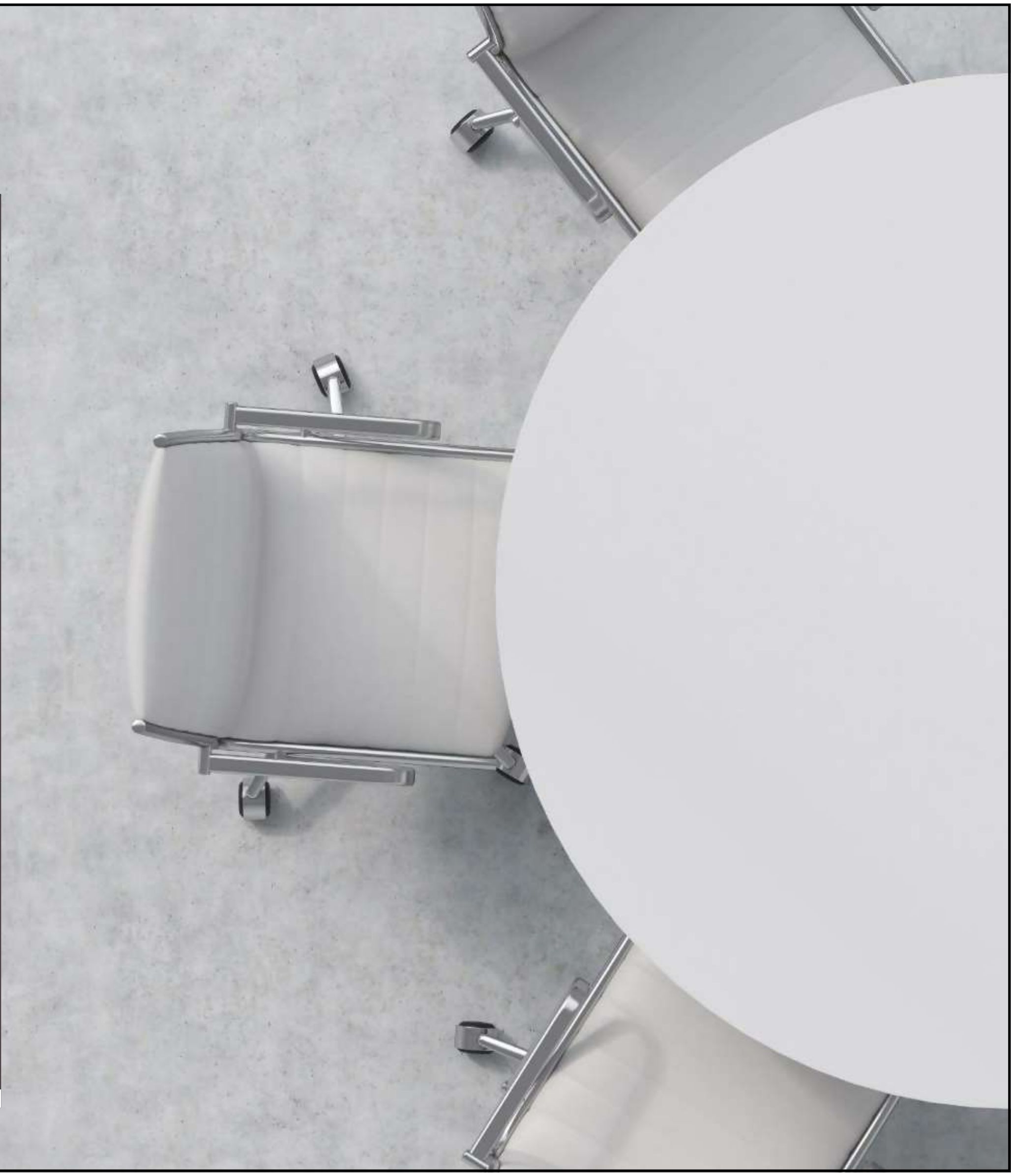
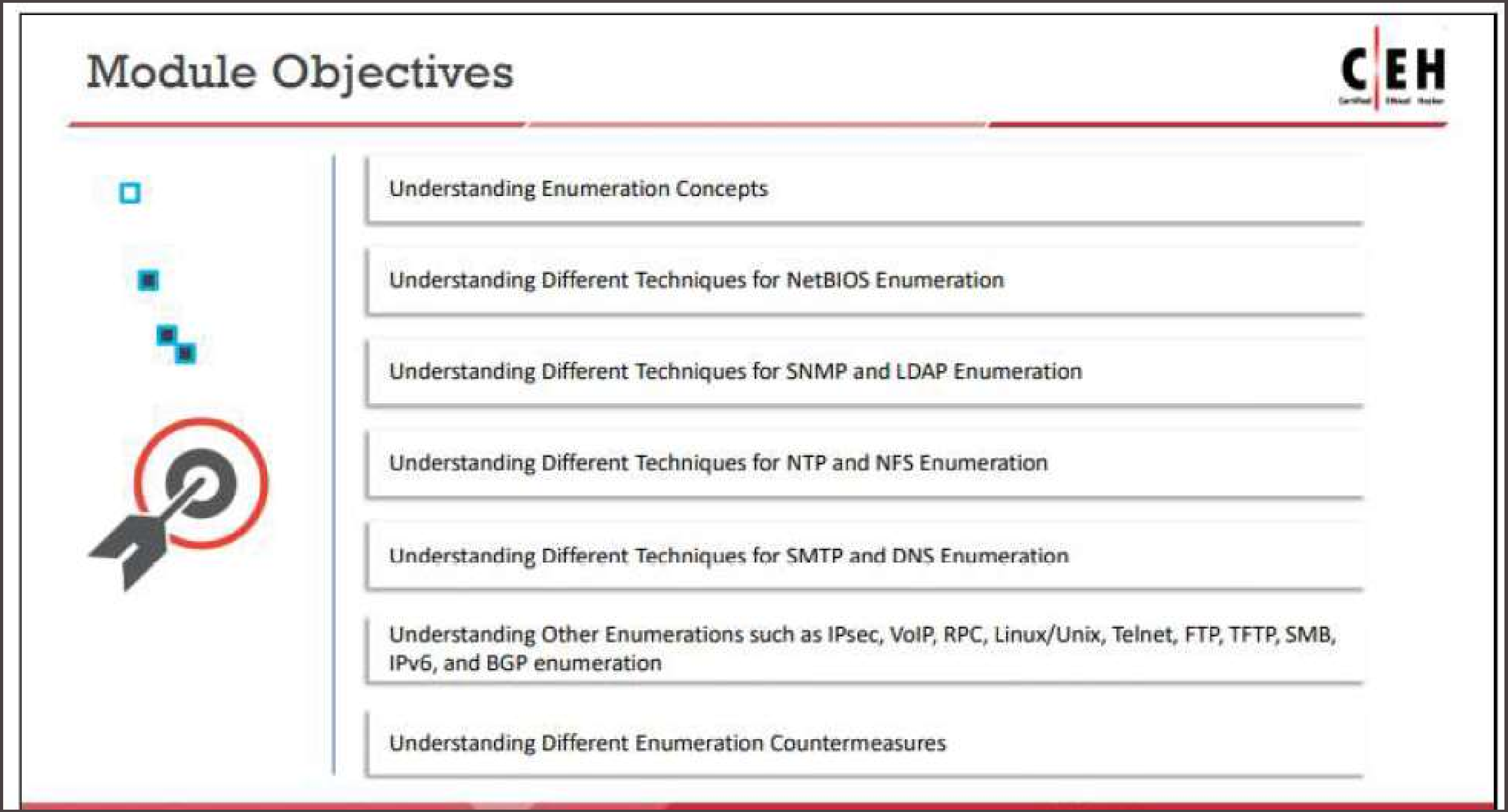
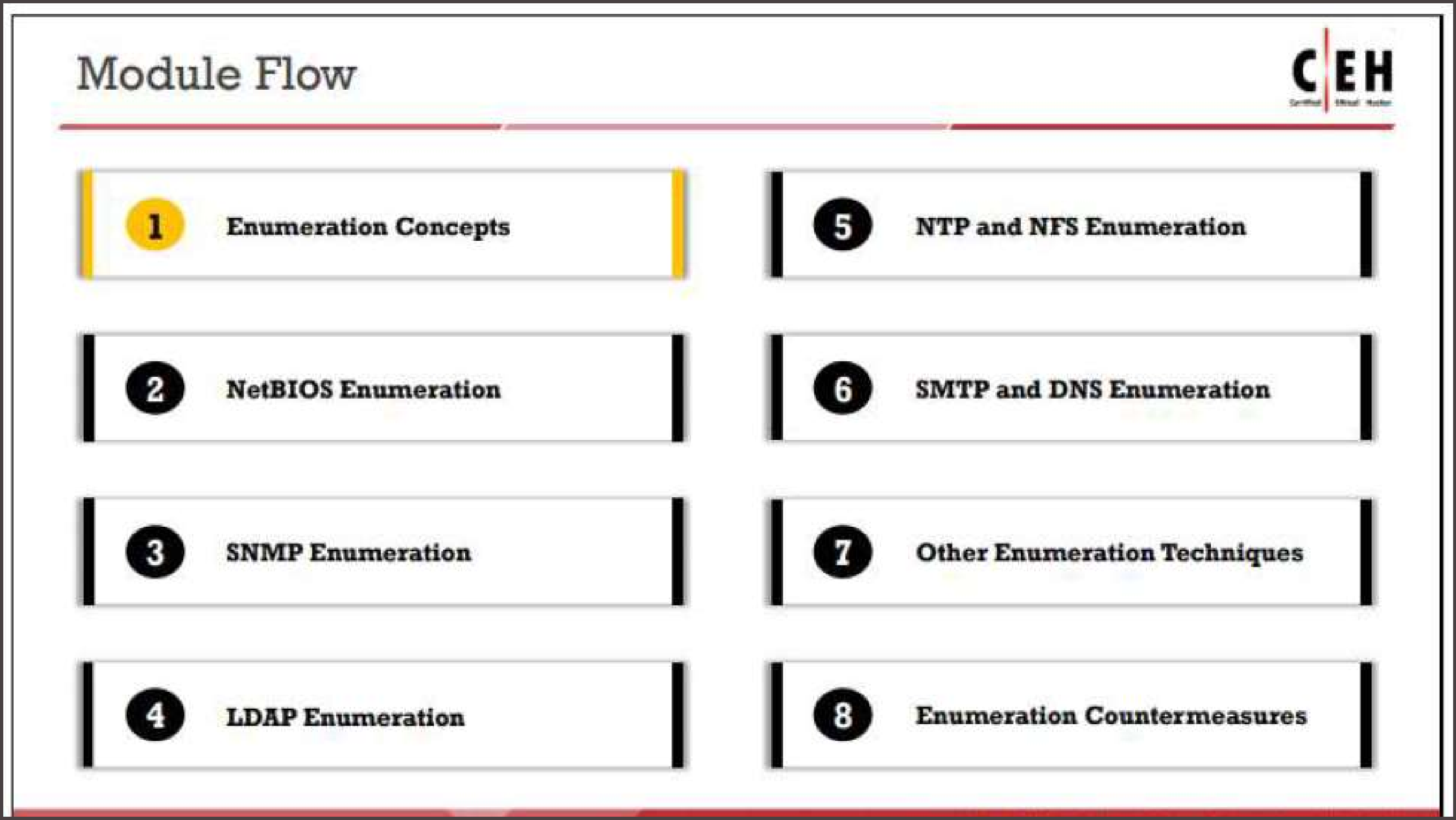
ENUMERATION





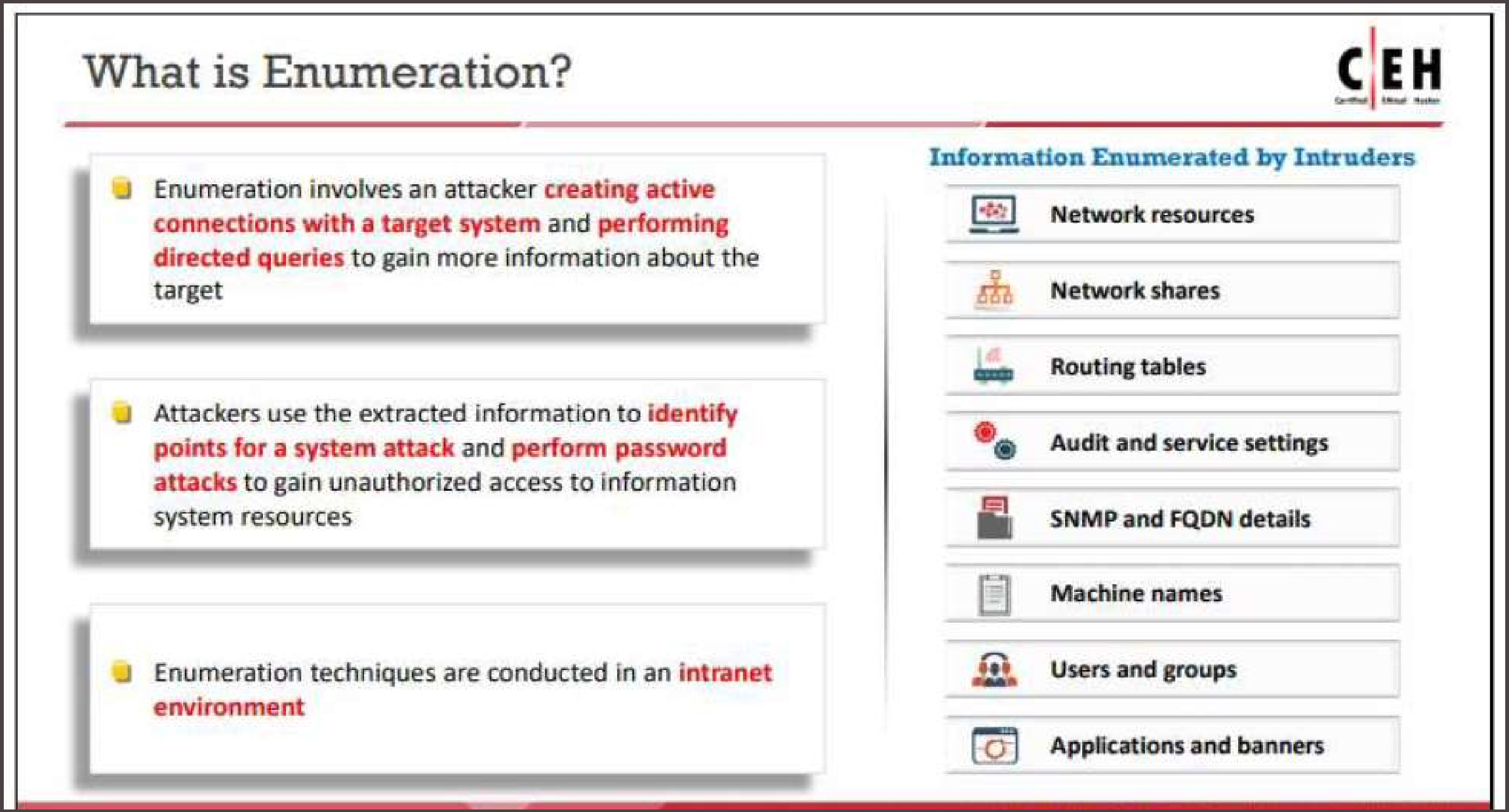
Module Objectives

- Understanding Enumeration Concepts
- Understanding Different Techniques for NetBIOS Enumeration
- Understanding Different Techniques for SNMP and LDAP Enumeration
- Understanding Different Techniques for NTP and NFS Enumeration
- Understanding Different Techniques for SMTP and DNS Enumeration
- Understanding Other Enumerations such as IPsec, VoIP, RPC, Linux/Unix, Telnet, FTP, TFTP, SMB, IPv6, and BGP enumeration
- Understanding Different Enumeration Countermeasures



Module Flow

1. Enumeration Concepts
2. NetBIOS Enumeration
3. SNMP Enumeration
4. LDAP Enumeration
5. NTP and NFS Enumeration
6. SMTP and DNS Enumeration
7. Other Enumeration Techniques
8. Enumeration Countermeasures



What is Enumeration?

C|EH

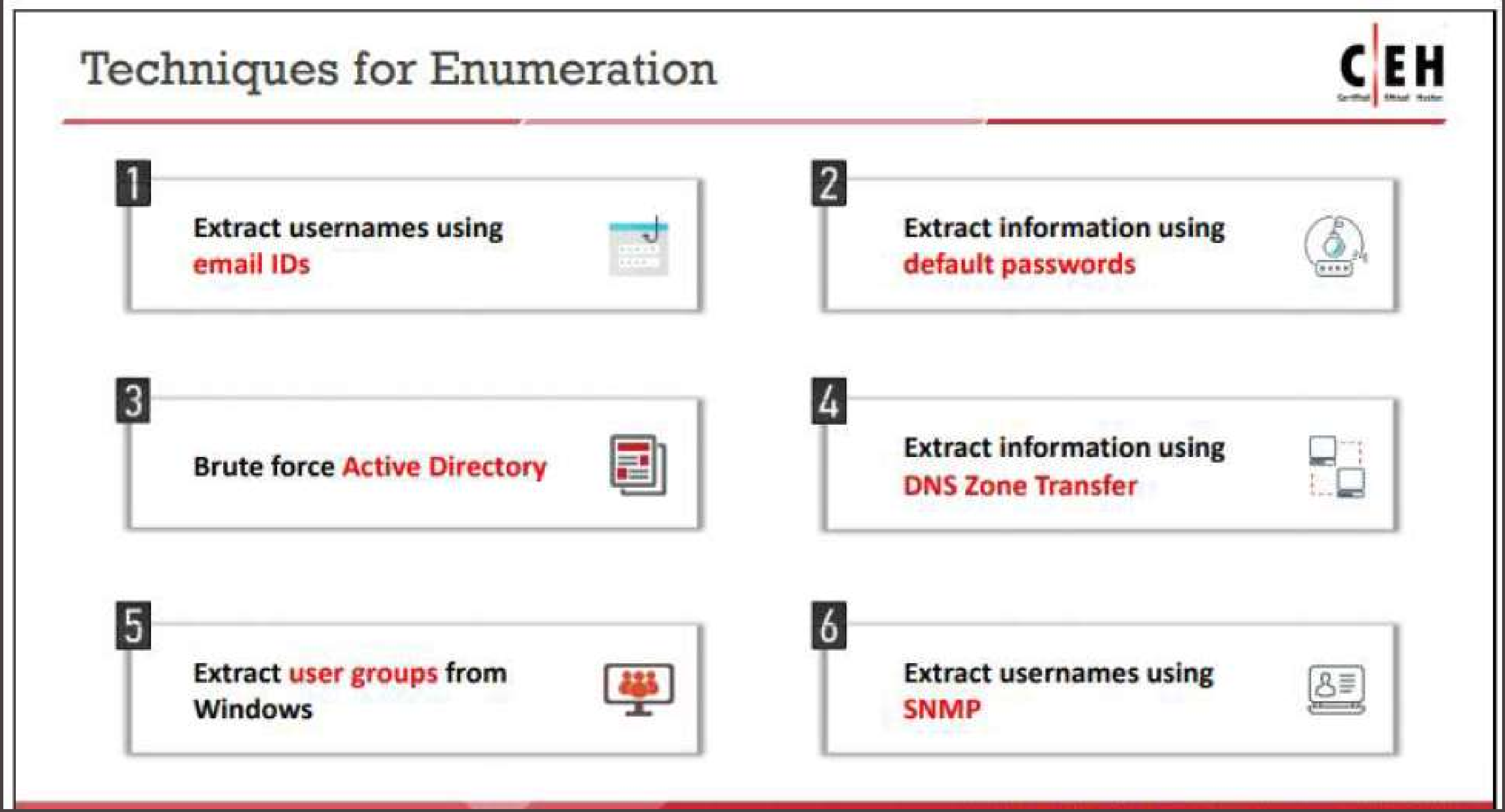
- Enumeration involves an attacker **creating active connections with a target system** and **performing directed queries** to gain more information about the target

- Attackers use the extracted information to **identify points for a system attack** and **perform password attacks** to gain unauthorized access to information system resources

- Enumeration techniques are conducted in an **intranet environment**

Information Enumerated by Intruders

- Network resources
- Network shares
- Routing tables
- Audit and service settings
- SNMP and FQDN details
- Machine names
- Users and groups
- Applications and banners



Techniques for Enumeration

1. Extract usernames using email IDs

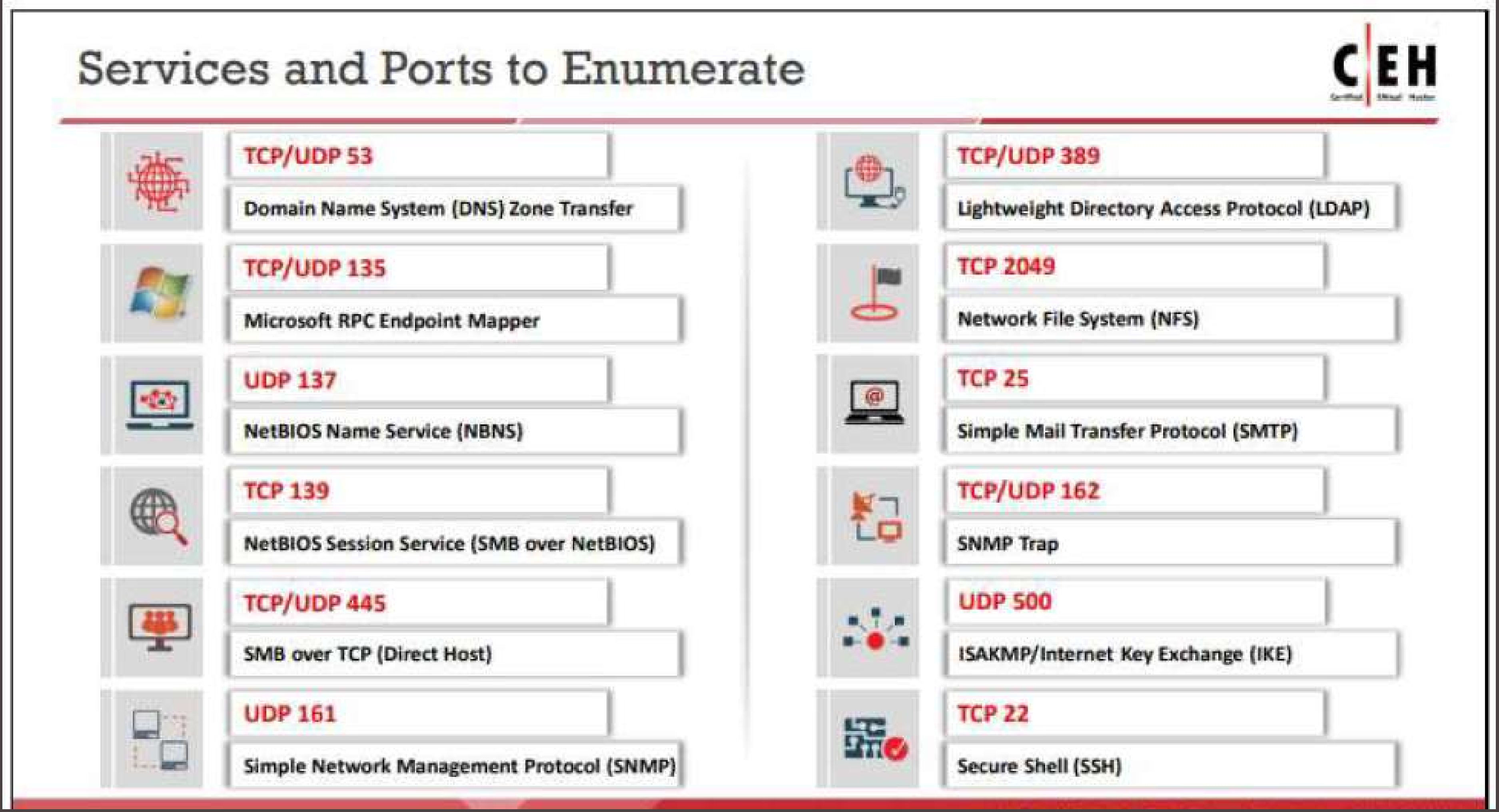
2. Extract information using default passwords

3. Brute force Active Directory

4. Extract information using DNS Zone Transfer

5. Extract user groups from Windows

6. Extract usernames using SNMP



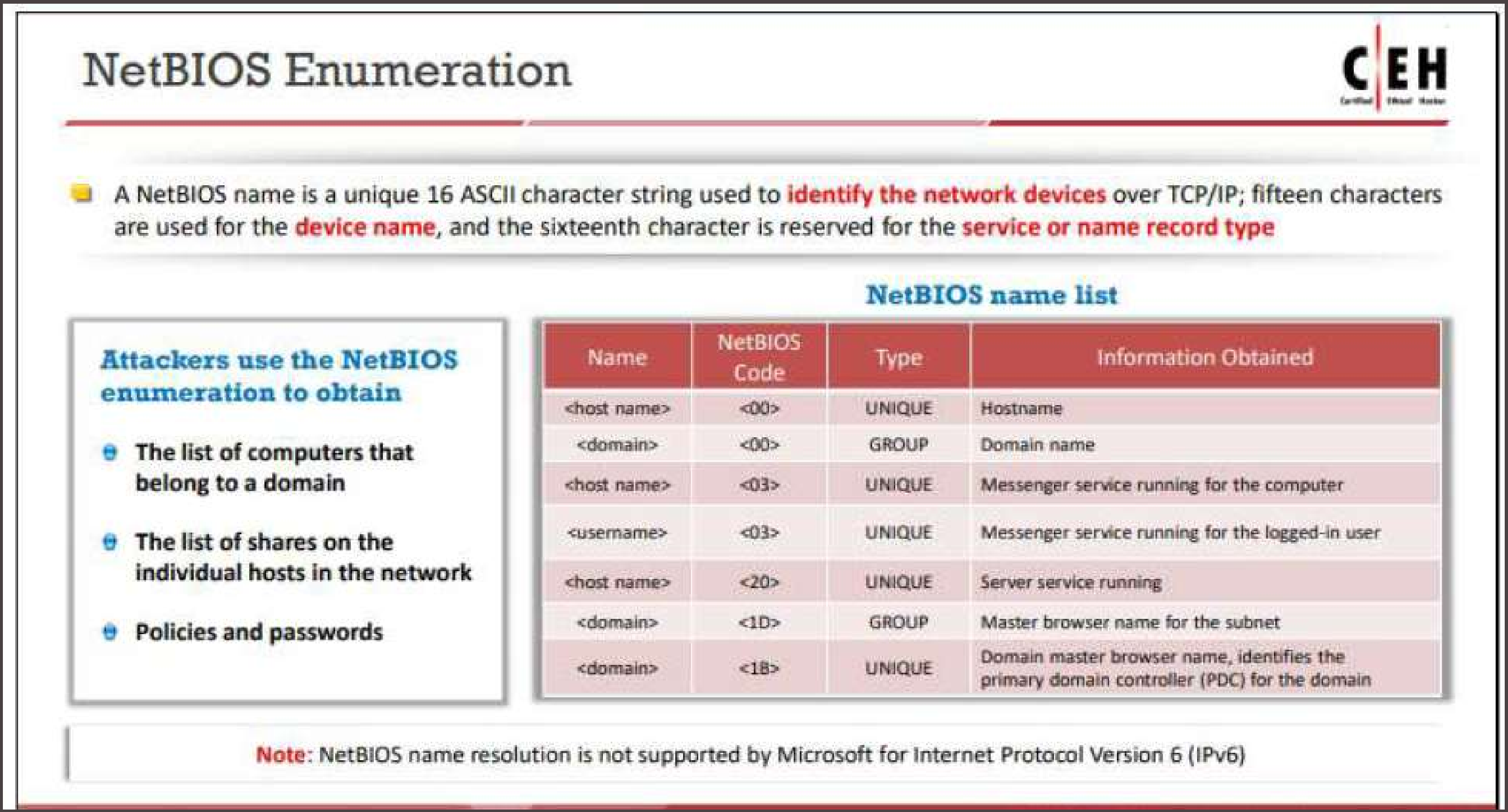
Services and Ports to Enumerate

TCP/UDP 53	Domain Name System (DNS) Zone Transfer
TCP/UDP 135	Microsoft RPC Endpoint Mapper
UDP 137	NetBIOS Name Service (NBNS)
TCP 139	NetBIOS Session Service (SMB over NetBIOS)
TCP/UDP 445	SMB over TCP (Direct Host)
UDP 161	Simple Network Management Protocol (SNMP)
TCP/UDP 389	Lightweight Directory Access Protocol (LDAP)
TCP 2049	Network File System (NFS)
TCP 25	Simple Mail Transfer Protocol (SMTP)
TCP/UDP 162	SNMP Trap
UDP 500	ISAKMP/Internet Key Exchange (IKE)
TCP 22	Secure Shell (SSH)

Module Flow

C|EH

1. Enumeration Concepts
2. NetBIOS Enumeration
3. SNMP Enumeration
4. LDAP Enumeration
5. NTP and NFS Enumeration
6. SMTP and DNS Enumeration
7. Other Enumeration Techniques
8. Enumeration Countermeasures



NetBIOS Enumeration

- A NetBIOS name is a unique 16 ASCII character string used to **identify the network devices** over TCP/IP; fifteen characters are used for the **device name**, and the sixteenth character is reserved for the **service or name record type**

Attackers use the NetBIOS enumeration to obtain

- The list of computers that belong to a domain
- The list of shares on the individual hosts in the network
- Policies and passwords

NetBIOS name list

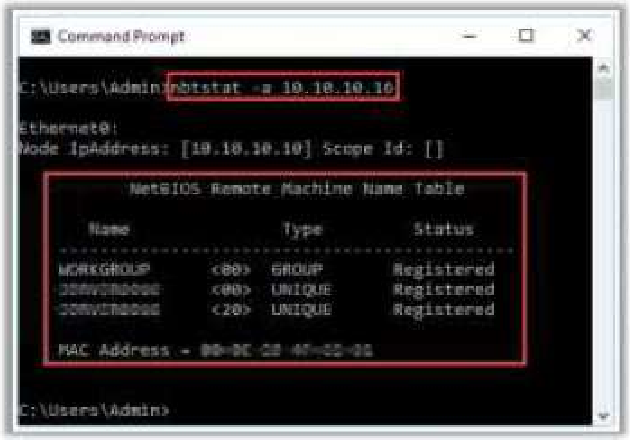
Name	NetBIOS Code	Type	Information Obtained
<host name>	<00>	UNIQUE	Hostname
<domain>	<00>	GROUP	Domain name
<host name>	<03>	UNIQUE	Messenger service running for the computer
<username>	<03>	UNIQUE	Messenger service running for the logged-in user
<host name>	<20>	UNIQUE	Server service running
<domain>	<1D>	GROUP	Master browser name for the subnet
<domain>	<1B>	UNIQUE	Domain master browser name, identifies the primary domain controller (PDC) for the domain

Note: NetBIOS name resolution is not supported by Microsoft for Internet Protocol Version 6 (IPv6)

NetBIOS Enumeration (Cont'd)

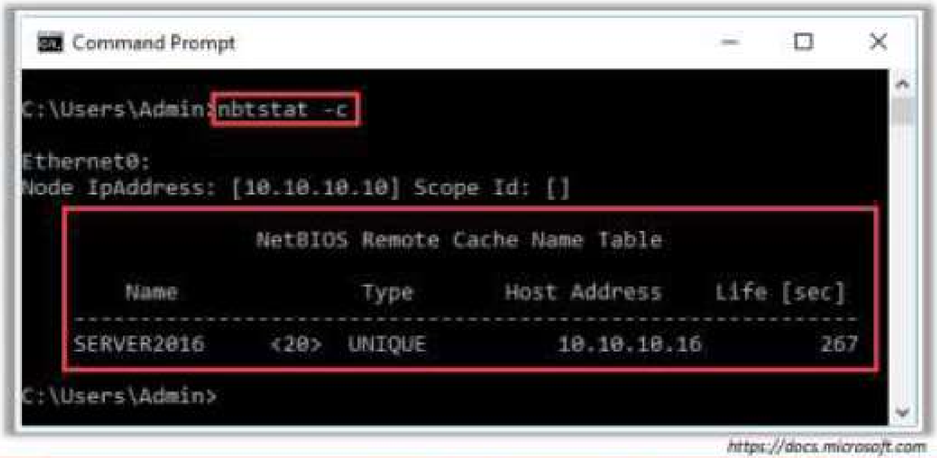
- The nbtstat utility in Windows displays NetBIOS over **TCP/IP** (NetBT) **protocol statistics**, **NetBIOS name tables** for both the local and remote computers, and the **NetBIOS name cache**

- Run the `nbtstat` command "`nbtstat -a <IP address of the remote machine>`" to obtain the NetBIOS name table of a remote computer



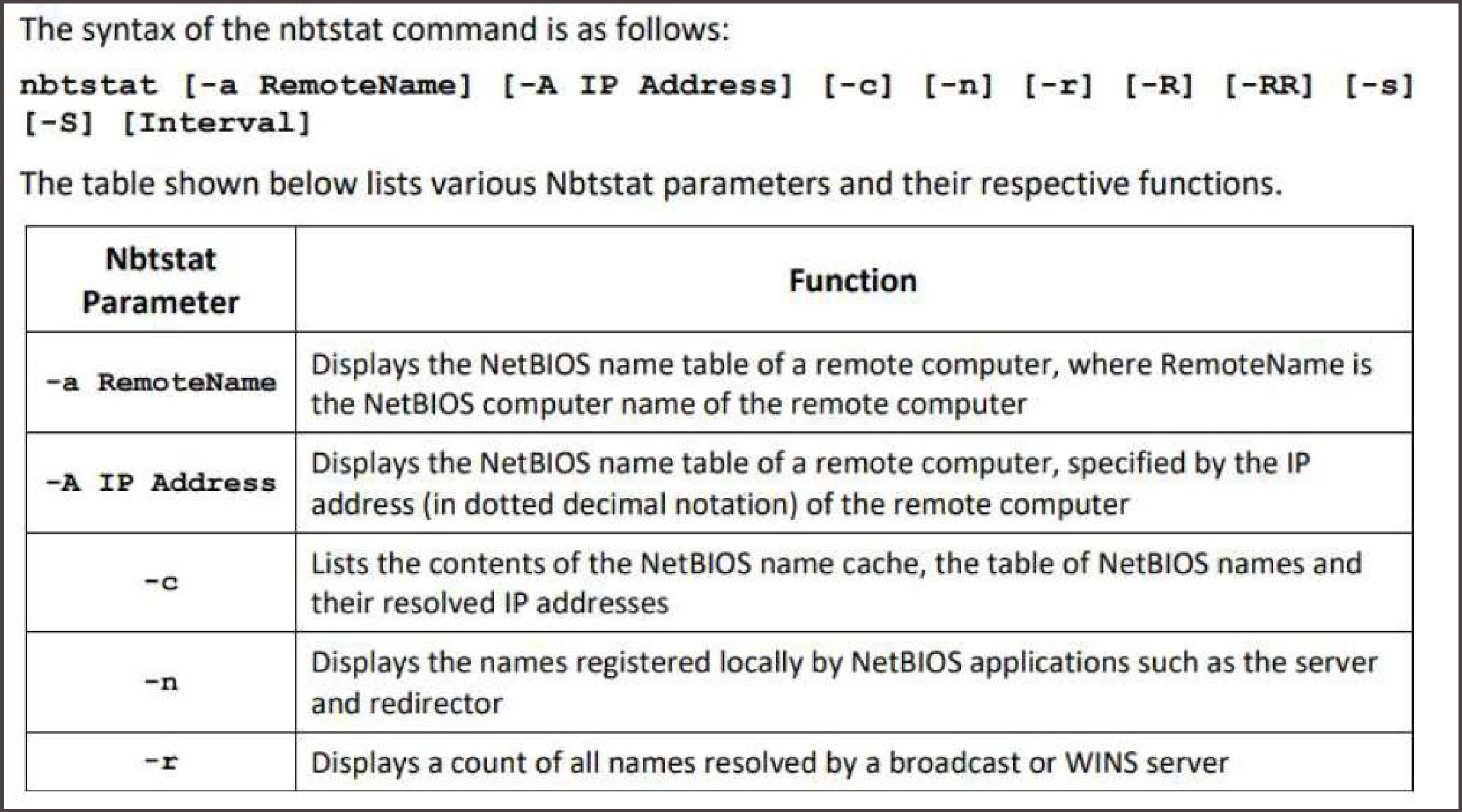
```
C:\Users\Admin>nbtstat -a 10.10.10.16

Ethernet0:
Node IpAddress: [10.10.10.10] Scope Id: []

           NetBIOS Remote Machine Name Table

       Name               Type         Status
    ---------------------------------------------
    WORKGROUP      <00>  GROUP       Registered
    [illegible]    <00>  UNIQUE      Registered
    [illegible]    <20>  UNIQUE      Registered

    MAC Address = 00-0C-29-[illegible]

C:\Users\Admin>
```

- Run the `nbtstat` command "`nbtstat -c`" to obtain the contents of the NetBIOS name cache, table of NetBIOS names, and their resolved IP addresses



```
C:\Users\Admin>nbtstat -c

Ethernet0:
Node IpAddress: [10.10.10.10] Scope Id: []

                  NetBIOS Remote Cache Name Table

    Name              Type       Host Address     Life [sec]
    ------------------------------------------------------------
    SERVER2016     <20>  UNIQUE          10.10.10.16          267

C:\Users\Admin>
```

https://docs.microsoft.com

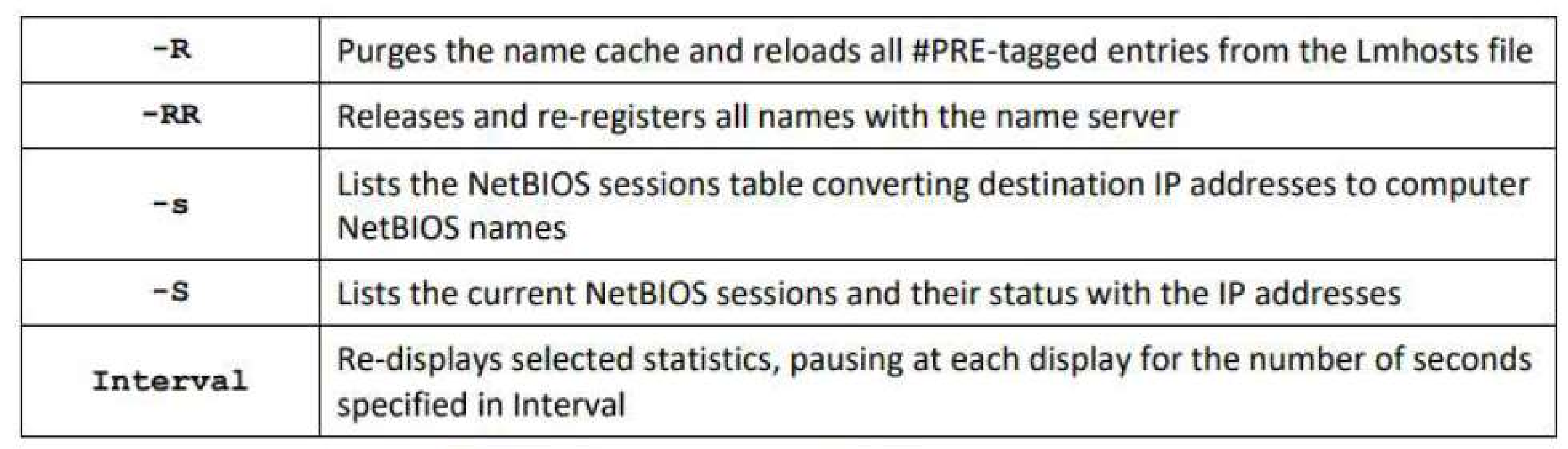


The syntax of the nbtstat command is as follows:

```
nbtstat [-a RemoteName] [-A IP Address] [-c] [-n] [-r] [-R] [-RR] [-s]
[-S] [Interval]
```

The table shown below lists various Nbtstat parameters and their respective functions.

Nbtstat Parameter	Function
`-a RemoteName`	Displays the NetBIOS name table of a remote computer, where RemoteName is the NetBIOS computer name of the remote computer
`-A IP Address`	Displays the NetBIOS name table of a remote computer, specified by the IP address (in dotted decimal notation) of the remote computer
`-c`	Lists the contents of the NetBIOS name cache, the table of NetBIOS names and their resolved IP addresses
`-n`	Displays the names registered locally by NetBIOS applications such as the server and redirector
`-r`	Displays a count of all names resolved by a broadcast or WINS server



-R	Purges the name cache and reloads all #PRE-tagged entries from the Lmhosts file
-RR	Releases and re-registers all names with the name server
-s	Lists the NetBIOS sessions table converting destination IP addresses to computer NetBIOS names
-S	Lists the current NetBIOS sessions and their status with the IP addresses
Interval	Re-displays selected statistics, pausing at each display for the number of seconds specified in Interval

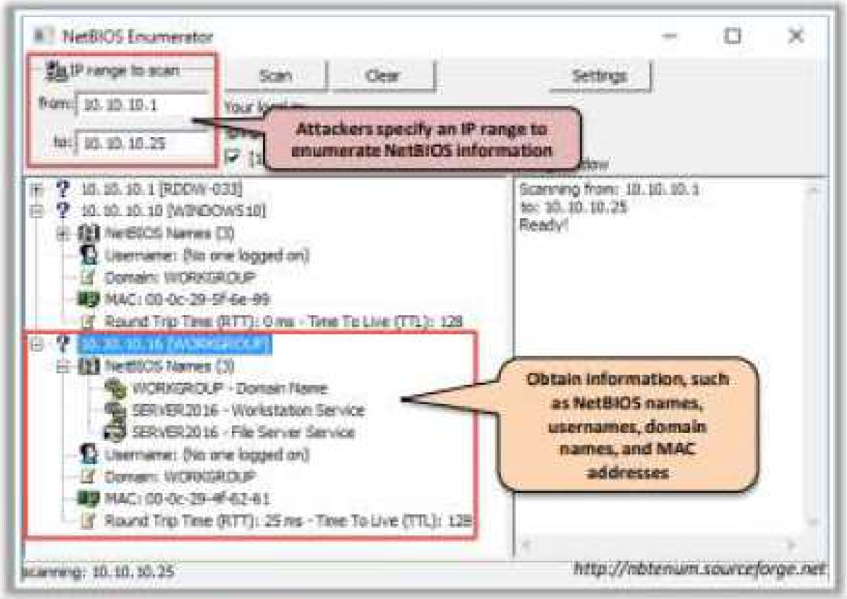
NetBIOS Enumeration Tools

nmap -sV -v --script nbstat.nse ip address

C|EH

NetBIOS Enumerator

NetBIOS Enumerator helps to enumerate details, such as **NetBIOS names**, **Usernames**, **Domain names**, and **MAC addresses**, for a given range of IP addresses



from: 10.10.10.1
to: 10.10.10.25

Attackers specify an IP range to enumerate NetBIOS information

```
⊞ ? 10.10.10.1 [RDDW-033]
⊟ ? 10.10.10.10 [WINDOWS10]
    ⊞ NetBIOS Names (3)
      Username: (No one logged on)
      Domain: WORKGROUP
      MAC: 00-0c-29-5f-6e-99
      Round Trip Time (RTT): 0 ms - Time To Live (TTL): 128
⊟ ? 10.10.10.16 [WORKGROUP]
    ⊟ NetBIOS Names (3)
        WORKGROUP - Domain Name
        SERVER2016 - Workstation Service
        SERVER2016 - File Server Service
      Username: (No one logged on)
      Domain: WORKGROUP
      MAC: 00-0c-29-4f-62-61
      Round Trip Time (RTT): 25 ms - Time To Live (TTL): 128
```

```
Scanning from: 10.10.10.1
to: 10.10.10.25
Ready!
```

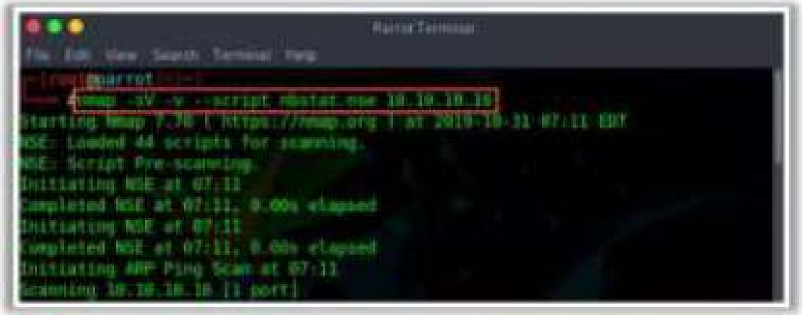
scanning: 10.10.10.25

Obtain information, such as NetBIOS names, usernames, domain names, and MAC addresses

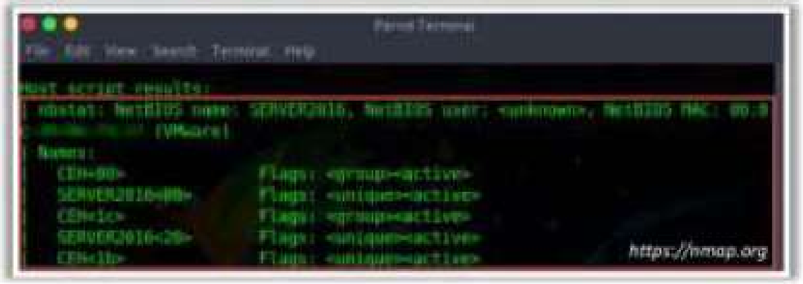
http://nbtenum.sourceforge.net

Nmap

Nmap's nbstat NSE script allow attackers to retrieve targets' **NetBIOS names** and **MAC addresses**



```
┌─[root@parrot]─[~]
└──╼ #nmap -sV -v --script nbstat.nse 10.10.10.16
Starting Nmap 7.70 ( https://nmap.org ) at 2019-10-31 07:11 EDT
NSE: Loaded 44 scripts for scanning.
NSE: Script Pre-scanning.
Initiating NSE at 07:11
Completed NSE at 07:11, 0.00s elapsed
Initiating NSE at 07:11
Completed NSE at 07:11, 0.00s elapsed
Initiating ARP Ping Scan at 07:11
Scanning 10.10.10.16 [1 port]
```

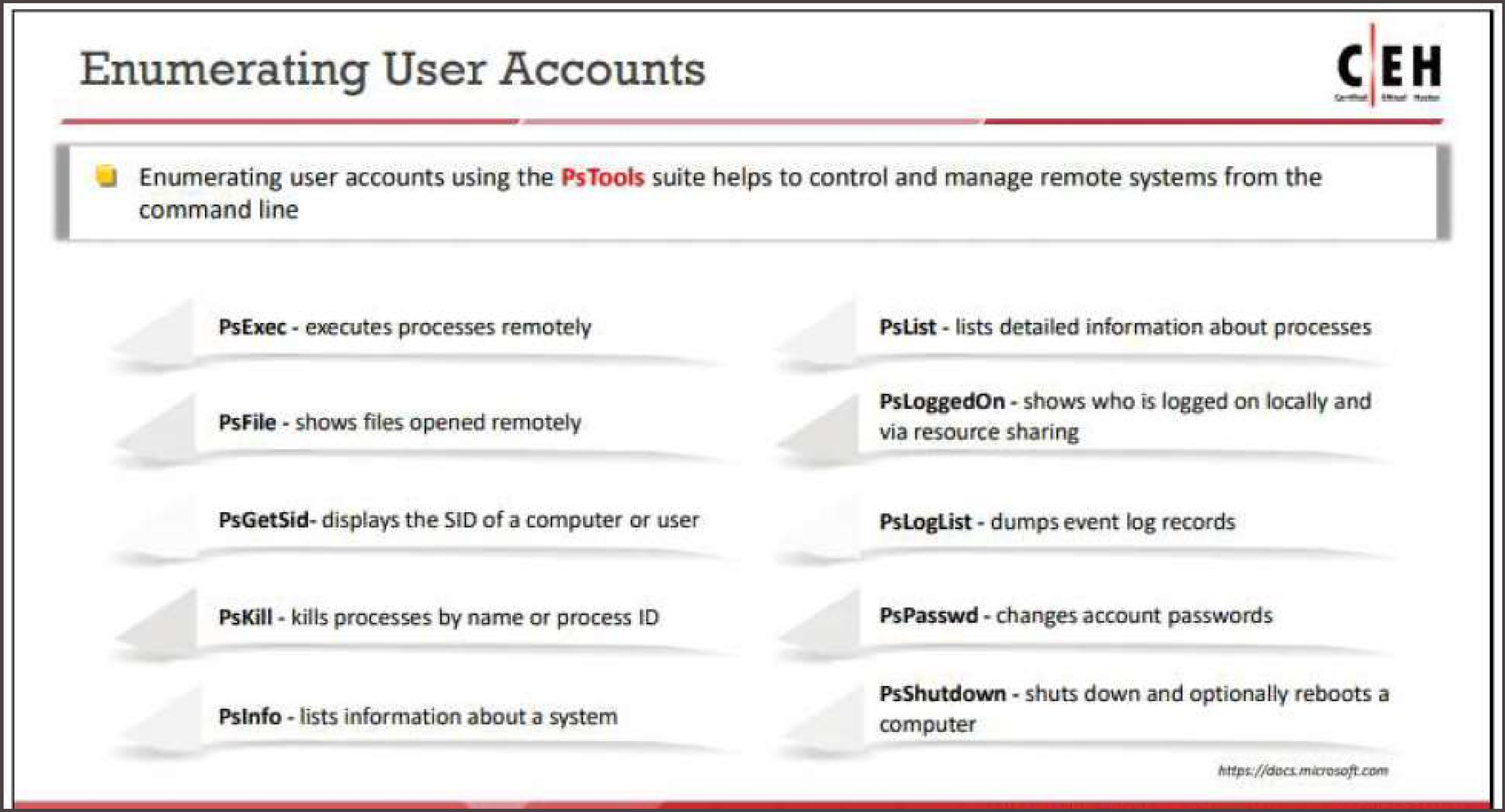


```
Host script results:
| nbstat: NetBIOS name: SERVER2016, NetBIOS user: <unknown>, NetBIOS MAC: 00:0
| [redacted] (VMware)
| Names:
|   CEH<00>            Flags: <group><active>
|   SERVER2016<00>     Flags: <unique><active>
|   CEH<1c>            Flags: <group><active>
|   SERVER2016<20>     Flags: <unique><active>
|   CEH<1b>            Flags: <unique><active>
```

https://nmap.org

Other NetBIOS Enumeration Tools:

- **Global Network Inventory** *http://www.magnetosoft.com*
- **Advanced IP Scanner** *http://www.advanced-ip-scanner.com*
- **Hyena** *https://www.systemtools.com*
- **Nsauditor Network Security Auditor** *https://www.nsauditor.com*



Enumerating User Accounts

- Enumerating user accounts using the **PsTools** suite helps to control and manage remote systems from the command line

PsExec - executes processes remotely

PsFile - shows files opened remotely

PsGetSid- displays the SID of a computer or user

PsKill - kills processes by name or process ID

PsInfo - lists information about a system

PsList - lists detailed information about processes

PsLoggedOn - shows who is logged on locally and via resource sharing

PsLogList - dumps event log records

PsPasswd - changes account passwords

PsShutdown - shuts down and optionally reboots a computer

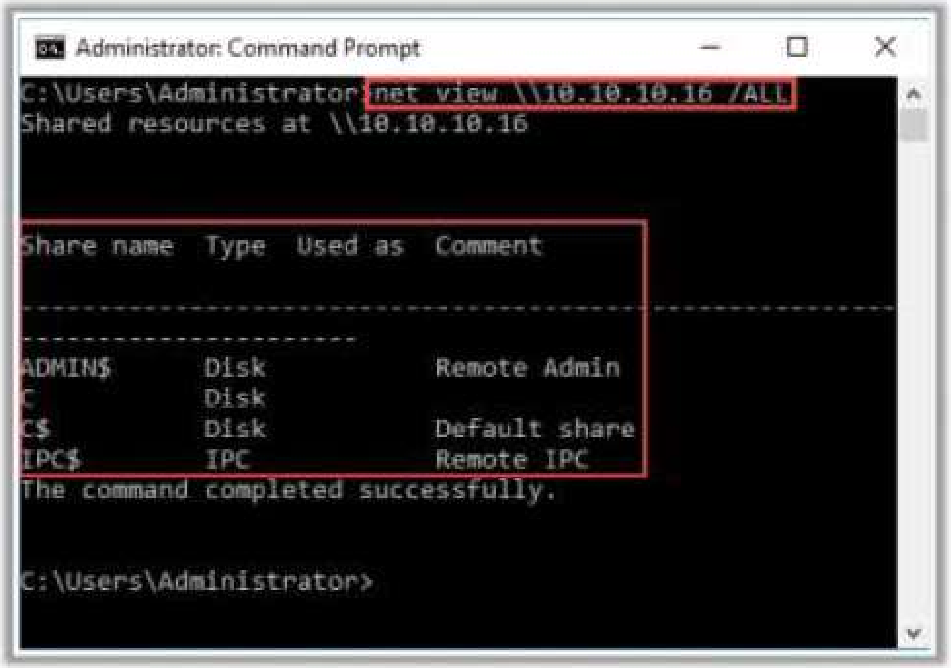
https://docs.microsoft.com

Enumerating Shared Resources Using Net View

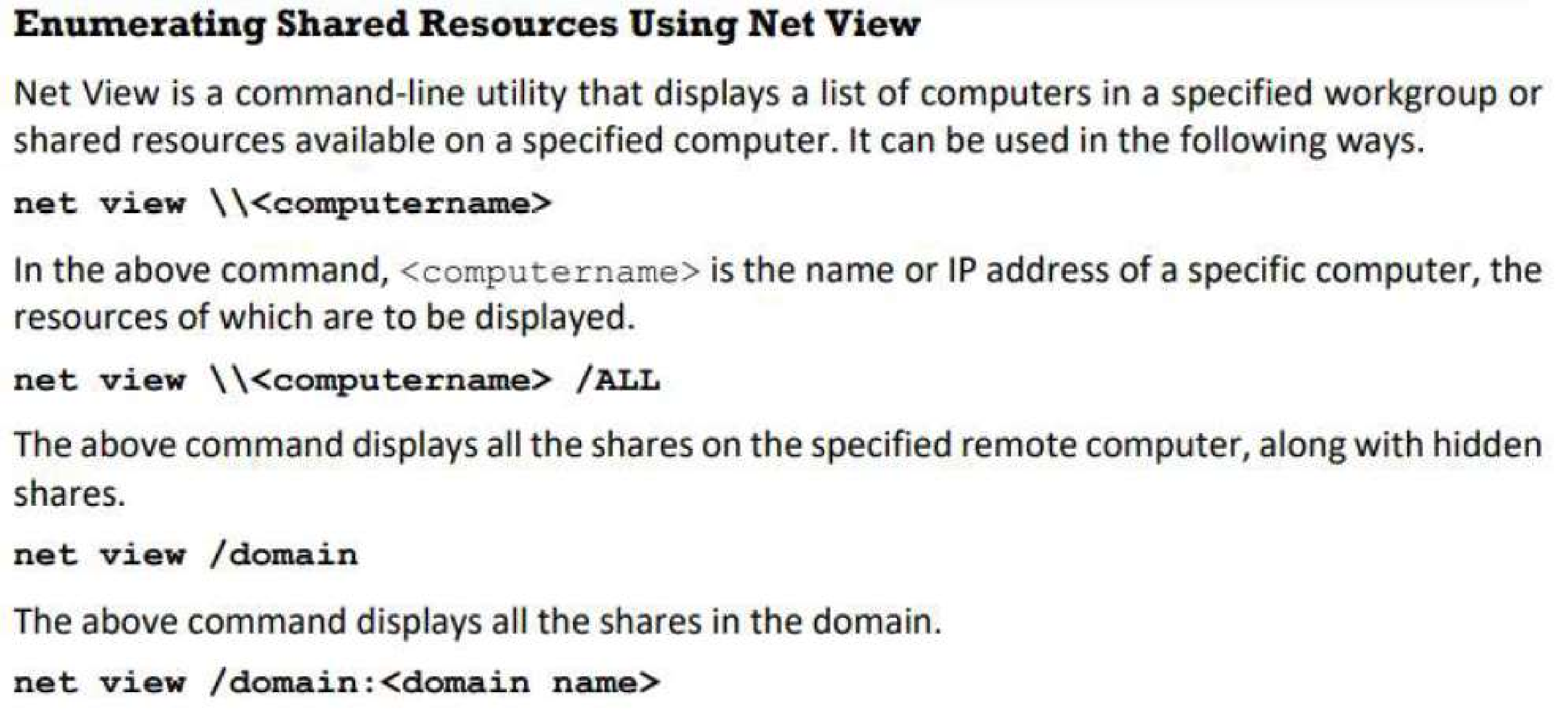
- The Net View utility is used to obtain a list of all the **shared resources of a remote host** or **workgroup**

Net View Commands

- **net view \\<computername>**
- **net view /domain:<domain name>**



```
C:\Users\Administrator>net view \\10.10.10.16 /ALL
Shared resources at \\10.10.10.16



Share name   Type   Used as   Comment

-------------------------------------------------------------------------------
----------------------
ADMIN$       Disk             Remote Admin
C            Disk
C$           Disk             Default share
IPC$         IPC              Remote IPC
The command completed successfully.


C:\Users\Administrator>
```



Enumerating Shared Resources Using Net View

Net View is a command-line utility that displays a list of computers in a specified workgroup or shared resources available on a specified computer. It can be used in the following ways.

```
net view \\<computername>
```

In the above command, `<computername>` is the name or IP address of a specific computer, the resources of which are to be displayed.

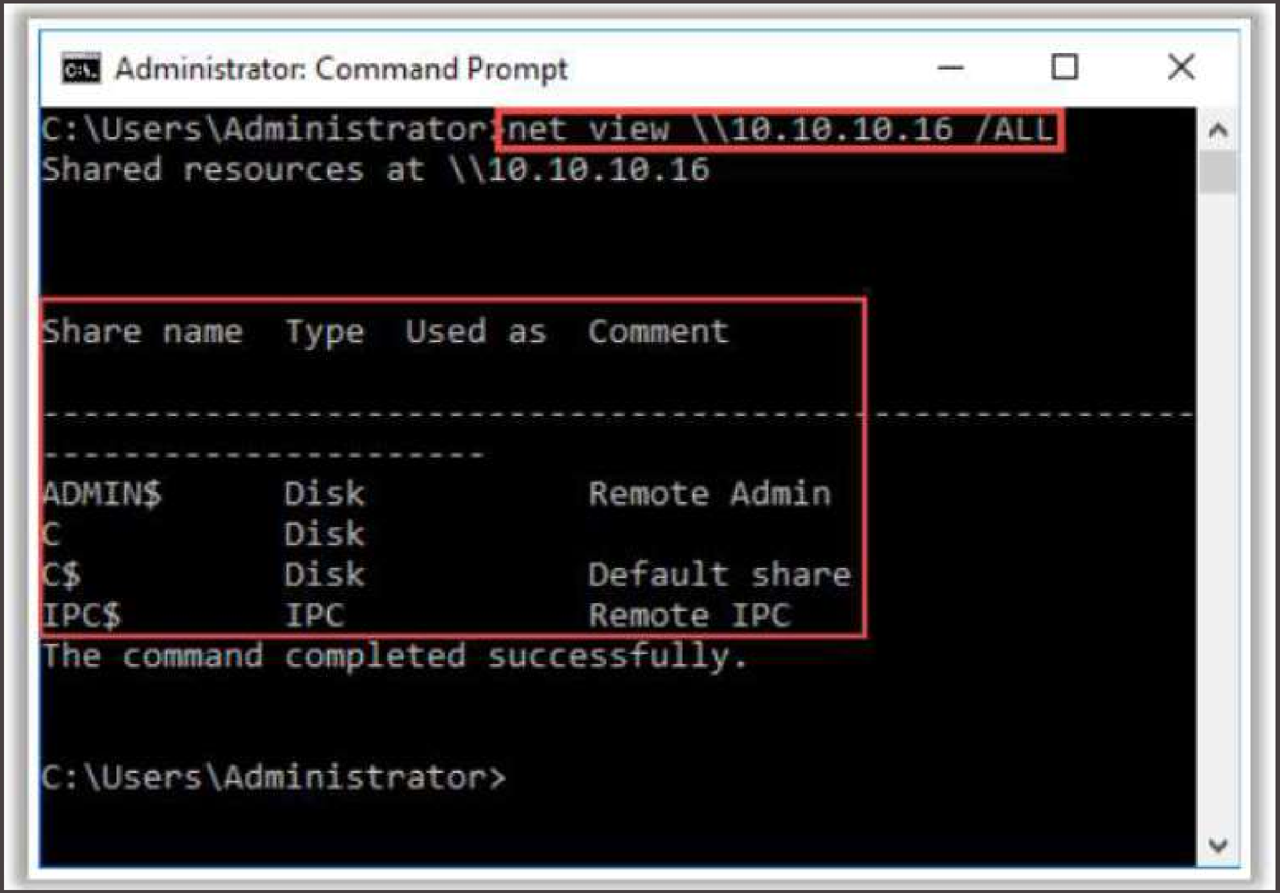
```
net view \\<computername> /ALL
```

The above command displays all the shares on the specified remote computer, along with hidden shares.

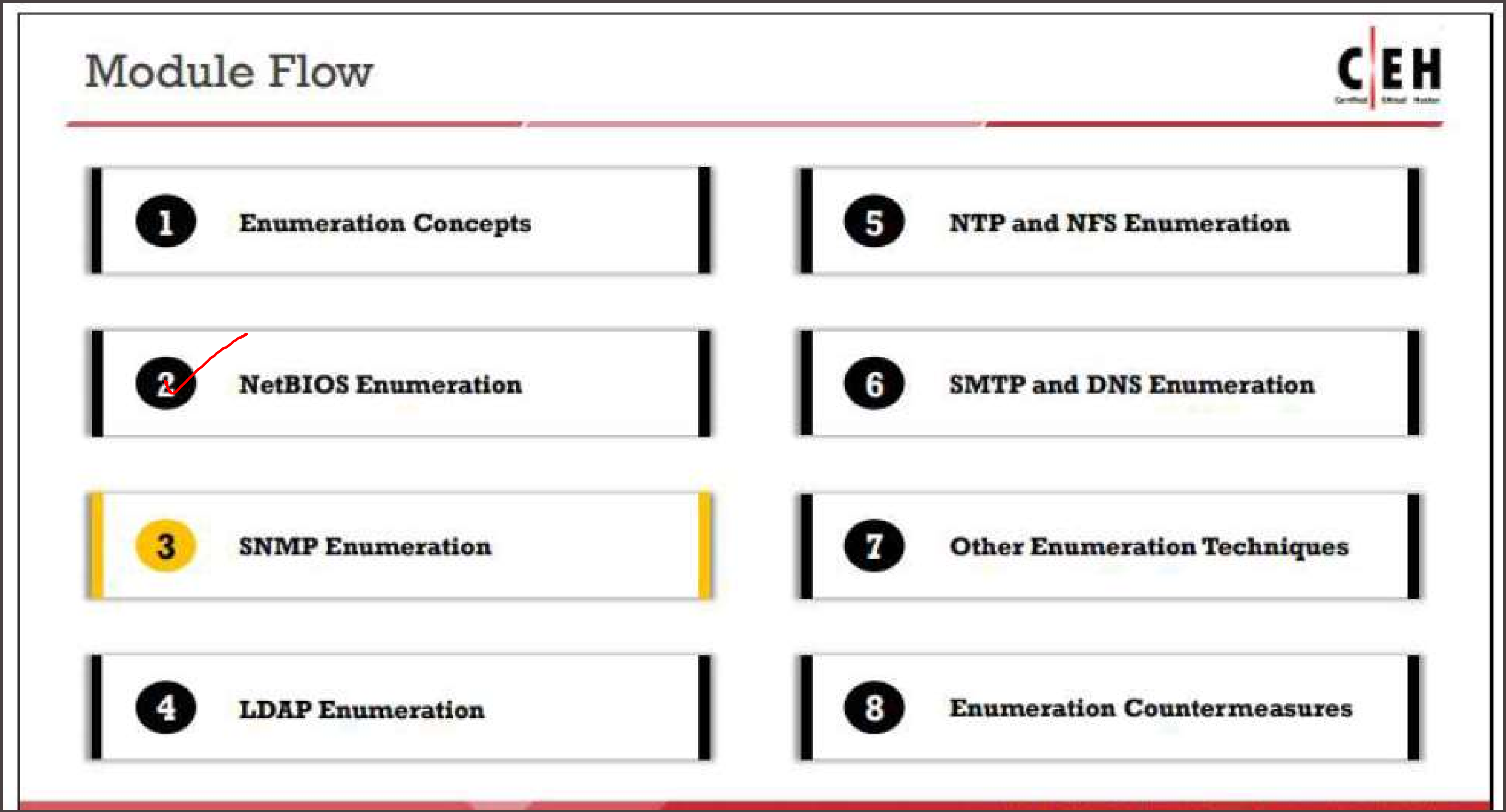
```
net view /domain
```

The above command displays all the shares in the domain.

```
net view /domain:<domain name>
```



```
C:\Users\Administrator>net view \\10.10.10.16 /ALL
Shared resources at \\10.10.10.16



Share name  Type  Used as  Comment


-------------------------------------------------------------------------
---------------------
ADMIN$      Disk           Remote Admin
C           Disk
C$          Disk           Default share
IPC$        IPC            Remote IPC
The command completed successfully.


C:\Users\Administrator>
```



Module Flow

1 Enumeration Concepts

2 NetBIOS Enumeration

3 SNMP Enumeration

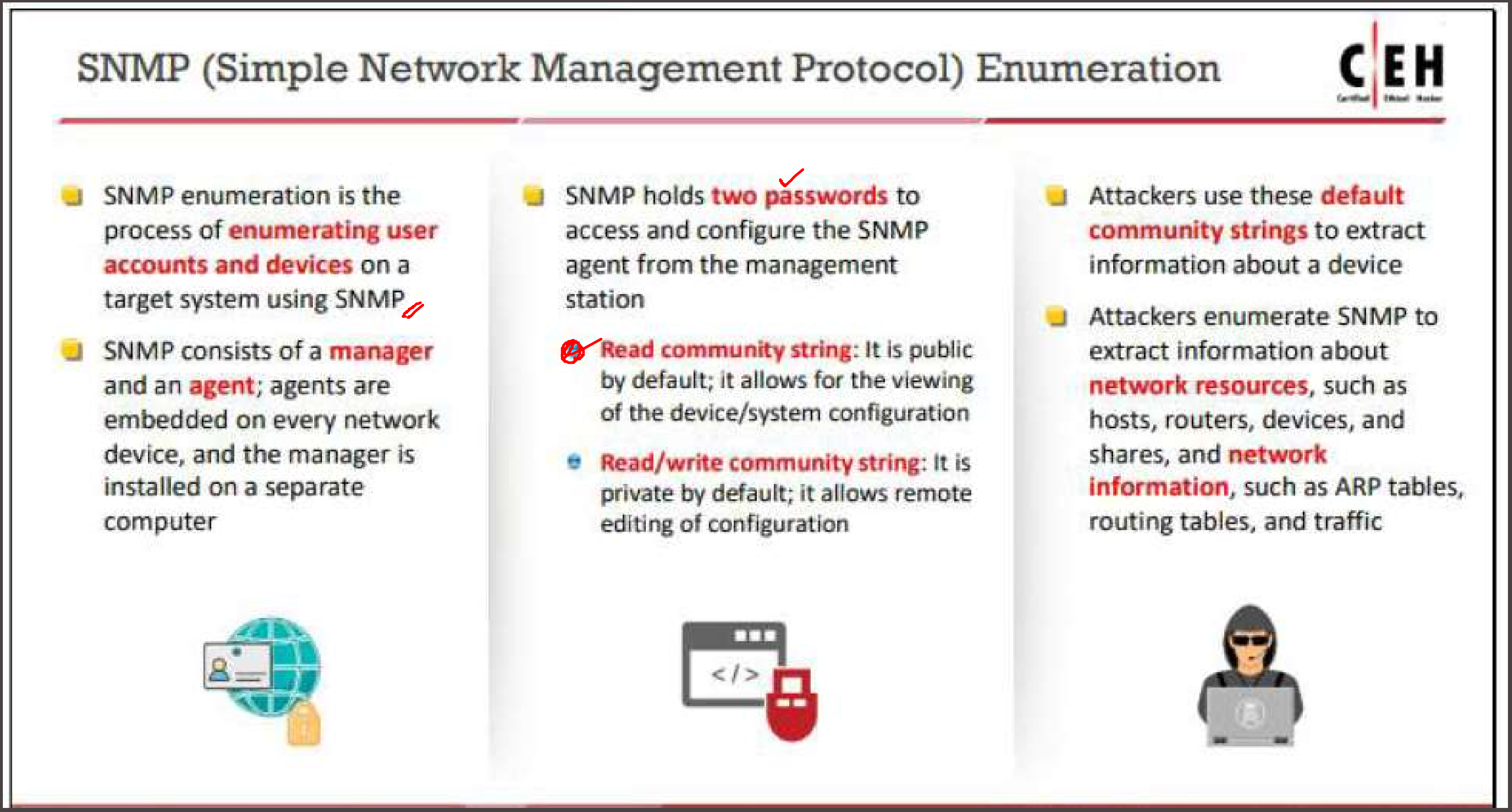
4 LDAP Enumeration

5 NTP and NFS Enumeration

6 SMTP and DNS Enumeration

7 Other Enumeration Techniques

8 Enumeration Countermeasures

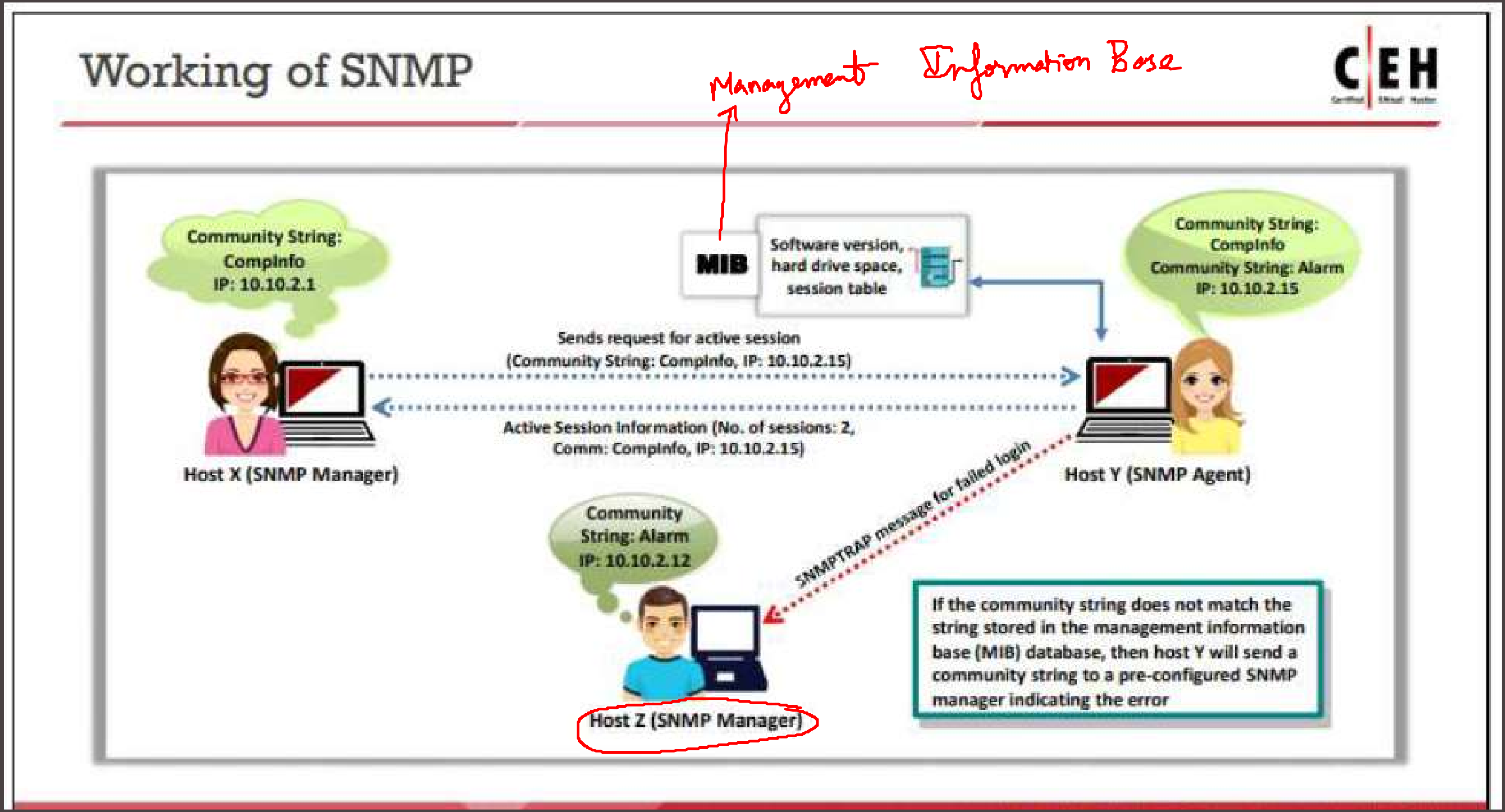


SNMP (Simple Network Management Protocol) Enumeration

- SNMP enumeration is the process of **enumerating user accounts and devices** on a target system using SNMP
- SNMP consists of a **manager** and an **agent**; agents are embedded on every network device, and the manager is installed on a separate computer

- SNMP holds **two passwords** to access and configure the SNMP agent from the management station
  - **Read community string**: It is public by default; it allows for the viewing of the device/system configuration
  - **Read/write community string**: It is private by default; it allows remote editing of configuration

- Attackers use these **default community strings** to extract information about a device
- Attackers enumerate SNMP to extract information about **network resources**, such as hosts, routers, devices, and shares, and **network information**, such as ARP tables, routing tables, and traffic

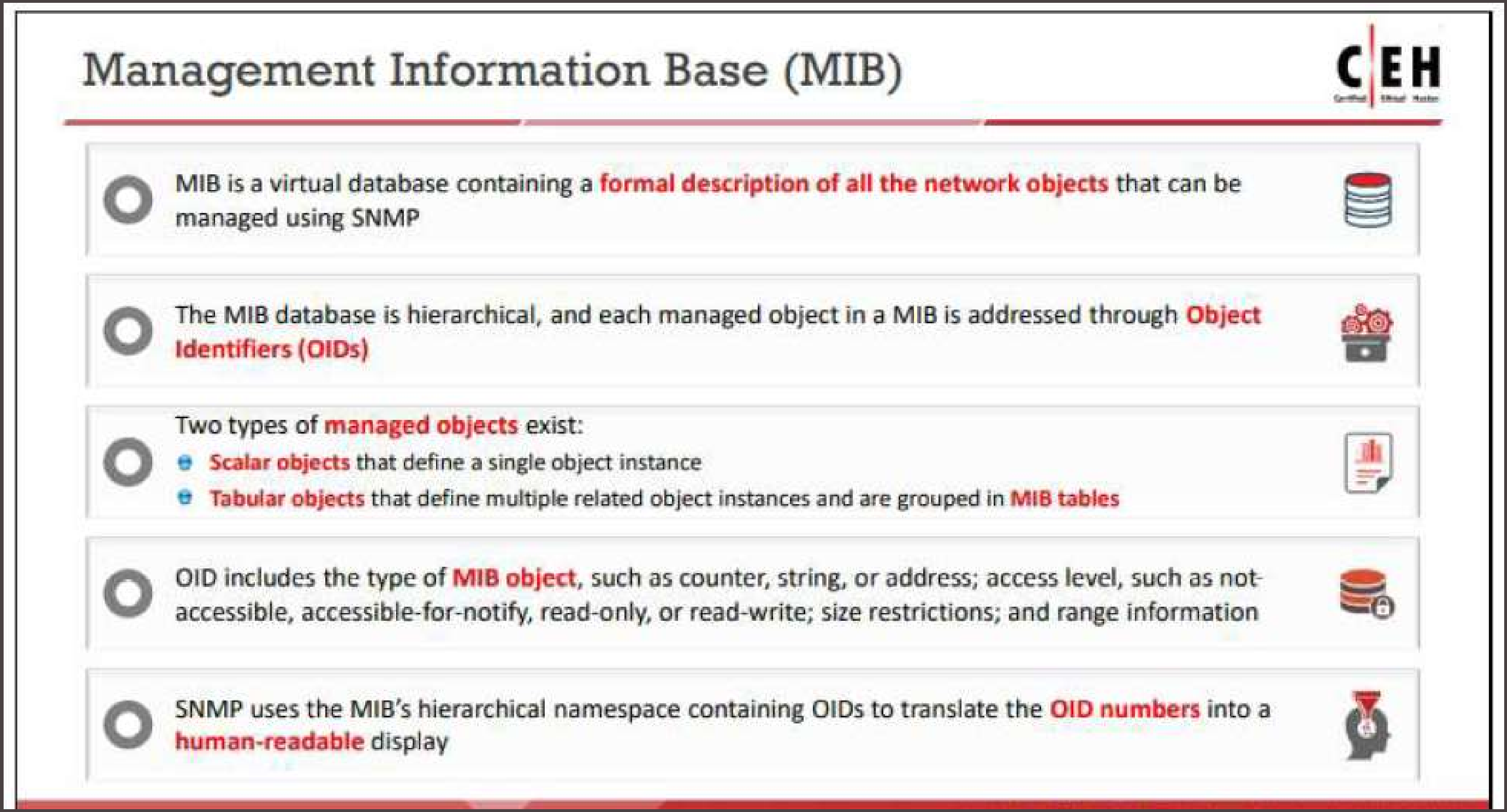


Working of SNMP

Management Information Base

MIB
Software version, hard drive space, session table
Community String: CompInfo IP: 10.10.2.1
Community String: CompInfo Community String: Alarm IP: 10.10.2.15
Sends request for active session (Community String: CompInfo, IP: 10.10.2.15)
Active Session Information (No. of sessions: 2, Comm: CompInfo, IP: 10.10.2.15)
Host X (SNMP Manager)
Host Y (SNMP Agent)
SNMPTRAP message for failed login
Community String: Alarm IP: 10.10.2.12
Host Z (SNMP Manager)

If the community string does not match the string stored in the management information base (MIB) database, then host Y will send a community string to a pre-configured SNMP manager indicating the error



Management Information Base (MIB)

CEH

- MIB is a virtual database containing a **formal description of all the network objects** that can be managed using SNMP

- The MIB database is hierarchical, and each managed object in a MIB is addressed through **Object Identifiers (OIDs)**

- Two types of **managed objects** exist:
  - **Scalar objects** that define a single object instance
  - **Tabular objects** that define multiple related object instances and are grouped in **MIB tables**

- OID includes the type of **MIB object**, such as counter, string, or address; access level, such as not-accessible, accessible-for-notify, read-only, or read-write; size restrictions; and range information

- SNMP uses the MIB's hierarchical namespace containing OIDs to translate the **OID numbers** into a **human-readable** display

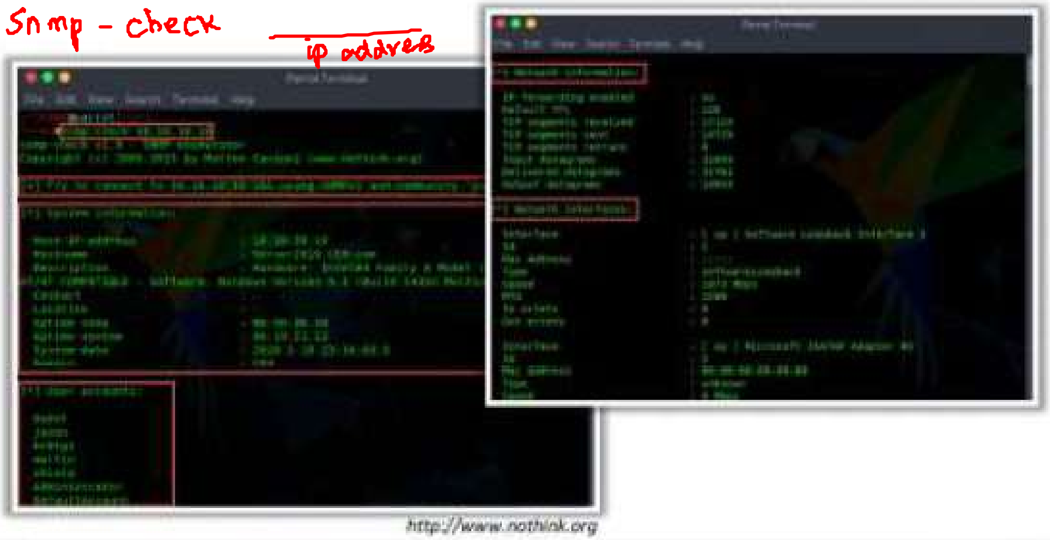
SNMP Enumeration Tools

C|EH Certified Ethical Hacker

Snmpcheck	Snmpcheck allows one to **enumerate** the **SNMP devices** and place the output in a very **human-readable** and friendly **format**

SoftPerfect Network Scanner	SoftPerfect Network Scanner **discovers shared folders** and retrieves practically any information about network devices **via WMI**, **SNMP**, **HTTP**, **SSH**, and **PowerShell**

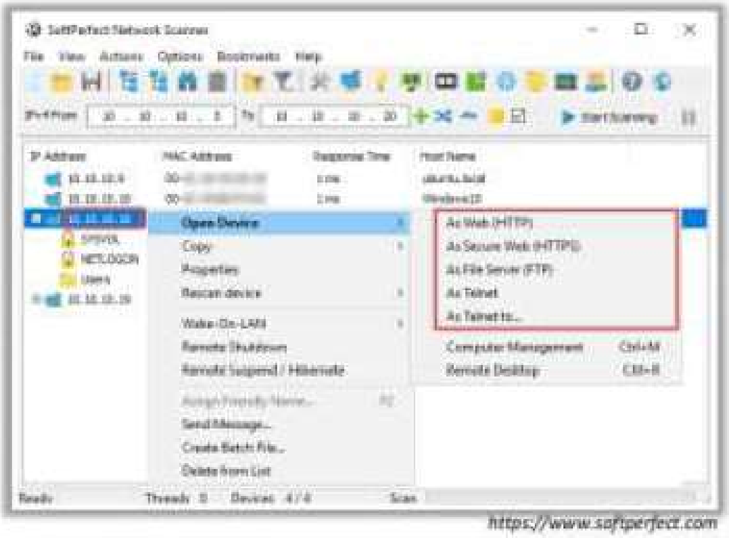
Snmp - check ___ ip addreß



```
[*] System information:

  Host IP address          : 10.10.10.[illegible]
  Hostname                 : [illegible]
  Description              : Hardware: [illegible] Family [illegible] Model [illegible]
  Contact                  : -
  Location                 : -
  Uptime snmp              : [illegible]
  Uptime system            : [illegible]
  System date              : [illegible]
  Domain                   : -

[*] User accounts:

  Guest
  Jason
  krbtgt
  martin
  shiela
  Administrator
  DefaultAccount
```

```
[*] Network information:

  IP forwarding enabled    : no
  Default TTL              : 128
  TCP segments received    : [illegible]
  TCP segments sent        : [illegible]
  TCP segments retrans     : [illegible]
  Input datagrams          : [illegible]
  Delivered datagrams      : [illegible]
  Output datagrams         : [illegible]

[*] Network interfaces:

  Interface                : [ up ] Software Loopback Interface 1
  Id                       : 1
  Mac Address              : :::::
  Type                     : softwareLoopback
  Speed                    : 1073 Mbps
  MTU                      : 1500
  In octets                : 0
  Out octets               : 0

  Interface                : [ up ] Microsoft ISATAP Adapter #2
  Id                       : [illegible]
  Mac Address              : 00:00:00:00:00:00
  Type                     : unknown
  Speed                    : 0 Mbps
```

http://www.nothink.org

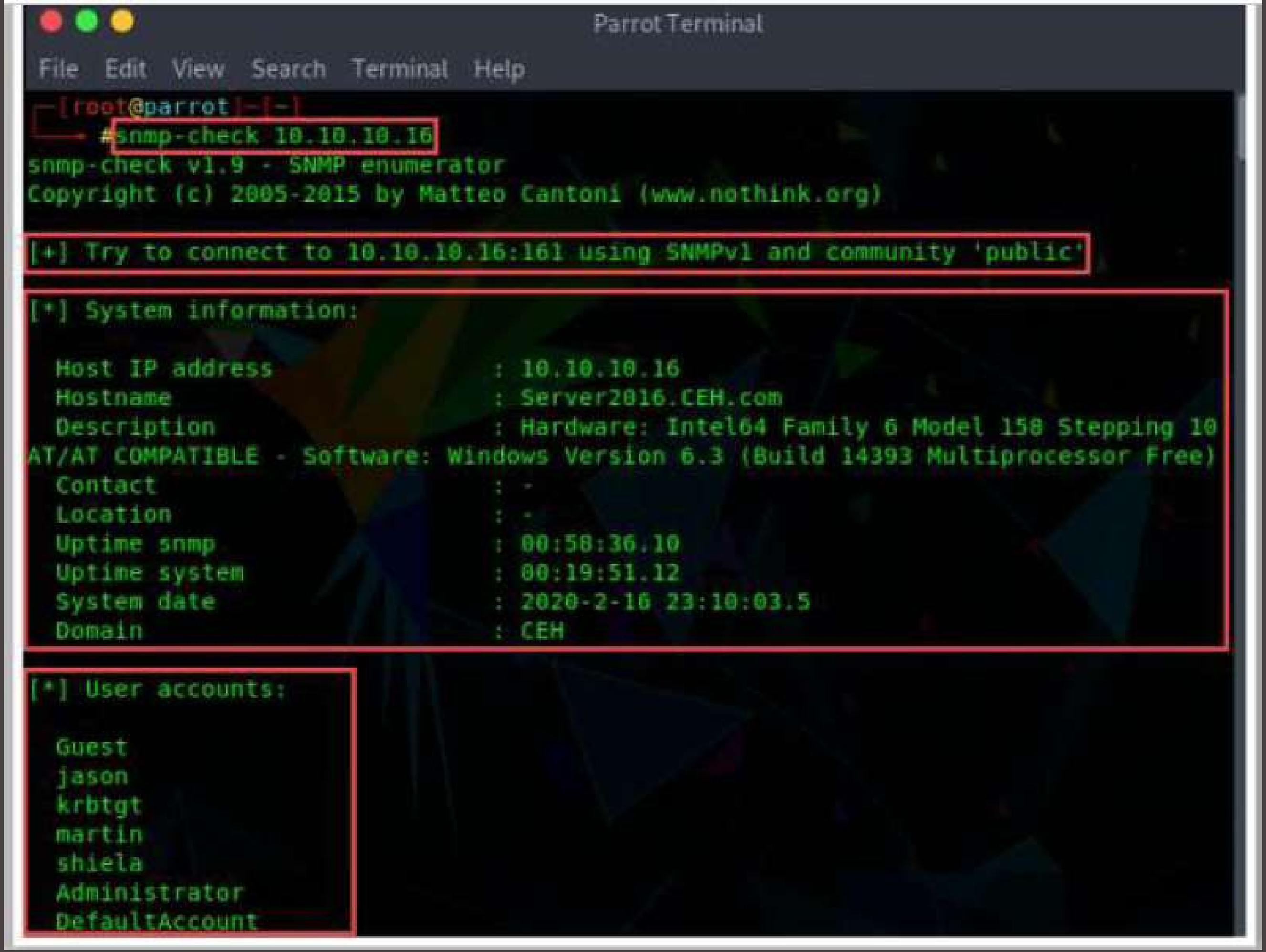


IP Address	MAC Address	Response Time	Host Name
10.10.10.9	00-[redacted]	1 ms	ubuntu.local
10.10.10.10	00-[redacted]	1 ms	Windows10
10.10.10.[illegible]			
SYSVOL			
NETLOGON			
Users			
10.10.10.19			

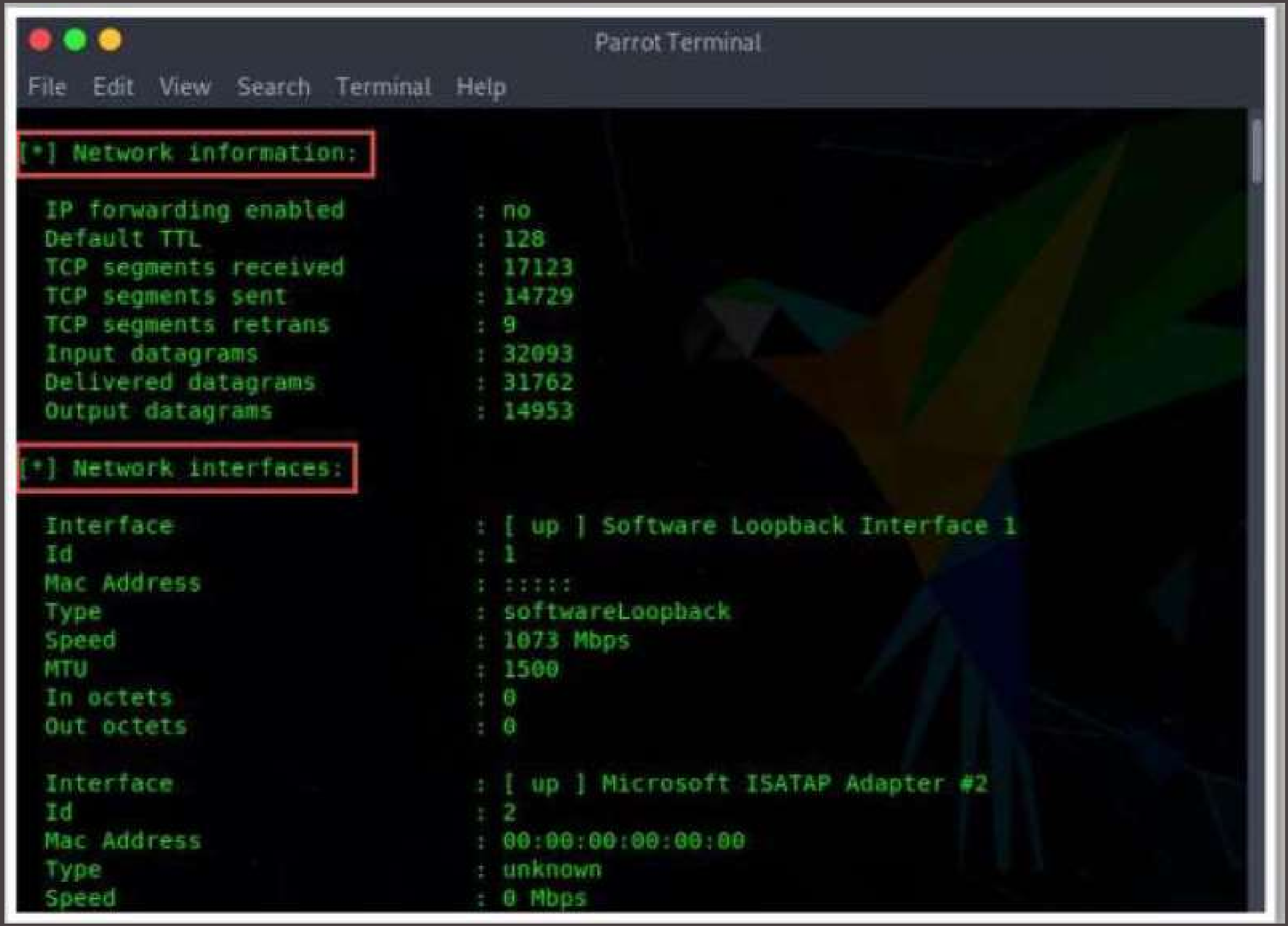
https://www.softperfect.com

Other SNMP Enumeration Tools:

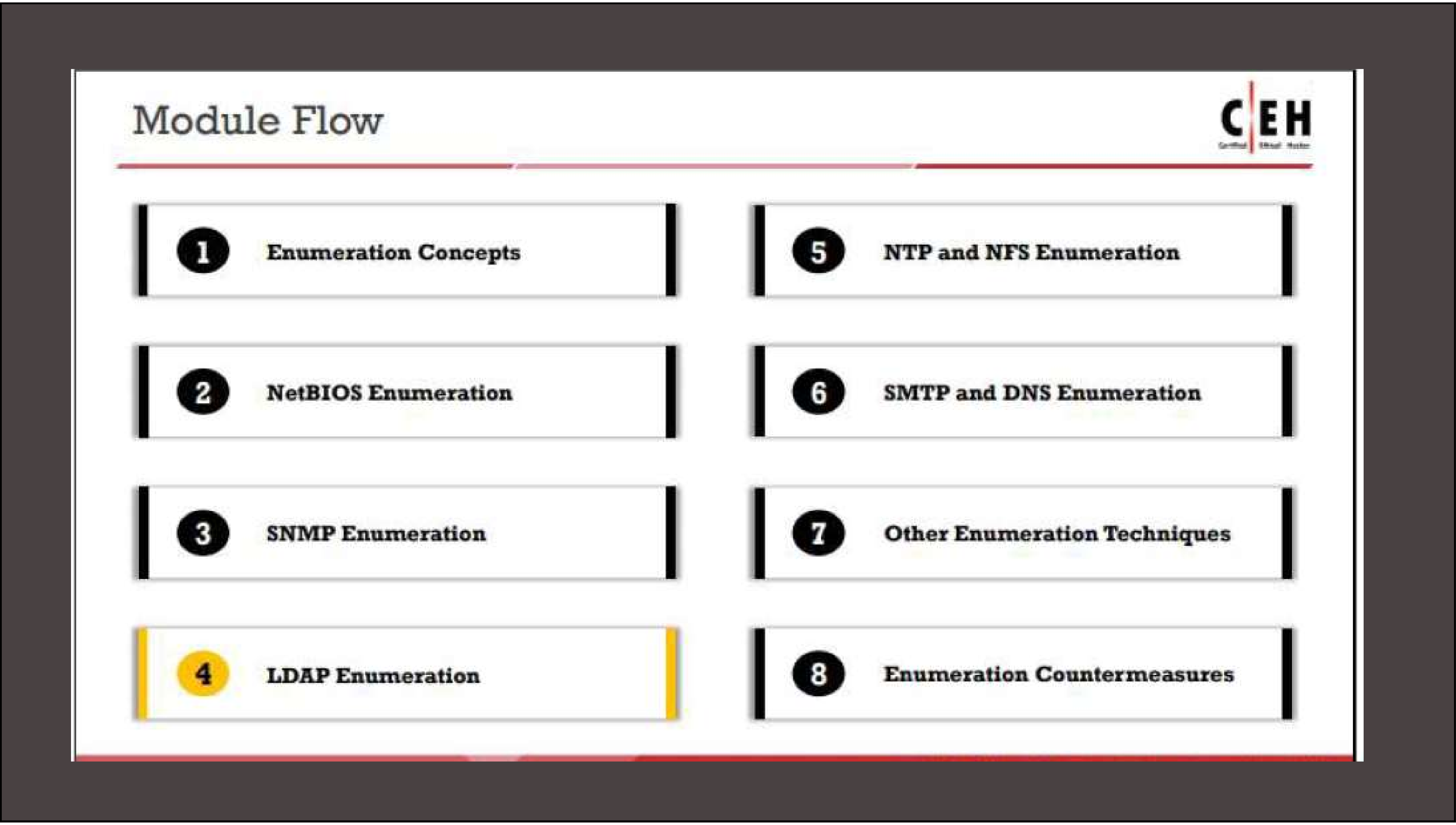
- **Network Performance Monitor** https://www.solarwinds.com
- **OpUtils** https://www.manageengine.com
- **PRTG Network Monitor** https://www.paessler.com
- **Engineer's Toolset** https://www.solarwinds.com



```
┌─[root@parrot]─[~]
└──╼ #snmp-check 10.10.10.16
snmp-check v1.9 - SNMP enumerator
Copyright (c) 2005-2015 by Matteo Cantoni (www.nothink.org)

[+] Try to connect to 10.10.10.16:161 using SNMPv1 and community 'public'

[*] System information:

  Host IP address               : 10.10.10.16
  Hostname                      : Server2016.CEH.com
  Description                   : Hardware: Intel64 Family 6 Model 158 Stepping 10
AT/AT COMPATIBLE - Software: Windows Version 6.3 (Build 14393 Multiprocessor Free)
  Contact                       : -
  Location                      : -
  Uptime snmp                   : 00:58:36.10
  Uptime system                 : 00:19:51.12
  System date                   : 2020-2-16 23:10:03.5
  Domain                        : CEH

[*] User accounts:

  Guest
  jason
  krbtgt
  martin
  shiela
  Administrator
  DefaultAccount
```

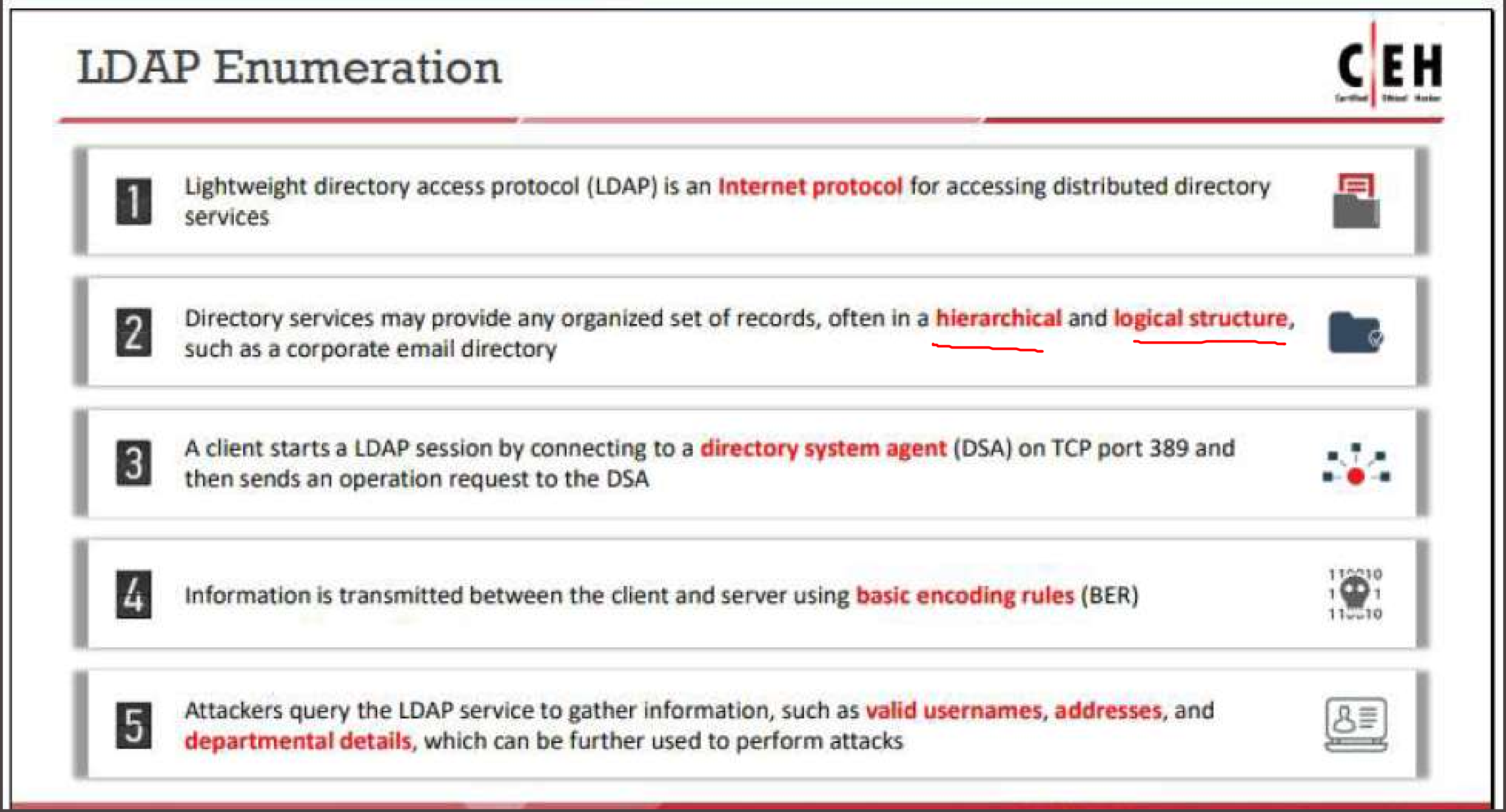


```
[*] Network information:

  IP forwarding enabled          : no
  Default TTL                    : 128
  TCP segments received          : 17123
  TCP segments sent              : 14729
  TCP segments retrans           : 9
  Input datagrams                : 32093
  Delivered datagrams            : 31762
  Output datagrams               : 14953

[*] Network interfaces:

  Interface                      : [ up ] Software Loopback Interface 1
  Id                             : 1
  Mac Address                    : :::::
  Type                           : softwareLoopback
  Speed                          : 1073 Mbps
  MTU                            : 1500
  In octets                      : 0
  Out octets                     : 0

  Interface                      : [ up ] Microsoft ISATAP Adapter #2
  Id                             : 2
  Mac Address                    : 00:00:00:00:00:00
  Type                           : unknown
  Speed                          : 0 Mbps
```



Module Flow

1. Enumeration Concepts
2. NetBIOS Enumeration
3. SNMP Enumeration
4. LDAP Enumeration
5. NTP and NFS Enumeration
6. SMTP and DNS Enumeration
7. Other Enumeration Techniques
8. Enumeration Countermeasures



LDAP Enumeration

1. Lightweight directory access protocol (LDAP) is an **Internet protocol** for accessing distributed directory services

2. Directory services may provide any organized set of records, often in a **hierarchical** and **logical structure**, such as a corporate email directory

3. A client starts a LDAP session by connecting to a **directory system agent** (DSA) on TCP port 389 and then sends an operation request to the DSA

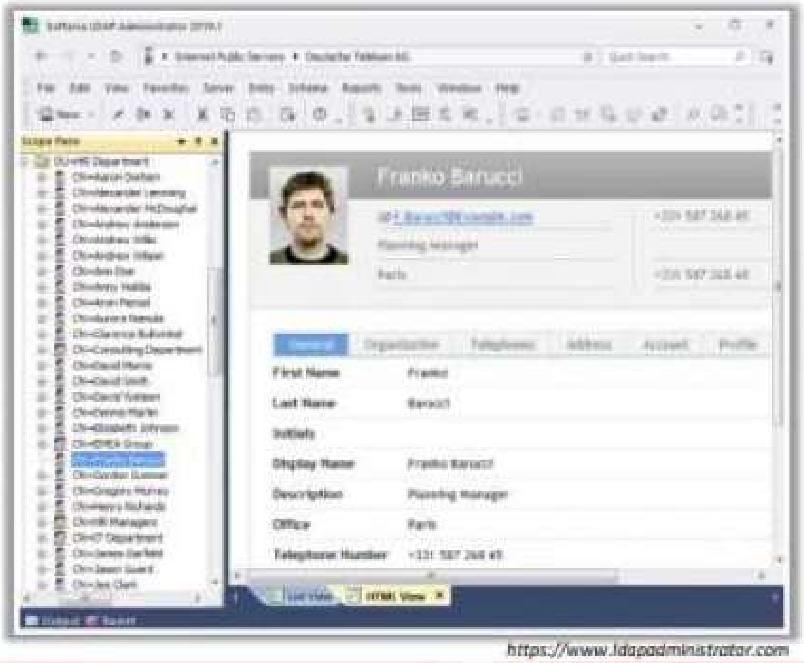
4. Information is transmitted between the client and server using **basic encoding rules** (BER)

5. Attackers query the LDAP service to gather information, such as **valid usernames**, **addresses**, and **departmental details**, which can be further used to perform attacks

LDAP Enumeration Tools

Softerra LDAP Administrator

Softerra LDAP Administrator provides various features essential for **LDAP development**, deployment, and **administration of directories**


File Edit View Favorites Server Entry Schema Reports Tools Window Help
Scope Pane
Franko Barucci
Planning Manager
Paris
General Organization Telephones Address Account Profile
List View
HTML View

First Name Franko

Last Name Barucci

Initials

Display Name Franko Barucci

Description Planning Manager

Office Paris

Telephone Number +331 587 268 45

https://www.ldapadministrator.com

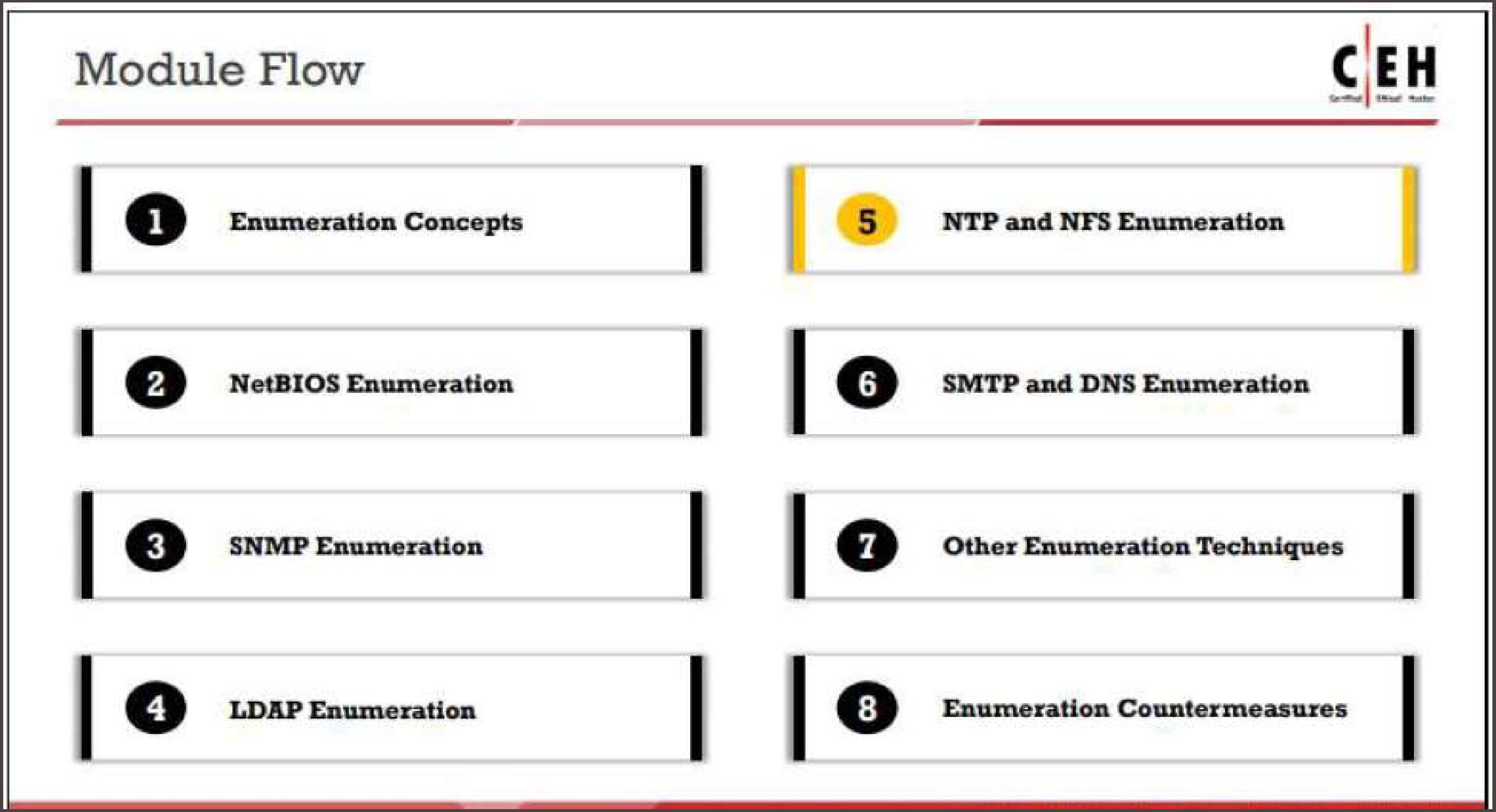
LDAP Admin Tool
https://www.ldapsoft.com

LDAP Account Manager
https://www.ldap-account-manager.org

LDAP Search
https://securityxploded.com

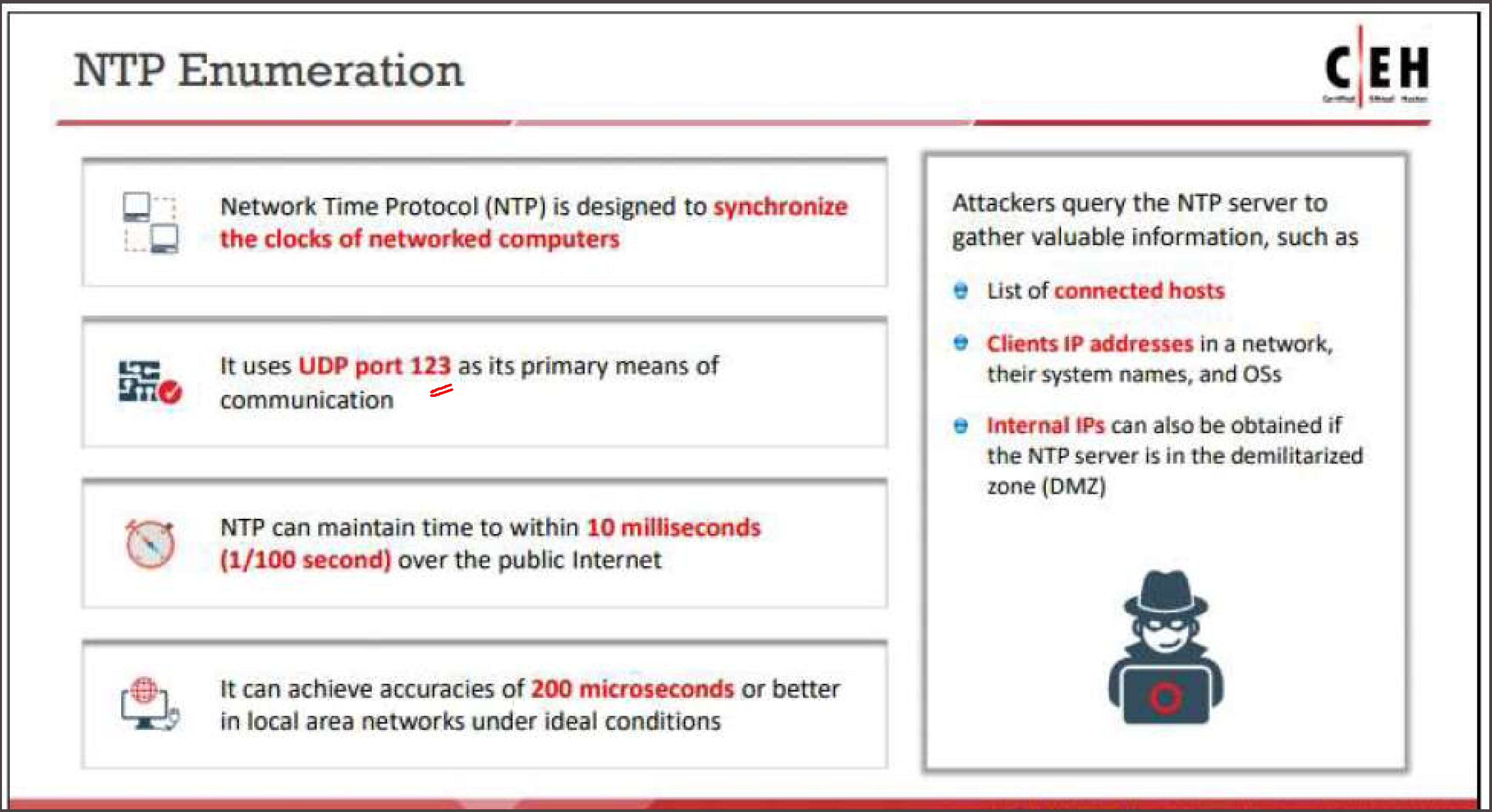
JXplorer
http://www.jxplorer.org

Active Directory Explorer (AD Explorer)
https://docs.microsoft.com



Module Flow

1. Enumeration Concepts
2. NetBIOS Enumeration
3. SNMP Enumeration
4. LDAP Enumeration
5. NTP and NFS Enumeration
6. SMTP and DNS Enumeration
7. Other Enumeration Techniques
8. Enumeration Countermeasures



NTP Enumeration

Network Time Protocol (NTP) is designed to **synchronize the clocks of networked computers**

It uses **UDP port 123** as its primary means of communication

NTP can maintain time to within **10 milliseconds (1/100 second)** over the public Internet

It can achieve accuracies of **200 microseconds** or better in local area networks under ideal conditions

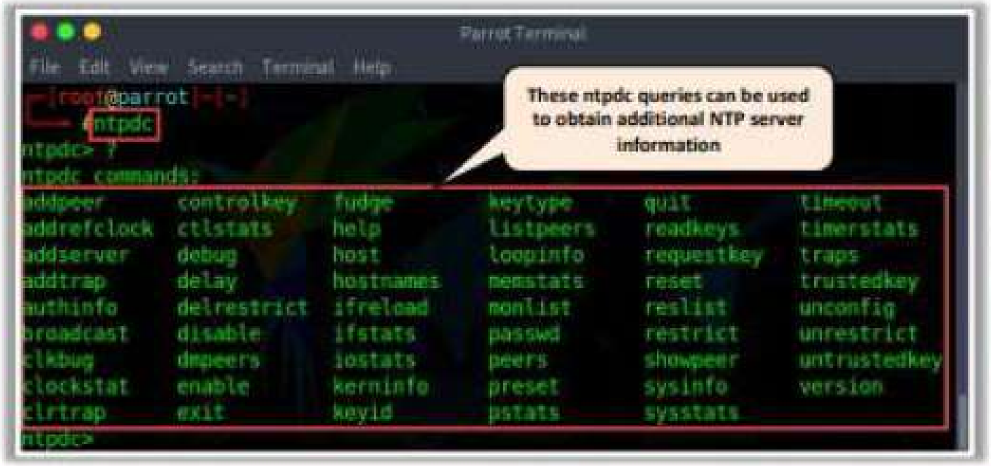
Attackers query the NTP server to gather valuable information, such as

- List of **connected hosts**
- **Clients IP addresses** in a network, their system names, and OSs
- **Internal IPs** can also be obtained if the NTP server is in the demilitarized zone (DMZ)

NTP Enumeration Commands

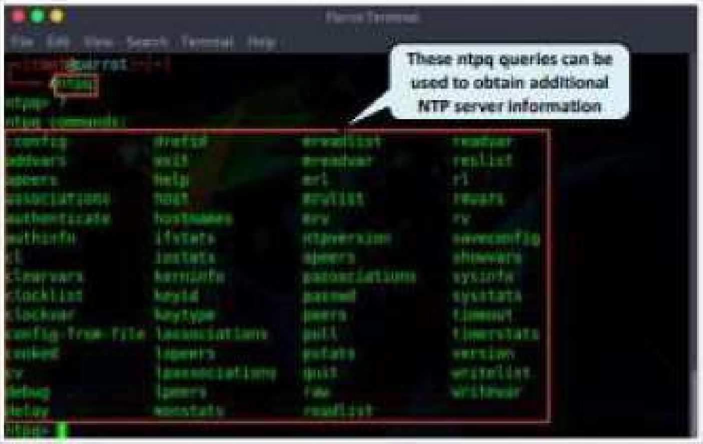
- **ntptrace**
  - Traces a chain of NTP servers back to the primary source
  - `ntptrace [-n] [-m maxhosts] [servername/IP_address]`
- **ntpdc**
  - Monitors operation of the NTP daemon, ntpd
  - `ntpdc [-ilnps] [-c command] [host] [...]`

- **ntpq**
  - Monitors NTP daemon (ntpd) operations and determines performance
  - `ntpq [-inp] [-c command] [host] [...]`



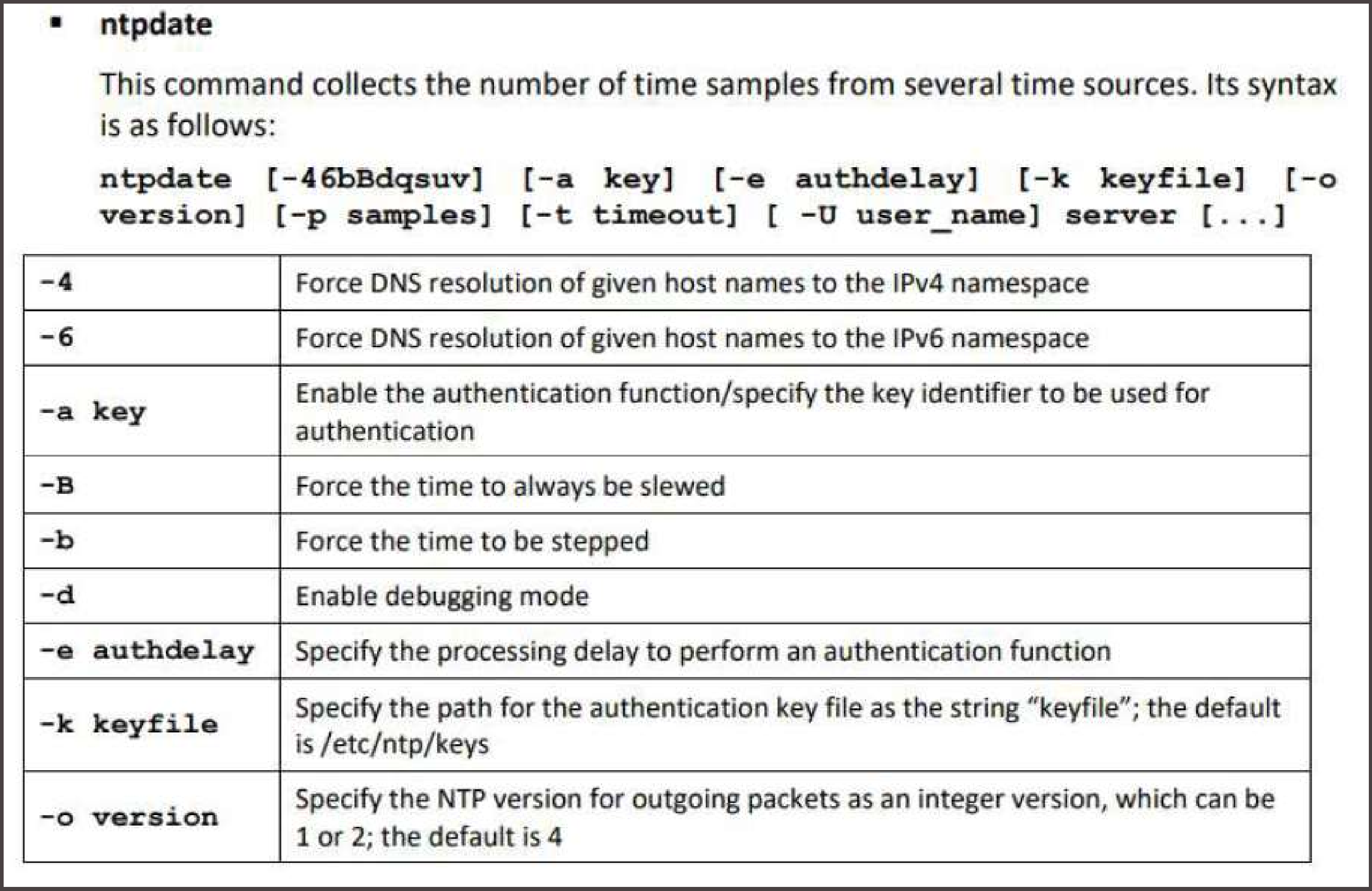
```
┌─[root@parrot]─[~]
└──╼ #ntpdc
ntpdc> ?
ntpdc commands:
addpeer      controlkey   fudge        keytype      quit         timeout
addrefclock  ctlstats     help         listpeers    readkeys     timerstats
addserver    debug        host         loopinfo     requestkey   traps
addtrap      delay        hostnames    memstats     reset        trustedkey
authinfo     delrestrict  ifreload     monlist      reslist      unconfig
broadcast    disable      ifstats      passwd       restrict     unrestrict
clkbug       dmpeers      iostats      peers        showpeer     untrustedkey
clockstat    enable       kerninfo     preset       sysinfo      version
clrtrap      exit         keyid        pstats       sysstats
ntpdc>
```

These ntpdc queries can be used to obtain additional NTP server information



```
┌─[root@parrot]─[~]
└──╼ #ntpq
ntpq> ?
ntpq commands:
:config          drefid           mreadlist        readvar
addvars          exit             mreadvar         reslist
apeers           help             mrl              rl
associations     host             mrulist          rmvars
authenticate     hostnames        mrv              rv
authinfo         ifstats          ntpversion       saveconfig
cl               iostats          opeers           showvars
clearvars        kerninfo         passociations    sysinfo
clocklist        keyid            passwd           sysstats
clockvar         keytype          peers            timeout
config-from-file lassociations    poll             timerstats
cooked           lopeers          pstats           version
cv               lpassociations   quit             writelist
debug            lpeers           raw              writevar
delay            monstats         readlist
ntpq>
```

These ntpq queries can be used to obtain additional NTP server information

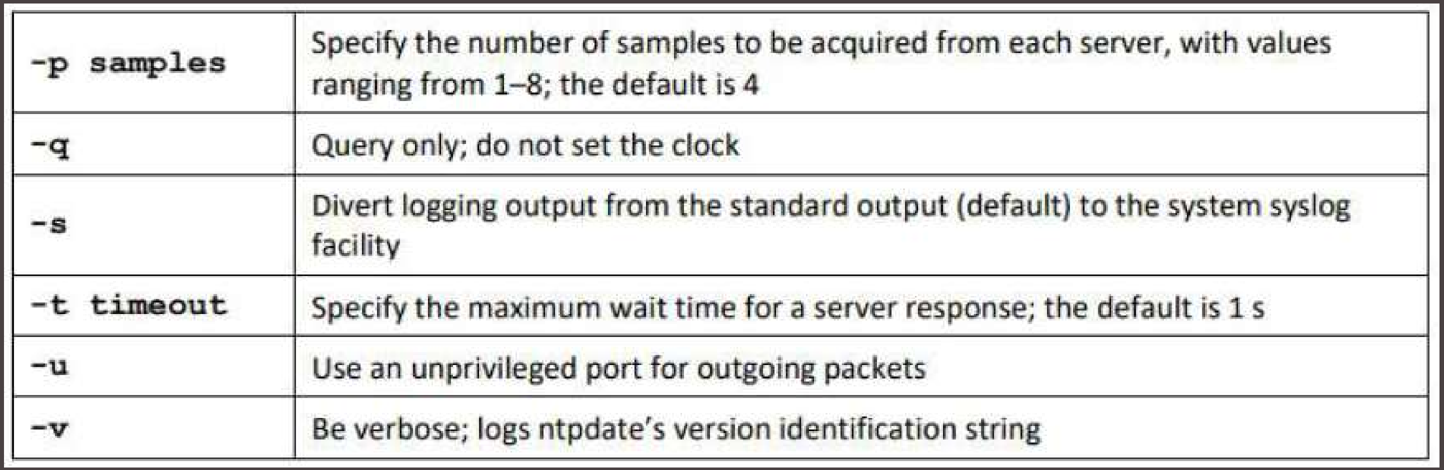


- **ntpdate**

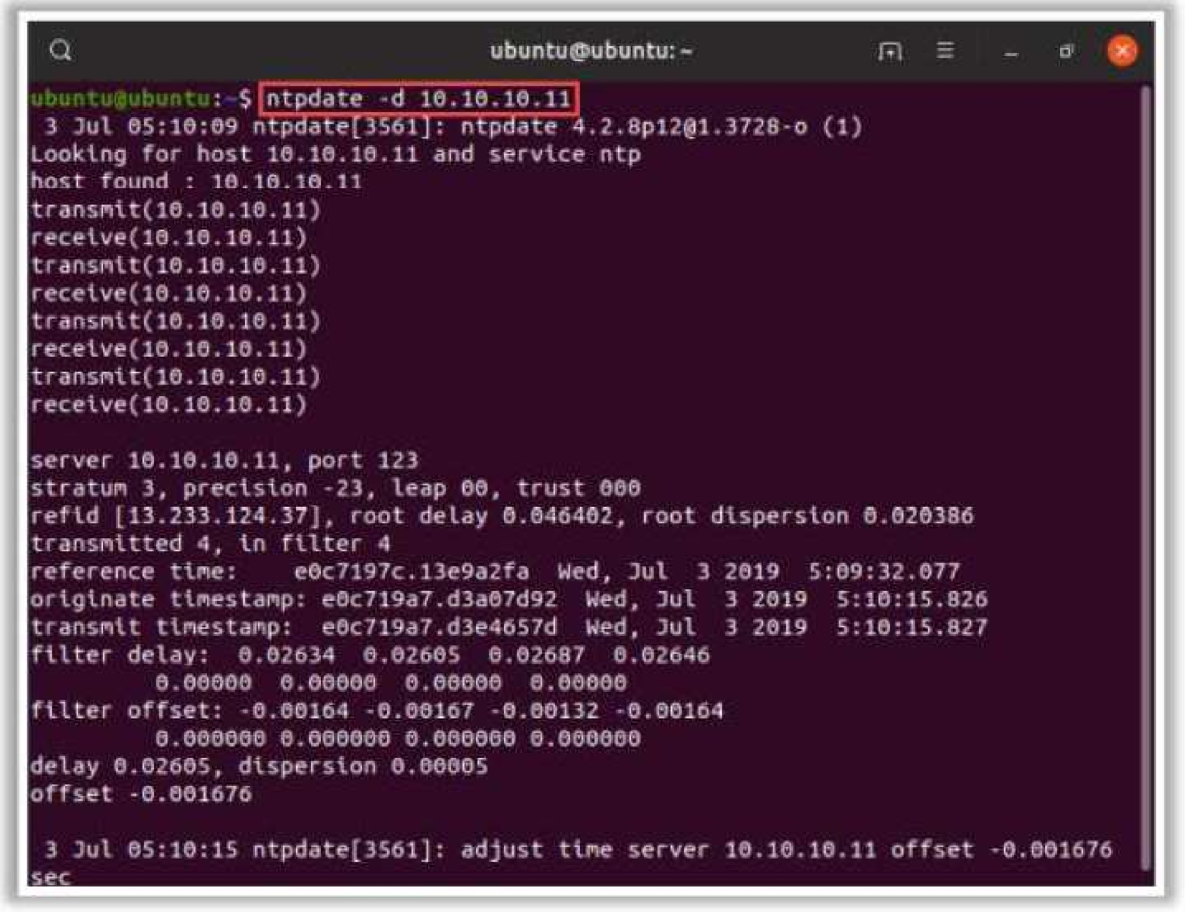
  This command collects the number of time samples from several time sources. Its syntax is as follows:

```
ntpdate [-46bBdqsuv] [-a key] [-e authdelay] [-k keyfile] [-o
version] [-p samples] [-t timeout] [ -U user_name] server [...]
```

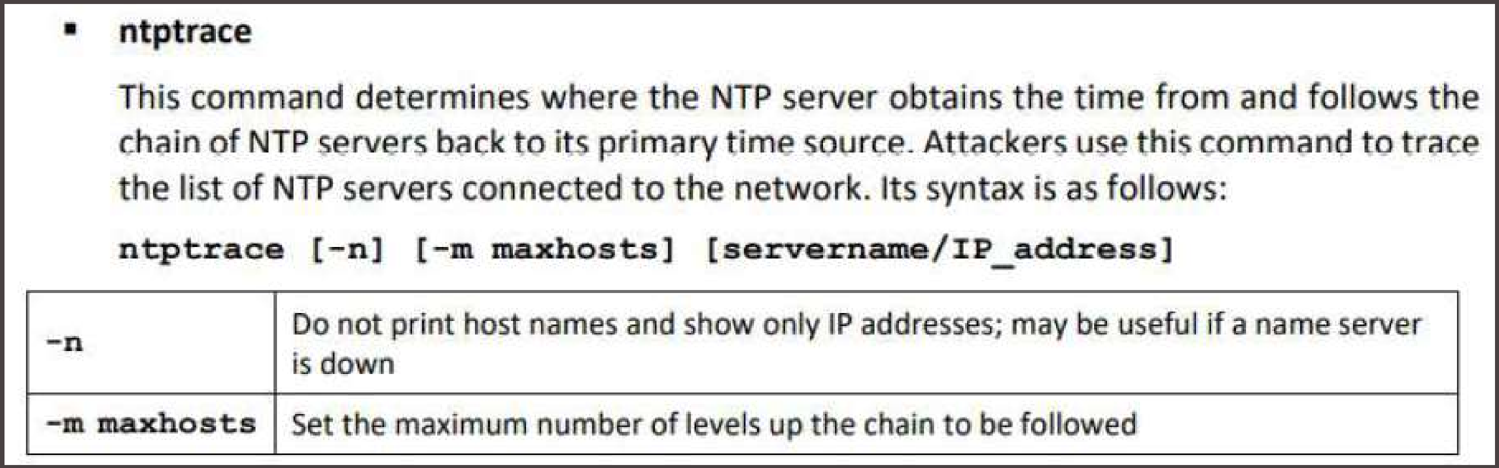
`-4`	Force DNS resolution of given host names to the IPv4 namespace
`-6`	Force DNS resolution of given host names to the IPv6 namespace
`-a key`	Enable the authentication function/specify the key identifier to be used for authentication
`-B`	Force the time to always be slewed
`-b`	Force the time to be stepped
`-d`	Enable debugging mode
`-e authdelay`	Specify the processing delay to perform an authentication function
`-k keyfile`	Specify the path for the authentication key file as the string “keyfile”; the default is /etc/ntp/keys
`-o version`	Specify the NTP version for outgoing packets as an integer version, which can be 1 or 2; the default is 4



`-p samples`	Specify the number of samples to be acquired from each server, with values ranging from 1–8; the default is 4
`-q`	Query only; do not set the clock
`-s`	Divert logging output from the standard output (default) to the system syslog facility
`-t timeout`	Specify the maximum wait time for a server response; the default is 1 s
`-u`	Use an unprivileged port for outgoing packets
`-v`	Be verbose; logs ntpdate’s version identification string



```
ubuntu@ubuntu:~$ ntpdate -d 10.10.10.11
 3 Jul 05:10:09 ntpdate[3561]: ntpdate 4.2.8p12@1.3728-o (1)
Looking for host 10.10.10.11 and service ntp
host found : 10.10.10.11
transmit(10.10.10.11)
receive(10.10.10.11)
transmit(10.10.10.11)
receive(10.10.10.11)
transmit(10.10.10.11)
receive(10.10.10.11)
transmit(10.10.10.11)
receive(10.10.10.11)

server 10.10.10.11, port 123
stratum 3, precision -23, leap 00, trust 000
refid [13.233.124.37], root delay 0.046402, root dispersion 0.020386
transmitted 4, in filter 4
reference time:    e0c7197c.13e9a2fa  Wed, Jul  3 2019  5:09:32.077
originate timestamp: e0c719a7.d3a07d92  Wed, Jul  3 2019  5:10:15.826
transmit timestamp:  e0c719a7.d3e4657d  Wed, Jul  3 2019  5:10:15.827
filter delay:  0.02634  0.02605  0.02687  0.02646
         0.00000  0.00000  0.00000  0.00000
filter offset: -0.00164 -0.00167 -0.00132 -0.00164
         0.000000 0.000000 0.000000 0.000000
delay 0.02605, dispersion 0.00005
offset -0.001676

 3 Jul 05:10:15 ntpdate[3561]: adjust time server 10.10.10.11 offset -0.001676
sec
```

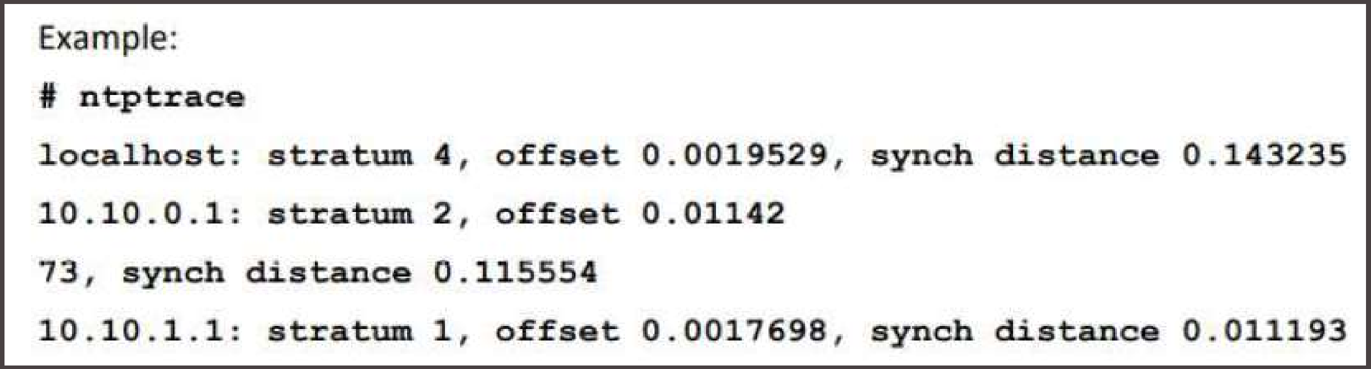


- **ntptrace**

  This command determines where the NTP server obtains the time from and follows the chain of NTP servers back to its primary time source. Attackers use this command to trace the list of NTP servers connected to the network. Its syntax is as follows:

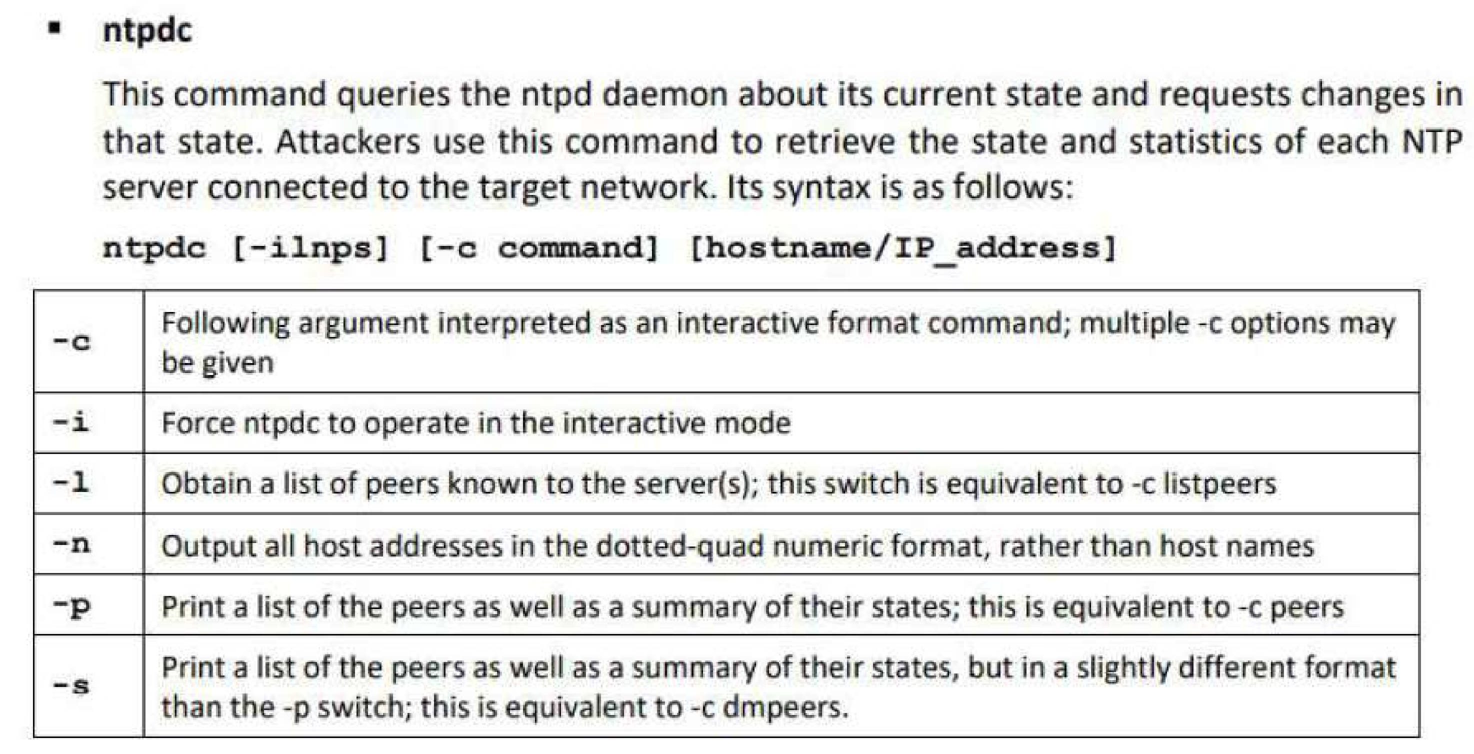
```
ntptrace [-n] [-m maxhosts] [servername/IP_address]
```

`-n`	Do not print host names and show only IP addresses; may be useful if a name server is down
`-m maxhosts`	Set the maximum number of levels up the chain to be followed



Example:

```
# ntptrace
localhost: stratum 4, offset 0.0019529, synch distance 0.143235
10.10.0.1: stratum 2, offset 0.01142
73, synch distance 0.115554
10.10.1.1: stratum 1, offset 0.0017698, synch distance 0.011193
```

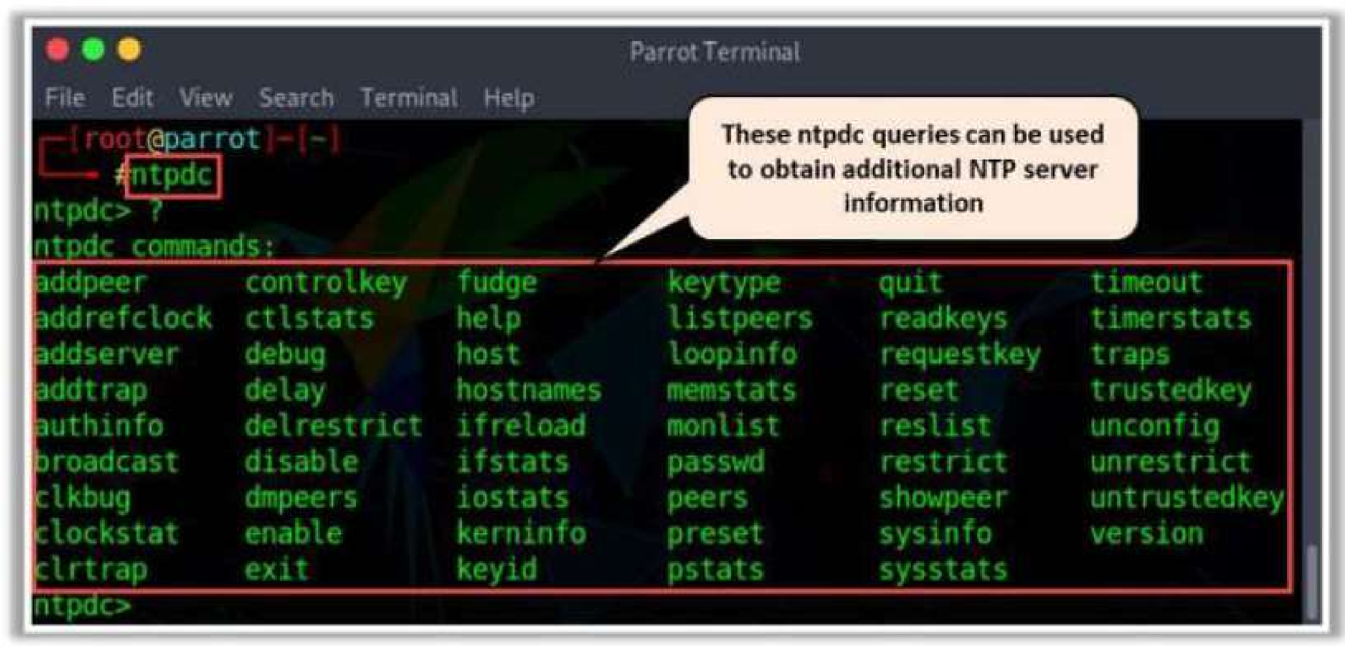


- **ntpdc**

  This command queries the ntpd daemon about its current state and requests changes in that state. Attackers use this command to retrieve the state and statistics of each NTP server connected to the target network. Its syntax is as follows:

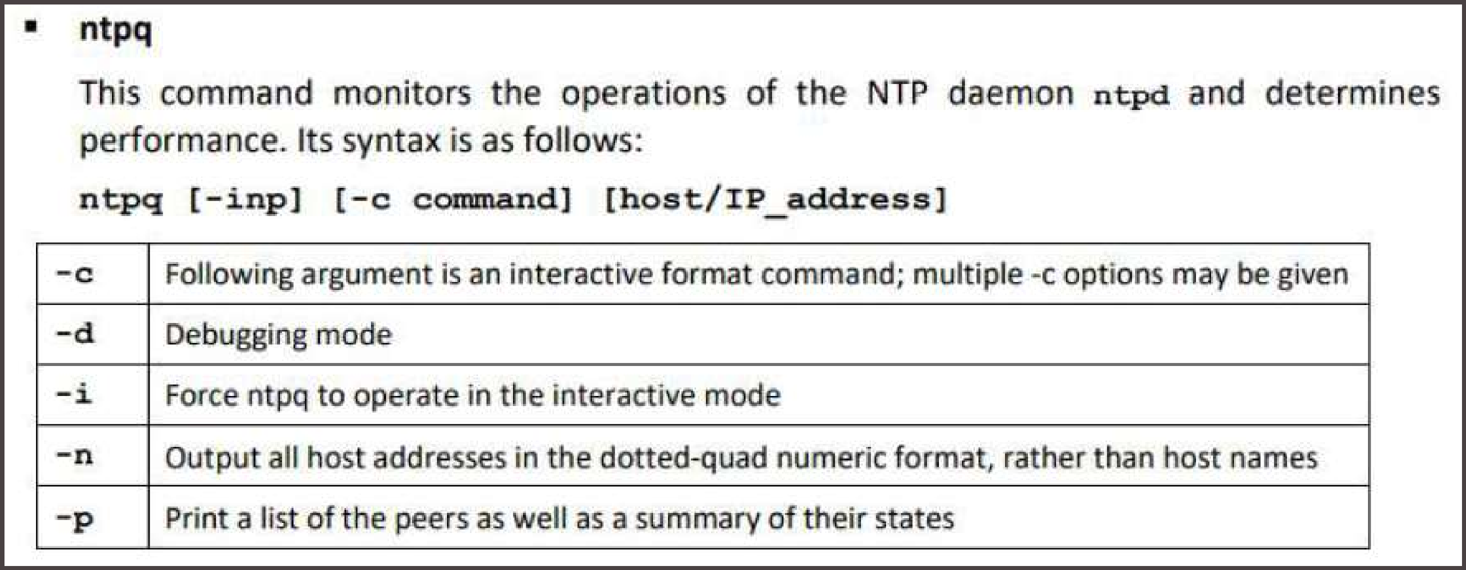
```
ntpdc [-ilnps] [-c command] [hostname/IP_address]
```

`-c`	Following argument interpreted as an interactive format command; multiple -c options may be given
`-i`	Force ntpdc to operate in the interactive mode
`-l`	Obtain a list of peers known to the server(s); this switch is equivalent to -c listpeers
`-n`	Output all host addresses in the dotted-quad numeric format, rather than host names
`-p`	Print a list of the peers as well as a summary of their states; this is equivalent to -c peers
`-s`	Print a list of the peers as well as a summary of their states, but in a slightly different format than the -p switch; this is equivalent to -c dmpeers.



```
┌─[root@parrot]─[~]
└──╼ #ntpdc
ntpdc> ?
ntpdc commands:
addpeer      controlkey   fudge        keytype      quit         timeout
addrefclock  ctlstats     help         listpeers    readkeys     timerstats
addserver    debug        host         loopinfo     requestkey   traps
addtrap      delay        hostnames    memstats     reset        trustedkey
authinfo     delrestrict  ifreload     monlist      reslist      unconfig
broadcast    disable      ifstats      passwd       restrict     unrestrict
clkbug       dmpeers      iostats      peers        showpeer     untrustedkey
clockstat    enable       kerninfo     preset       sysinfo      version
clrtrap      exit         keyid        pstats       sysstats
ntpdc>
```

These ntpdc queries can be used to obtain additional NTP server information



- **ntpq**

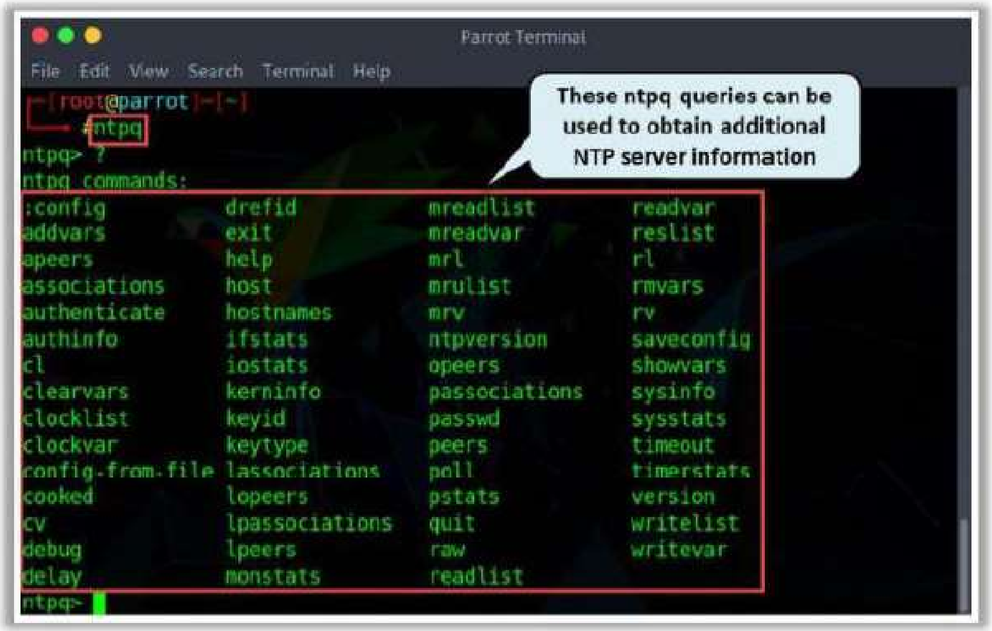
This command monitors the operations of the NTP daemon `ntpd` and determines performance. Its syntax is as follows:

```
ntpq [-inp] [-c command] [host/IP_address]
```

`-c`	Following argument is an interactive format command; multiple -c options may be given
`-d`	Debugging mode
`-i`	Force ntpq to operate in the interactive mode
`-n`	Output all host addresses in the dotted-quad numeric format, rather than host names
`-p`	Print a list of the peers as well as a summary of their states

Example:

```
ntpq> version
ntpq 4.2.8p10@1.3728-o
ntpq> host
current host is localhost
```

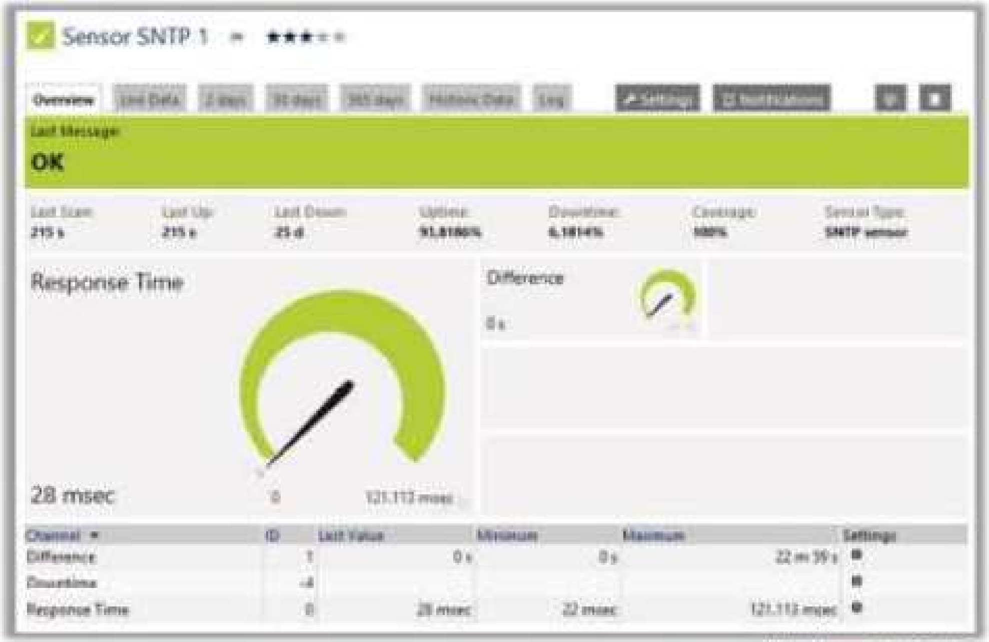


```
┌─[root@parrot]─[~]
└──╼ #ntpq
ntpq> ?
ntpq commands:
:config          drefid          mreadlist       readvar
addvars          exit            mreadvar        reslist
apeers           help            mrl             rl
associations     host            mrulist         rmvars
authenticate     hostnames       mrv             rv
authinfo         ifstats         ntpversion      saveconfig
cl               iostats         opeers          showvars
clearvars        kerninfo        passociations   sysinfo
clocklist        keyid           passwd          sysstats
clockvar         keytype         peers           timeout
config-from-file lassociations   poll            timerstats
cooked           lopeers         pstats          version
cv               lpassociations  quit            writelist
debug            lpeers          raw             writevar
delay            monstats        readlist
ntpq>
```

These ntpq queries can be used to obtain additional NTP server information

NTP Enumeration Tools

- **PRTG Network Monitor** includes **SNTP Sensor monitor**, a simple network time protocol (SNTP) server that shows the response time of the server and time difference in comparison to the local system time



Last Message
OK

Last Scan	Last Up	Last Down	Uptime	Downtime	Coverage	Sensor Type
215 s	215 s	25 d	93,8186%	6,1814%	100%	SNTP sensor

Response Time 28 msec

Difference 0 s

Channel ▾	ID	Last Value	Minimum	Maximum	Settings
Difference	1	0 s	0 s	22 m 59 s	●
Downtime	-4				●
Response Time	0	28 msec	22 msec	121.113 msec	●

https://www.paessler.com

NTP Enumeration Tools

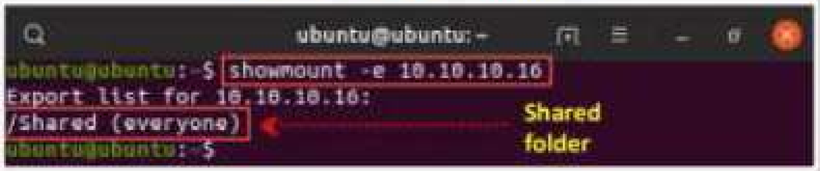
- Nmap (*https://nmap.org*)
- Wireshark (*https://www.wireshark.org*)
- udp-proto-scanner (*https://labs.portcullis.co.uk*)
- NTP Server Scanner (*http://www.bytefusion.com*)

NFS Enumeration

- The NFS system is generally implemented on the computer network, where the **centralization of data** is required for critical resources

- NFS enumeration enables attackers to identify the **exported directories**, **list of clients** connected to the NFS server along with their **IP addresses**, and the **shared data** associated with the IP addresses

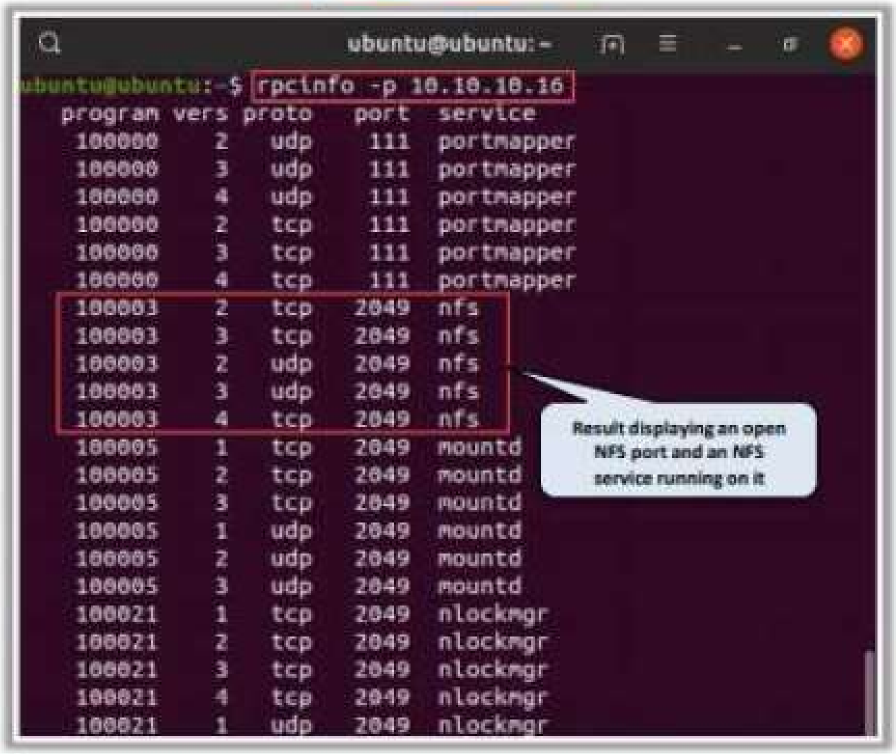
showmount command



```
ubuntu@ubuntu:~$ showmount -e 10.10.10.16
Export list for 10.10.10.16:
/Shared (everyone)
ubuntu@ubuntu:~$
```

Shared folder

rpcinfo command



```
ubuntu@ubuntu:~$ rpcinfo -p 10.10.10.16
   program vers proto   port  service
    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
    100003    2   tcp   2049  nfs
    100003    3   tcp   2049  nfs
    100003    2   udp   2049  nfs
    100003    3   udp   2049  nfs
    100003    4   tcp   2049  nfs
    100005    1   tcp   2049  mountd
    100005    2   tcp   2049  mountd
    100005    3   tcp   2049  mountd
    100005    1   udp   2049  mountd
    100005    2   udp   2049  mountd
    100005    3   udp   2049  mountd
    100021    1   tcp   2049  nlockmgr
    100021    2   tcp   2049  nlockmgr
    100021    3   tcp   2049  nlockmgr
    100021    4   tcp   2049  nlockmgr
    100021    1   udp   2049  nlockmgr
```

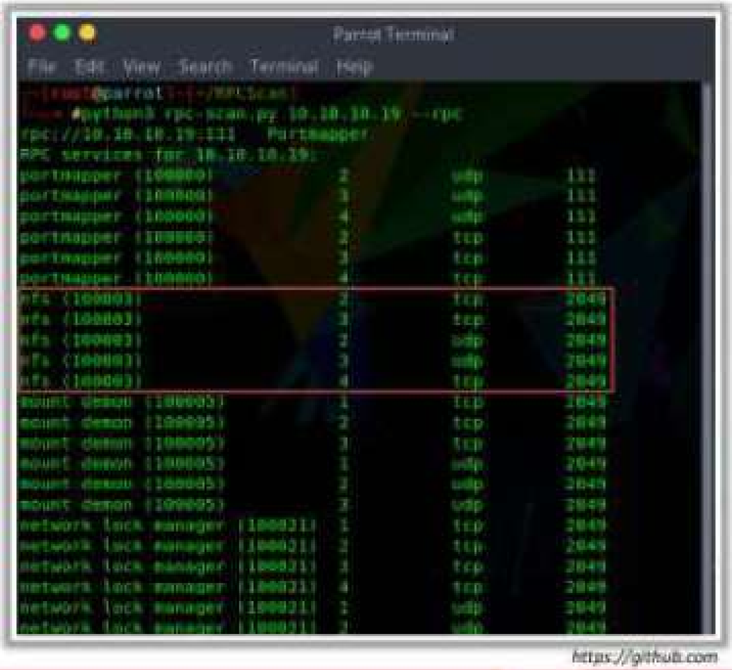
Result displaying an open NFS port and an NFS service running on it

python3 rpc-scan.py ______ --rpc
Ipaddress

NFS Enumeration Tools

./ ______
filename

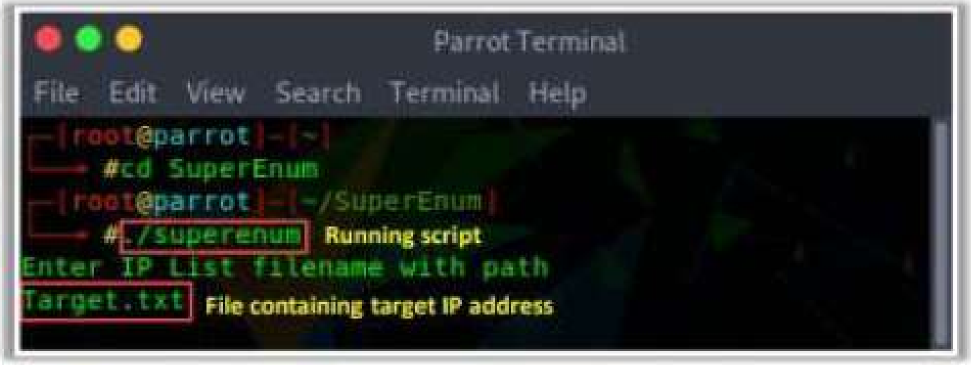
RPCScan RPSCan communicates with RPC services and checks misconfigurations on NFS shares



```
┌─[root@parrot]─[~/RPCScan]
└──╼ #python3 rpc-scan.py 10.10.10.19 --rpc
rpc://10.10.10.19:111    Portmapper
RPC services for 10.10.10.19:
portmapper (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
nfs (100003)                      2            tcp          2049
nfs (100003)                      3            tcp          2049
nfs (100003)                      2            udp          2049
nfs (100003)                      3            udp          2049
nfs (100003)                      4            tcp          2049
mount demon (100005)              1            tcp          2049
mount demon (100005)              2            tcp          2049
mount demon (100005)              3            tcp          2049
mount demon (100005)              1            udp          2049
mount demon (100005)              2            udp          2049
mount demon (100005)              3            udp          2049
network lock manager (100021)     1            tcp          2049
network lock manager (100021)     2            tcp          2049
network lock manager (100021)     3            tcp          2049
network lock manager (100021)     4            tcp          2049
network lock manager (100021)     1            udp          2049
network lock manager (100021)     2            udp          2049
```

https://github.com

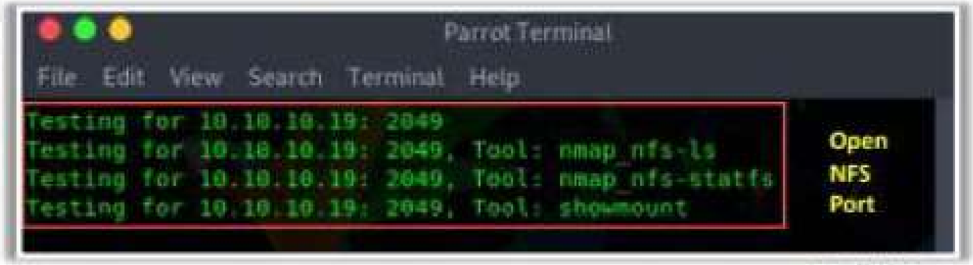
SuperEnum SuperEnum includes a script that does the basic enumeration of any open port



```
┌─[root@parrot]─[~]
└──╼ #cd SuperEnum
┌─[root@parrot]─[~/SuperEnum]
└──╼ #./superenum
Enter IP List filename with path
Target.txt
```

Running script

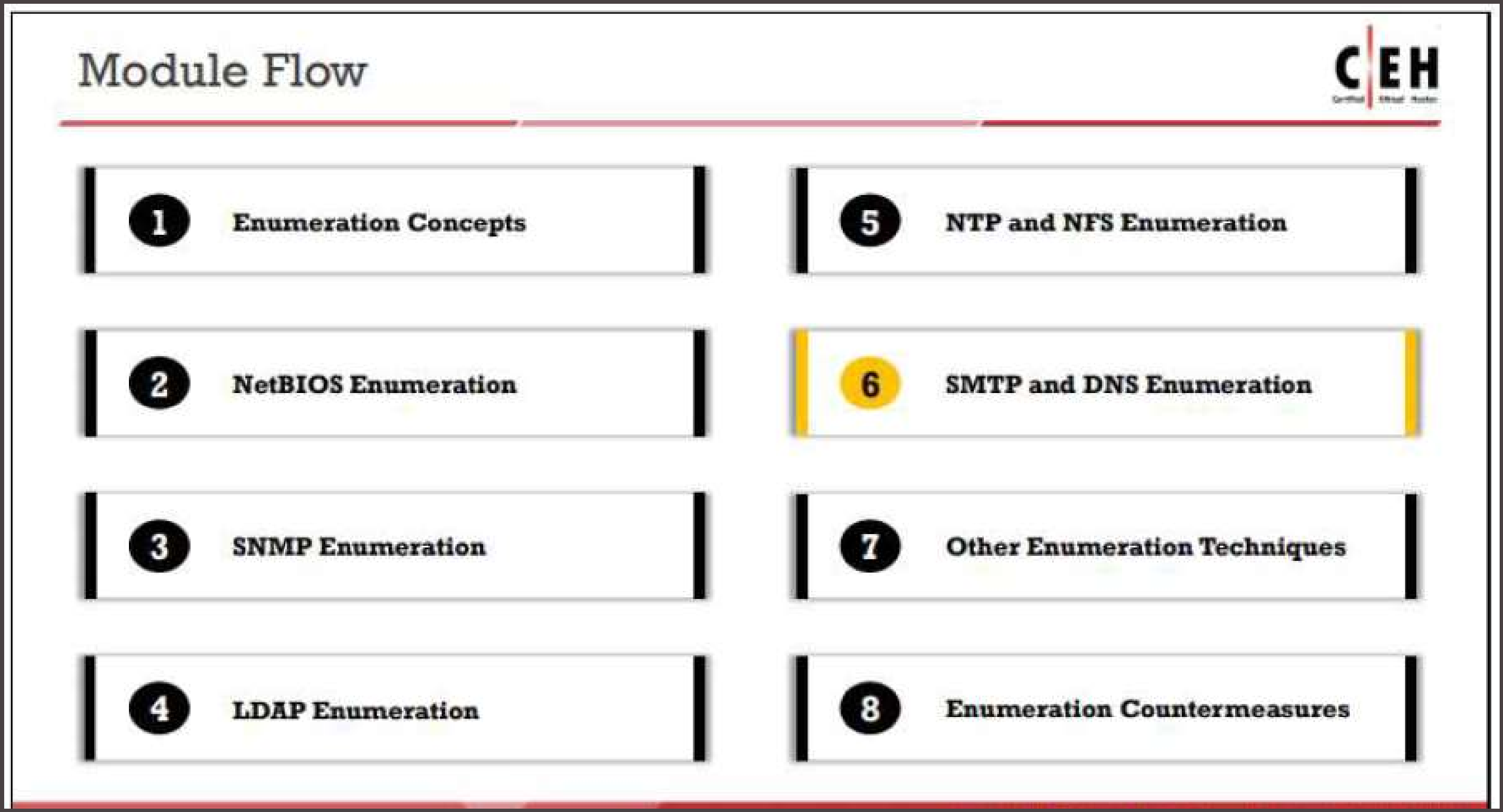
File containing target IP address



```
Testing for 10.10.10.19: 2049
Testing for 10.10.10.19: 2049, Tool: nmap_nfs-ls
Testing for 10.10.10.19: 2049, Tool: nmap_nfs-statfs
Testing for 10.10.10.19: 2049, Tool: showmount
```

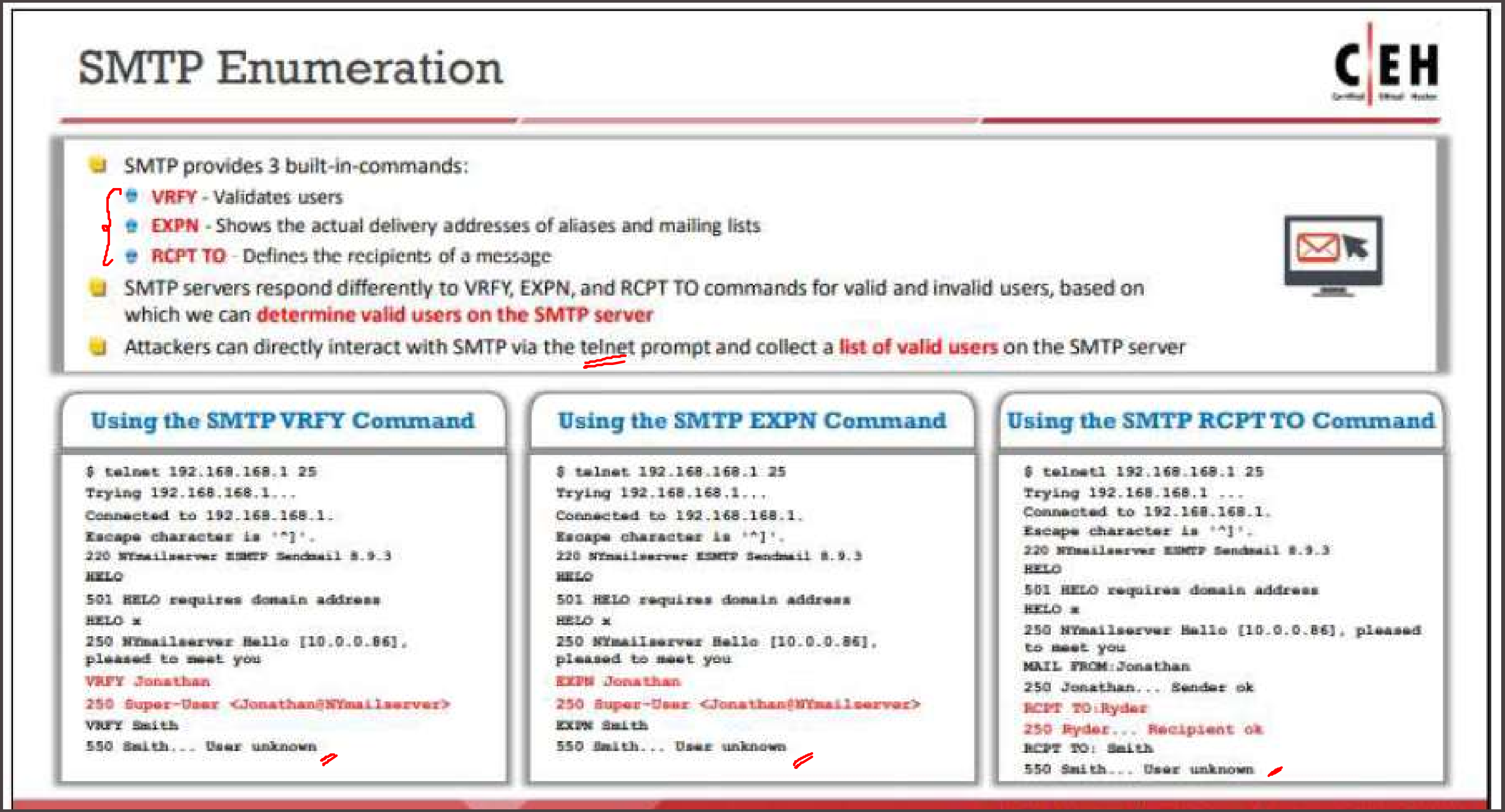
Open NFS Port

https://github.com



Module Flow

1. Enumeration Concepts
2. NetBIOS Enumeration
3. SNMP Enumeration
4. LDAP Enumeration
5. NTP and NFS Enumeration
6. SMTP and DNS Enumeration
7. Other Enumeration Techniques
8. Enumeration Countermeasures



SMTP Enumeration

C|EH Certified Ethical Hacker

- SMTP provides 3 built-in-commands:
  - **VRFY** - Validates users
  - **EXPN** - Shows the actual delivery addresses of aliases and mailing lists
  - **RCPT TO** - Defines the recipients of a message
- SMTP servers respond differently to VRFY, EXPN, and RCPT TO commands for valid and invalid users, based on which we can **determine valid users on the SMTP server**
- Attackers can directly interact with SMTP via the telnet prompt and collect a **list of valid users** on the SMTP server

Using the SMTP VRFY Command

```
$ telnet 192.168.168.1 25
Trying 192.168.168.1...
Connected to 192.168.168.1.
Escape character is '^]'.
220 NYmailserver ESMTP Sendmail 8.9.3
HELO
501 HELO requires domain address
HELO x
250 NYmailserver Hello [10.0.0.86],
pleased to meet you
VRFY Jonathan
250 Super-User <Jonathan@NYmailserver>
VRFY Smith
550 Smith... User unknown
```

Using the SMTP EXPN Command

```
$ telnet 192.168.168.1 25
Trying 192.168.168.1...
Connected to 192.168.168.1.
Escape character is '^]'.
220 NYmailserver ESMTP Sendmail 8.9.3
HELO
501 HELO requires domain address
HELO x
250 NYmailserver Hello [10.0.0.86],
pleased to meet you
EXPN Jonathan
250 Super-User <Jonathan@NYmailserver>
EXPN Smith
550 Smith... User unknown
```

Using the SMTP RCPT TO Command

```
$ telnetl 192.168.168.1 25
Trying 192.168.168.1 ...
Connected to 192.168.168.1.
Escape character is '^]'.
220 NYmailserver ESMTP Sendmail 8.9.3
HELO
501 HELO requires domain address
HELO x
250 NYmailserver Hello [10.0.0.86], pleased
to meet you
MAIL FROM:Jonathan
250 Jonathan... Sender ok
RCPT TO:Ryder
250 Ryder... Recipient ok
RCPT TO: Smith
550 Smith... User unknown
```

SMTP Enumeration Tools

smtp -uses-enum -M VRFy -u ________ username -t ________ ip address

CEH Certified Ethical Hacker

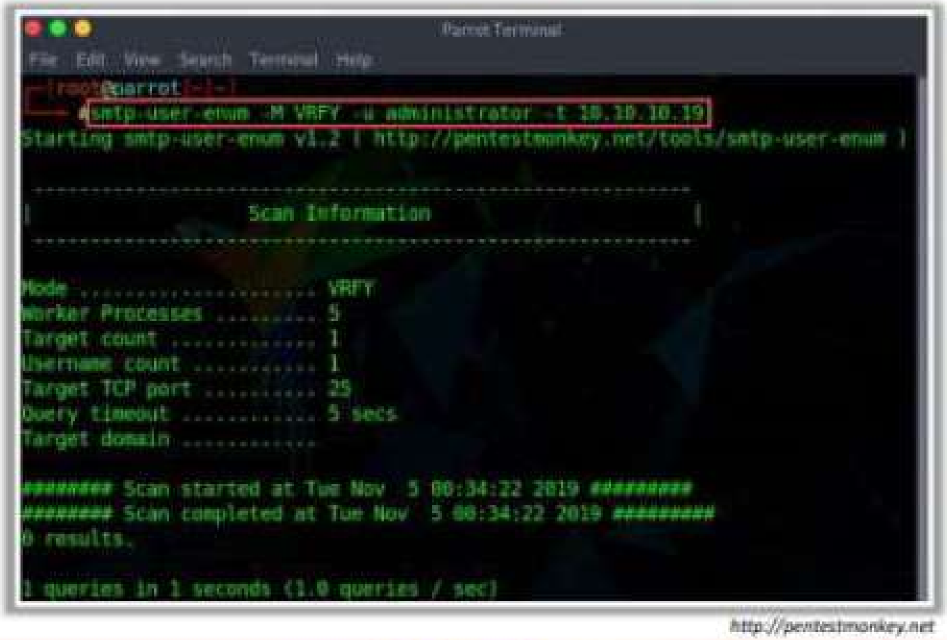
NetScan Tools Pro

- NetScanTools Pro's SMTP Email Generator tool tests the process of sending an email message through an **SMTP server**

smtp-user-enum

- It is a tool for **enumerating OS-level user accounts** on Solaris via the SMTP service (sendmail)
- Enumeration is performed by inspecting the responses to **VRFY**, **EXPN**, and **RCPT TO** commands

https://www.netscantools.com



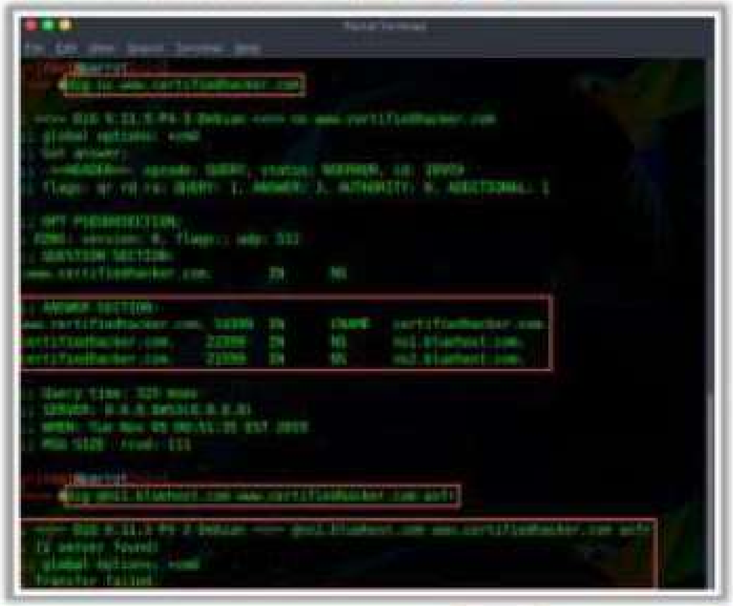
```
┌─[root@parrot]─[~]
└──╼ #smtp-user-enum -M VRFY -u administrator -t 10.10.10.19
Starting smtp-user-enum v1.2 ( http://pentestmonkey.net/tools/smtp-user-enum )

 ----------------------------------------------------------
|                   Scan Information                       |
 ----------------------------------------------------------

Mode ..................... VRFY
Worker Processes ......... 5
Target count ............. 1
Username count ........... 1
Target TCP port .......... 25
Query timeout ............ 5 secs
Target domain ............

######## Scan started at Tue Nov  5 00:34:22 2019 #########
######## Scan completed at Tue Nov  5 00:34:22 2019 #########
0 results.

1 queries in 1 seconds (1.0 queries / sec)
```

http://pentestmonkey.net

DNS Enumeration Using Zone Transfer

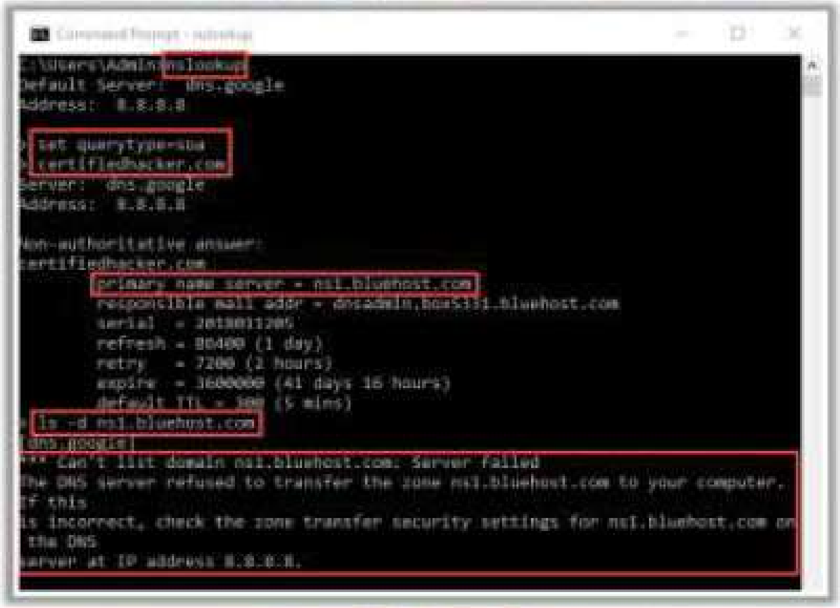
- If the target DNS server allows zone transfers, then attackers use this technique to obtain **DNS server names**, **hostnames**, **machine names**, **usernames**, **IP addresses**, **aliases**, etc. assigned within a target domain
- Attackers perform DNS zone transfer using tools, such as **nslookup**, **dig**, and **DNSRecon**; if DNS transfer setting is enabled on the target name server, it will provide DNS information, or else it will return an error saying it has failed or refuses the zone transfer

Linux DNS zone transfer using dig command



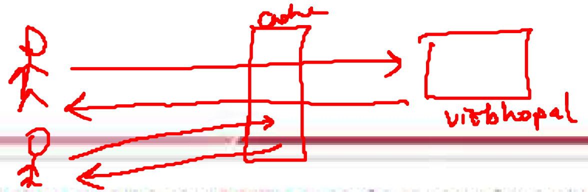
```
$dig ns www.certifiedhacker.com

; <<>> DiG 9.11.5-P4-5-Debian <<>> ns www.certifiedhacker.com
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: [illegible]
;; flags: qr rd ra; QUERY: 1, ANSWER: 3, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 512
;; QUESTION SECTION:
;www.certifiedhacker.com.       IN      NS

;; ANSWER SECTION:
www.certifiedhacker.com. 14399  IN      CNAME   certifiedhacker.com.
certifiedhacker.com.    21599   IN      NS      ns1.bluehost.com.
certifiedhacker.com.    21599   IN      NS      ns2.bluehost.com.

;; Query time: 325 msec
;; SERVER: 8.8.8.8#53(8.8.8.8)
;; WHEN: Tue Nov 05 00:51:35 EST 2019
;; MSG SIZE  rcvd: 111

$dig @ns1.bluehost.com www.certifiedhacker.com axfr

; <<>> DiG 9.11.5-P4-5-Debian <<>> @ns1.bluehost.com www.certifiedhacker.com axfr
; (1 server found)
;; global options: +cmd
; Transfer failed.
```

Windows DNS zone transfer using nslookup command



```
C:\Users\Admin>nslookup
Default Server:  dns.google
Address:  8.8.8.8

> set querytype=soa
> certifiedhacker.com
Server:  dns.google
Address:  8.8.8.8

Non-authoritative answer:
certifiedhacker.com
        primary name server = ns1.bluehost.com
        responsible mail addr = dnsadmin.box5331.bluehost.com
        serial  = 2018011205
        refresh = 86400 (1 day)
        retry   = 7200 (2 hours)
        expire  = 3600000 (41 days 16 hours)
        default TTL = 300 (5 mins)
> ls -d ns1.bluehost.com
[dns.google]
*** Can't list domain ns1.bluehost.com: Server failed
The DNS server refused to transfer the zone ns1.bluehost.com to your computer. If this
is incorrect, check the zone transfer security settings for ns1.bluehost.com on the DNS
server at IP address 8.8.8.8.
```

DNS Cache Snooping



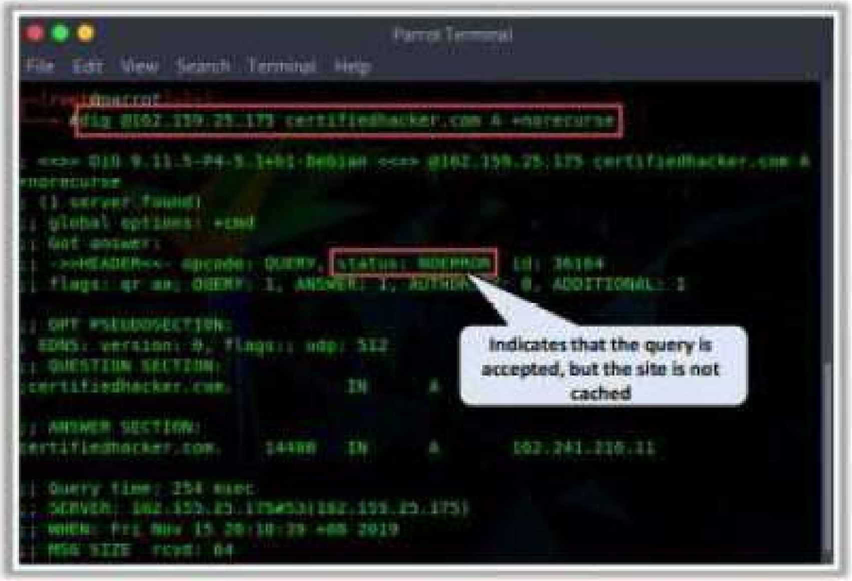
- DNS cache snooping is a **DNS enumeration** technique whereby an **attacker queries** the **DNS server** for a specific cached DNS record

Non-recursive Method

Attackers send a **non-recursive query** by setting the **Recursion Desired** (RD) bit in the query header to zero

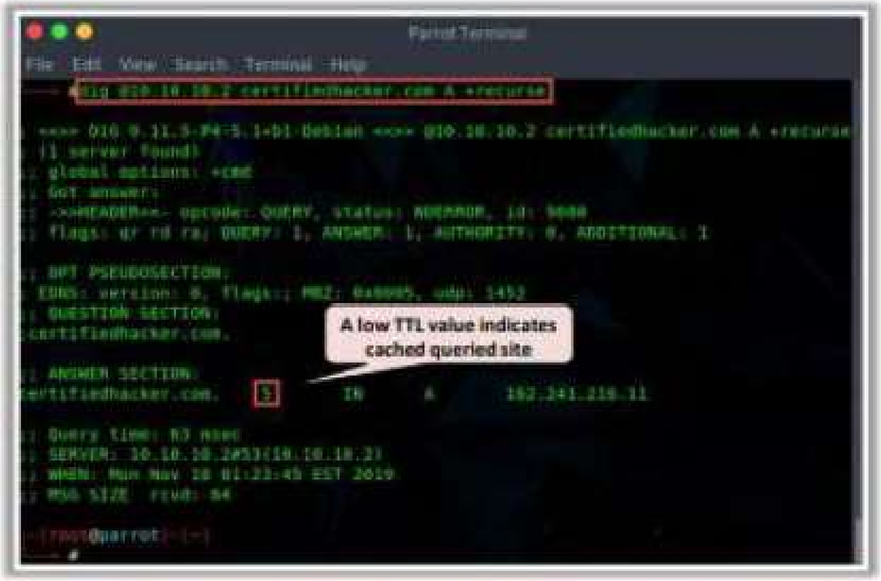
Recursive Method

Attackers send a recursive query to **determine the time** the **DNS record** resides in the cache



```
┌─[root@parrot]─[~]
└──╼ #dig @162.159.25.175 certifiedhacker.com A +norecurse

; <<>> DiG 9.11.5-P4-5.1+b1-Debian <<>> @162.159.25.175 certifiedhacker.com A
+norecurse
; (1 server found)
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 36184
;; flags: qr aa; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 512
;; QUESTION SECTION:
;certifiedhacker.com.           IN      A

;; ANSWER SECTION:
certifiedhacker.com.    14400   IN      A       162.241.216.11

;; Query time: 254 msec
;; SERVER: 162.159.25.175#53(162.159.25.175)
;; WHEN: Fri Nov 15 20:10:29 +08 2019
;; MSG SIZE  rcvd: 64
```

Indicates that the query is accepted, but the site is not cached



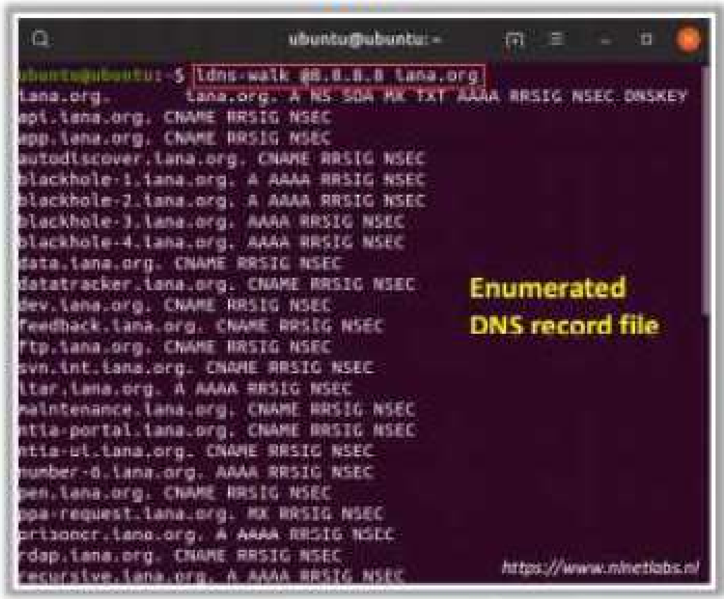
```
└──╼ #dig @10.10.10.2 certifiedhacker.com A +recurse

; <<>> DiG 9.11.5-P4-5.1+b1-Debian <<>> @10.10.10.2 certifiedhacker.com A +recurse
; (1 server found)
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 5000
;; flags: qr rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; MBZ: 0x0005, udp: 1452
;; QUESTION SECTION:
;certifiedhacker.com.

;; ANSWER SECTION:
certifiedhacker.com.    5       IN      A       162.241.216.11

;; Query time: 83 msec
;; SERVER: 10.10.10.2#53(10.10.10.2)
;; WHEN: Mon Nov 18 01:23:45 EST 2019
;; MSG SIZE  rcvd: 64

┌─[root@parrot]─[~]
└──╼ #
```

A low TTL value indicates cached queried site

DNSSEC Zone Walking

- DNSSEC zone walking is a DNS enumeration technique where an attacker attempts to **obtain internal records of the DNS server** if the DNS zone is not properly configured
- Attackers use tools, such as **LDNS** and **DNSRecon**, to exploit this vulnerability and **obtain the network information** of a target domain and further launch Internet-based attacks

LDNS

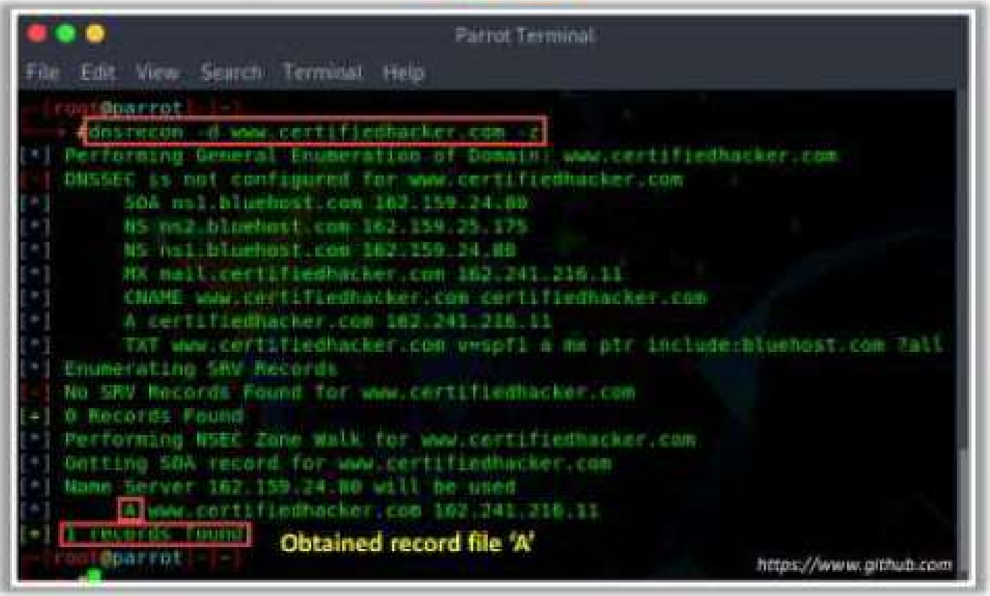


```
ubuntu@ubuntu:~$ ldns-walk @8.8.8.8 iana.org
iana.org.       iana.org. A NS SOA MX TXT AAAA RRSIG NSEC DNSKEY
api.iana.org. CNAME RRSIG NSEC
app.iana.org. CNAME RRSIG NSEC
autodiscover.iana.org. CNAME RRSIG NSEC
blackhole-1.iana.org. A AAAA RRSIG NSEC
blackhole-2.iana.org. A AAAA RRSIG NSEC
blackhole-3.iana.org. AAAA RRSIG NSEC
blackhole-4.iana.org. AAAA RRSIG NSEC
data.iana.org. CNAME RRSIG NSEC
datatracker.iana.org. CNAME RRSIG NSEC
dev.iana.org. CNAME RRSIG NSEC
feedback.iana.org. CNAME RRSIG NSEC
ftp.iana.org. CNAME RRSIG NSEC
svn.int.iana.org. CNAME RRSIG NSEC
itar.iana.org. A AAAA RRSIG NSEC
maintenance.iana.org. CNAME RRSIG NSEC
ntia-portal.iana.org. CNAME RRSIG NSEC
ntia-ui.iana.org. CNAME RRSIG NSEC
number-0.iana.org. AAAA RRSIG NSEC
pen.iana.org. CNAME RRSIG NSEC
ppa-request.iana.org. MX RRSIG NSEC
prisoncr.iana.org. A AAAA RRSIG NSEC
rdap.iana.org. CNAME RRSIG NSEC
recursive.iana.org. A AAAA RRSIG NSEC
```

Enumerated DNS record file

https://www.nlnetlabs.nl

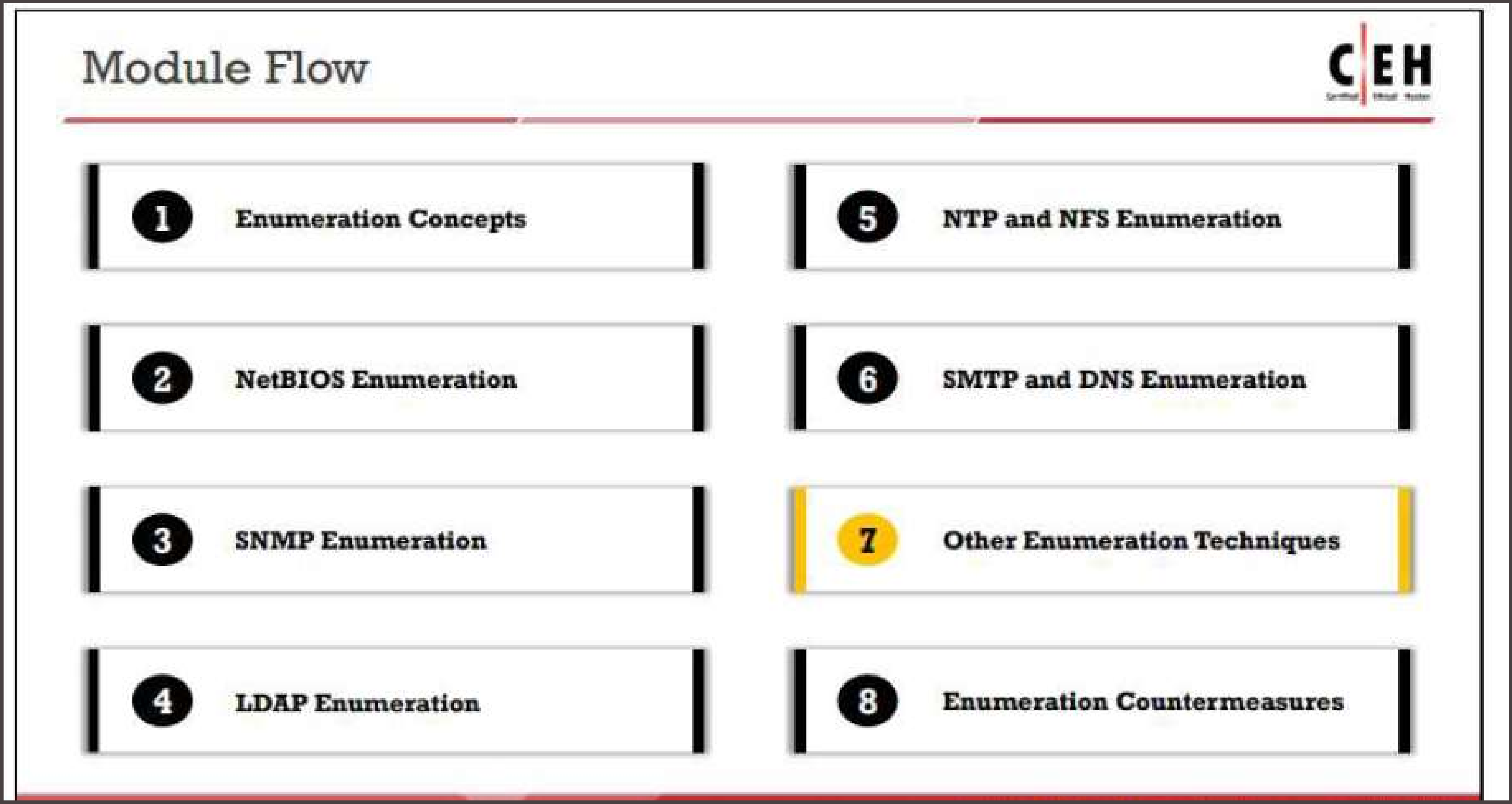
DNSRecon



```
┌─[root@parrot]─[~]
└──╼ #dnsrecon -d www.certifiedhacker.com -z
[*] Performing General Enumeration of Domain: www.certifiedhacker.com
[-] DNSSEC is not configured for www.certifiedhacker.com
[*]      SOA ns1.bluehost.com 162.159.24.80
[*]      NS ns2.bluehost.com 162.159.25.175
[*]      NS ns1.bluehost.com 162.159.24.80
[*]      MX mail.certifiedhacker.com 162.241.216.11
[*]      CNAME www.certifiedhacker.com certifiedhacker.com
[*]      A certifiedhacker.com 162.241.216.11
[*]      TXT www.certifiedhacker.com v=spf1 a mx ptr include:bluehost.com ?all
[*] Enumerating SRV Records
[-] No SRV Records Found for www.certifiedhacker.com
[+] 0 Records Found
[*] Performing NSEC Zone Walk for www.certifiedhacker.com
[*] Getting SOA record for www.certifiedhacker.com
[*] Name Server 162.159.24.80 will be used
[*]     A www.certifiedhacker.com 162.241.216.11
[+] 1 records found
┌─[root@parrot]─[~]
```

Obtained record file 'A'

https://www.github.com



Module Flow

1. Enumeration Concepts
2. NetBIOS Enumeration
3. SNMP Enumeration
4. LDAP Enumeration
5. NTP and NFS Enumeration
6. SMTP and DNS Enumeration
7. Other Enumeration Techniques
8. Enumeration Countermeasures

IPsec Enumeration

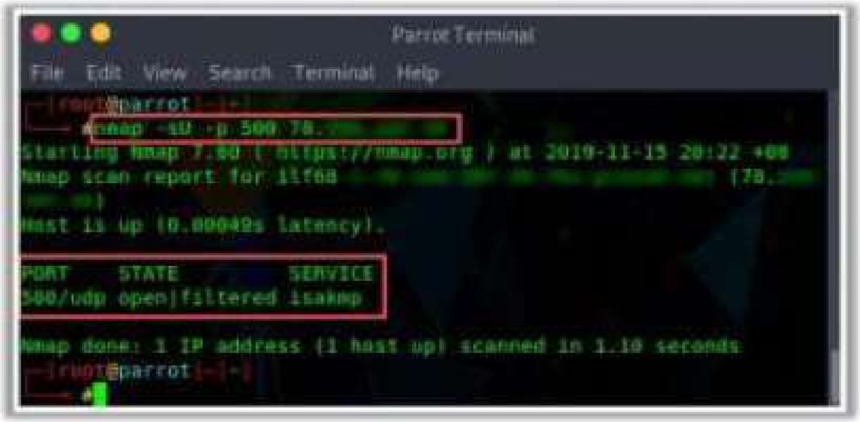
nmap -sU -p 500 ______
ipaddress

- IPsec uses Encapsulation Security Payload (ESP), Authentication Header (AH), and Internet Key Exchange (IKE) to secure **communication between virtual private network (VPN) end points**

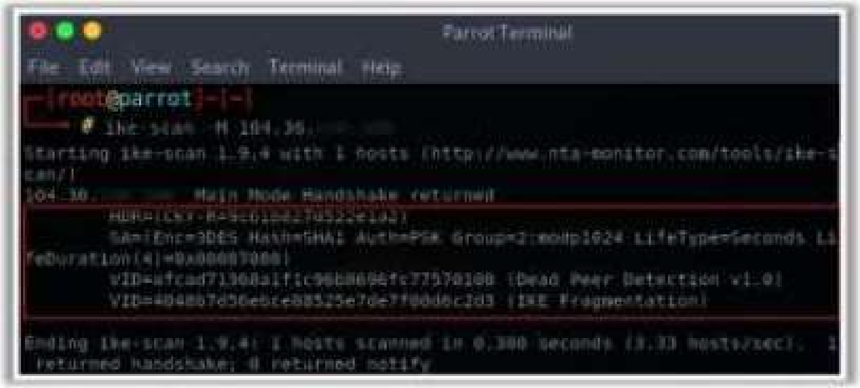
- Most IPsec based **VPNs use Internet Security Association and Key Management Protocol (ISAKMP)**, a part of IKE, to establish, negotiate, modify, and delete Security Associations (SA) and cryptographic keys in a VPN environment

- A simple **scanning for ISAKMP at UDP port 500** can indicate the presence of a VPN gateway

- Attackers can probe further using a tool, such as **ike-scan**, to enumerate sensitive information, including encryption and hashing algorithm, authentication type, key distribution algorithm, and SA LifeDuration



```
┌─[root@parrot]─[~]
└──╼ #nmap -sU -p 500 78.[redacted]
Starting Nmap 7.80 ( https://nmap.org ) at 2019-11-15 20:22 +08
Nmap scan report for ilf68[redacted] (78.[redacted]
[redacted])
Host is up (0.00049s latency).

PORT    STATE         SERVICE
500/udp open|filtered isakmp

Nmap done: 1 IP address (1 host up) scanned in 1.10 seconds
┌─[root@parrot]─[~]
└──╼ #
```



```
┌─[root@parrot]─[~]
└──╼ # ike-scan -M 104.36.[redacted]
Starting ike-scan 1.9.4 with 1 hosts (http://www.nta-monitor.com/tools/ike-s
can/)
104.36.[redacted]    Main Mode Handshake returned
        HDR=(CKY-R=9cb10d27d522e1a2)
        SA=(Enc=3DES Hash=SHA1 Auth=PSK Group=2:modp1024 LifeType=Seconds Li
feDuration(4)=0x00007080)
        VID=afcad71368a1f1c96b8696fc77570100 (Dead Peer Detection v1.0)
        VID=4048b7d56ebce88525e7de7f00d6c2d3 (IKE Fragmentation)

Ending ike-scan 1.9.4: 1 hosts scanned in 0.300 seconds (3.33 hosts/sec).  1
 returned handshake; 0 returned notify
```

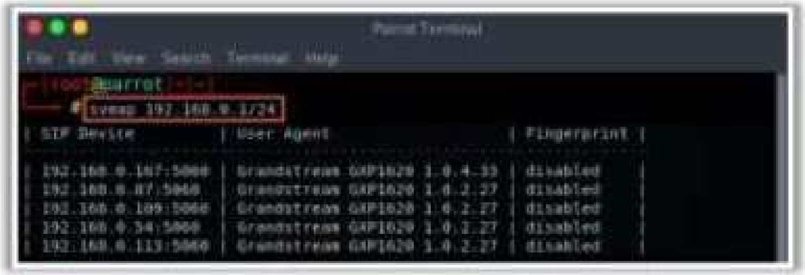
VoIP Enumeration

- VoIP uses **Session Initiation Protocol (SIP)** protocol to enable voice and video calls over an IP network

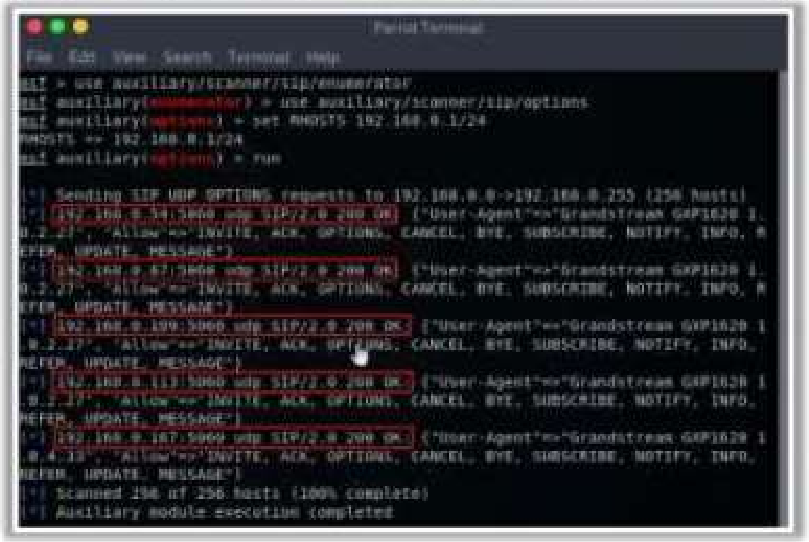
- SIP service generally uses **UDP/TCP ports 2000**, 2001, 5050, and 5061

- VoIP enumeration provides sensitive information, such as **VoIP gateway/servers**, **IP-PBX systems**, **client software** (softphones)/VoIP phones, User-agent IP addresses, and **user extensions**

- This information can be used to launch various VoIP attacks, such as **Denial-of-Service (DoS)**, Session Hijacking, **Caller ID spoofing**, **Eavesdropping**, Spamming over Internet Telephony (SPIT), and **VoIP phishing** (Vishing)



```
┌─[root@parrot]─[~]
└──╼ #svmap 192.168.0.1/24
| SIP Device            | User Agent                   | Fingerprint |
-----------------------------------------------------------------------
| 192.168.0.167:5060    | Grandstream GXP1620 1.0.4.33 | disabled    |
| 192.168.0.87:5060     | Grandstream GXP1620 1.0.2.27 | disabled    |
| 192.168.0.109:5060    | Grandstream GXP1620 1.0.2.27 | disabled    |
| 192.168.0.34:5060     | Grandstream GXP1620 1.0.2.27 | disabled    |
| 192.168.0.113:5060    | Grandstream GXP1620 1.0.2.27 | disabled    |
```

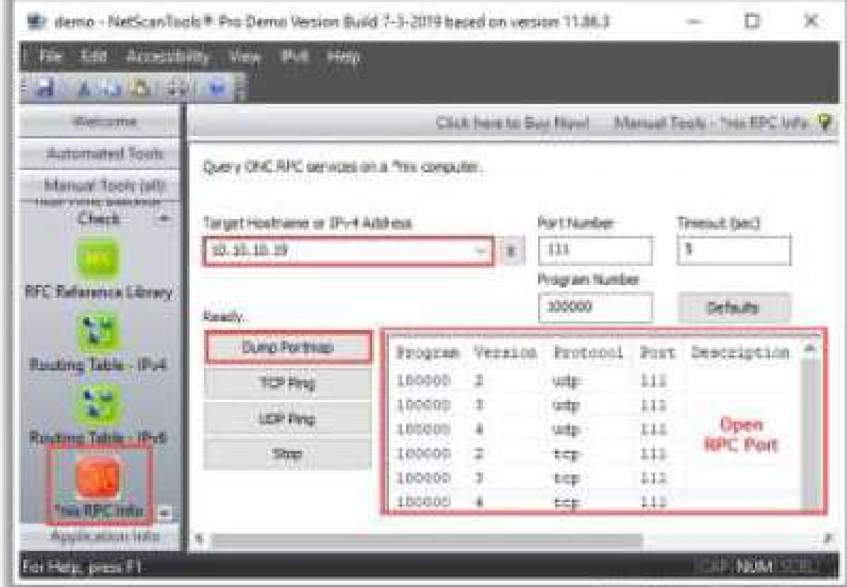


```
msf > use auxiliary/scanner/sip/enumerator
msf auxiliary(enumerator) > use auxiliary/scanner/sip/options
msf auxiliary(options) > set RHOSTS 192.168.0.1/24
RHOSTS => 192.168.0.1/24
msf auxiliary(options) > run

[*] Sending SIP UDP OPTIONS requests to 192.168.0.0->192.168.0.255 (256 hosts)
[*] 192.168.0.34:5060 udp SIP/2.0 200 OK: {"User-Agent"=>"Grandstream GXP1620 1.
0.2.27", "Allow"=>"INVITE, ACK, OPTIONS, CANCEL, BYE, SUBSCRIBE, NOTIFY, INFO, R
EFER, UPDATE, MESSAGE"}
[*] 192.168.0.87:5060 udp SIP/2.0 200 OK: {"User-Agent"=>"Grandstream GXP1620 1.
0.2.27", "Allow"=>"INVITE, ACK, OPTIONS, CANCEL, BYE, SUBSCRIBE, NOTIFY, INFO, R
EFER, UPDATE, MESSAGE"}
[*] 192.168.0.109:5060 udp SIP/2.0 200 OK: {"User-Agent"=>"Grandstream GXP1620 1
.0.2.27", "Allow"=>"INVITE, ACK, OPTIONS, CANCEL, BYE, SUBSCRIBE, NOTIFY, INFO,
REFER, UPDATE, MESSAGE"}
[*] 192.168.0.113:5060 udp SIP/2.0 200 OK: {"User-Agent"=>"Grandstream GXP1620 1
.0.2.27", "Allow"=>"INVITE, ACK, OPTIONS, CANCEL, BYE, SUBSCRIBE, NOTIFY, INFO,
REFER, UPDATE, MESSAGE"}
[*] 192.168.0.167:5060 udp SIP/2.0 200 OK: {"User-Agent"=>"Grandstream GXP1620 1
.0.4.33", "Allow"=>"INVITE, ACK, OPTIONS, CANCEL, BYE, SUBSCRIBE, NOTIFY, INFO,
REFER, UPDATE, MESSAGE"}
[*] Scanned 256 of 256 hosts (100% complete)
[*] Auxiliary module execution completed
```

RPC Enumeration

nmap -T4 -A ______ ipaddr

- Remote Procedure Call (RPC) allows clients and servers to communicate in **distributed client/server programs**
- Enumerating RPC endpoints enables attackers to **identify any vulnerable services** on these service ports



Query ONC RPC services on a *nix computer.

Target Hostname or IPv4 Address: 10.10.10.19

Port Number: 111

Timeout (sec): 5

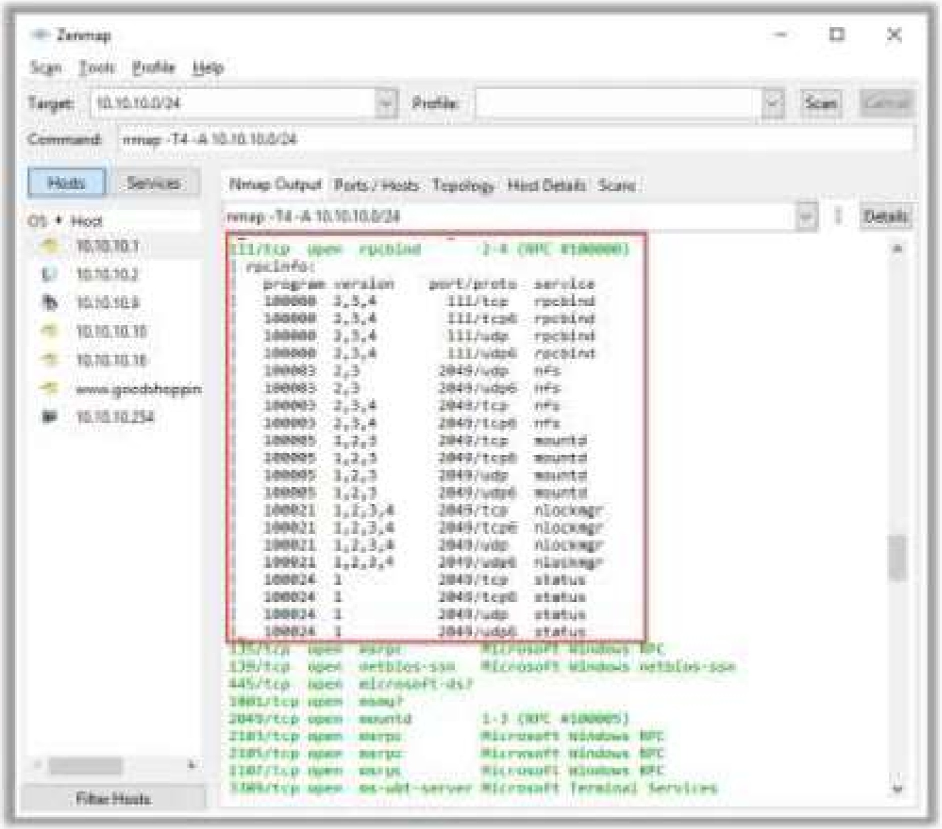
Program Number: 100000

Ready.

```
Program  Version  Protocol  Port  Description
100000   2        udp       111
100000   3        udp       111
100000   4        udp       111
100000   2        tcp       111
100000   3        tcp       111
100000   4        tcp       111
```

Open RPC Port

https://www.netscantools.com



Target: 10.10.10.0/24

Command: nmap -T4 -A 10.10.10.0/24

```
nmap -T4 -A 10.10.10.0/24
111/tcp  open  rpcbind       2-4 (RPC #100000)
| rpcinfo:
|   program version    port/proto  service
|   100000  2,3,4        111/tcp   rpcbind
|   100000  2,3,4        111/tcp6  rpcbind
|   100000  2,3,4        111/udp   rpcbind
|   100000  2,3,4        111/udp6  rpcbind
|   100003  2,3         2049/udp   nfs
|   100003  2,3         2049/udp6  nfs
|   100003  2,3,4       2049/tcp   nfs
|   100003  2,3,4       2049/tcp6  nfs
|   100005  1,2,3       2049/tcp   mountd
|   100005  1,2,3       2049/tcp6  mountd
|   100005  1,2,3       2049/udp   mountd
|   100005  1,2,3       2049/udp6  mountd
|   100021  1,2,3,4     2049/tcp   nlockmgr
|   100021  1,2,3,4     2049/tcp6  nlockmgr
|   100021  1,2,3,4     2049/udp   nlockmgr
|   100021  1,2,3,4     2049/udp6  nlockmgr
|   100024  1           2049/tcp   status
|   100024  1           2049/tcp6  status
|   100024  1           2049/udp   status
|_  100024  1           2049/udp6  status
135/tcp  open  msrpc         Microsoft Windows RPC
139/tcp  open  netbios-ssn   Microsoft Windows netbios-ssn
445/tcp  open  microsoft-ds?
1801/tcp open  msmq?
2049/tcp open  mountd        1-3 (RPC #100005)
2103/tcp open  msrpc         Microsoft Windows RPC
2105/tcp open  msrpc         Microsoft Windows RPC
2107/tcp open  msrpc         Microsoft Windows RPC
3389/tcp open  ms-wbt-server Microsoft Terminal Services
```

Unix/Linux User Enumeration

rusers

- Displays a list of users who are logged on to remote machines or machines on local network

Syntax: `/usr/bin/rusers [-a] [-l] [-u| -h| -i] [Host ...]`

rwho

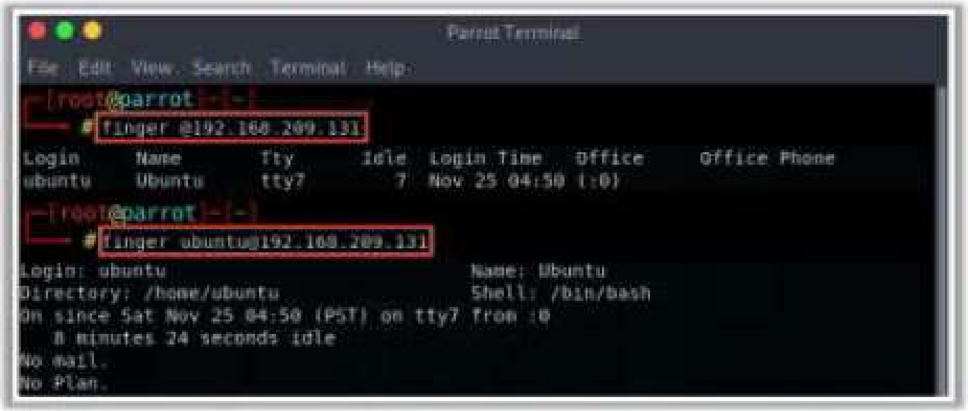
- Displays a list of users who are logged on to hosts on the local network

Syntax: `rwho [ -a]`

finger

- Displays information about system users, such as login name, real name, terminal name, idle time, login time, office location, and office phone numbers

Syntax: `finger [-l] [-m] [-p] [-s] [user ...] [user@host ... ]`



```
┌─[root@parrot]─[~]
└──╼ #finger @192.168.209.131
Login     Name      Tty      Idle  Login Time   Office     Office Phone
ubuntu    Ubuntu    tty7        7  Nov 25 04:50 (:0)
┌─[root@parrot]─[~]
└──╼ #finger ubuntu@192.168.209.131
Login: ubuntu                           Name: Ubuntu
Directory: /home/ubuntu                 Shell: /bin/bash
On since Sat Nov 25 04:50 (PST) on tty7 from :0
   8 minutes 24 seconds idle
No mail.
No Plan.
```

Telnet and SMB Enumeration

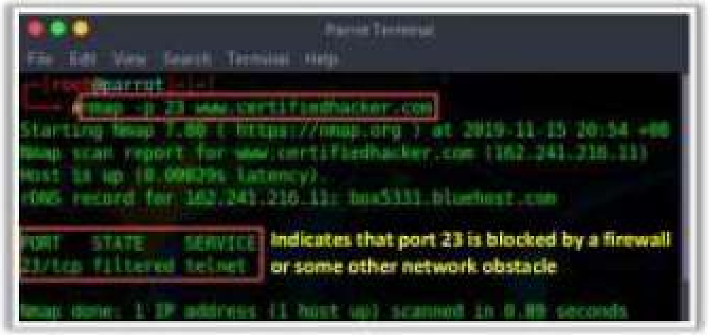
nmap -p 23 ____

nmap -p 445 -A ____ ipaddr

Telnet Enumeration

- If the Telnet port is found open, attackers can **access shared information**, including the hardware and software information of the target
- Telnet enumeration enables attackers to **exploit identified vulnerabilities** and perform brute-force attacks to gain unauthorized access to the target and launch further attacks

23



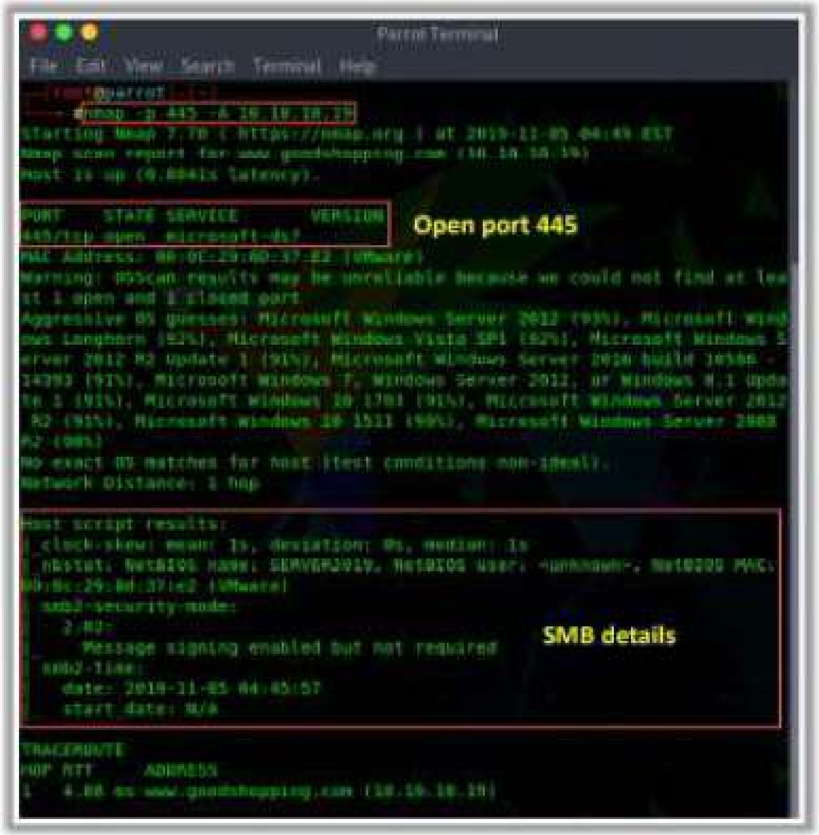
```
┌─[root@parrot]─[~]
└──╼ #nmap -p 23 www.certifiedhacker.com
Starting Nmap 7.80 ( https://nmap.org ) at 2019-11-15 20:34 +00
Nmap scan report for www.certifiedhacker.com (162.241.216.11)
Host is up (0.00029s latency).
rDNS record for 162.241.216.11: box5331.bluehost.com

PORT   STATE    SERVICE
23/tcp filtered telnet

Nmap done: 1 IP address (1 host up) scanned in 0.89 seconds
```

Indicates that port 23 is blocked by a firewall or some other network obstacle

445

SMB Enumeration

- Attackers use SMB enumeration tools, such as **Nmap**, **SMBMap**, **enum4linux**, and **nullinux**, to perform a directed scan on the SMB service running on port 445
- SMB enumeration helps attackers to perform **OS banner grabbing** on the target



```
┌─[root@parrot]─[~]
└──╼ #nmap -p 445 -A 10.10.10.19
Starting Nmap 7.70 ( https://nmap.org ) at 2019-11-05 04:45 EST
Nmap scan report for www.goodshopping.com (10.10.10.19)
Host is up (0.0041s latency).

PORT    STATE SERVICE       VERSION
445/tcp open  microsoft-ds?
MAC Address: 00:0C:29:0D:37:E2 (VMware)
Warning: OSScan results may be unreliable because we could not find at lea
st 1 open and 1 closed port
Aggressive OS guesses: Microsoft Windows Server 2012 (93%), Microsoft Wind
ows Longhorn (92%), Microsoft Windows Vista SP1 (92%), Microsoft Windows S
erver 2012 R2 Update 1 (91%), Microsoft Windows Server 2016 build 10586 -
14393 (91%), Microsoft Windows 7, Windows Server 2012, or Windows 8.1 Upda
te 1 (91%), Microsoft Windows 10 1703 (91%), Microsoft Windows Server 2012
 R2 (91%), Microsoft Windows 10 1511 (90%), Microsoft Windows Server 2008
R2 (90%)
No exact OS matches for host (test conditions non-ideal).
Network Distance: 1 hop

Host script results:
|_clock-skew: mean: 1s, deviation: 0s, median: 1s
|_nbstat: NetBIOS name: SERVER2019, NetBIOS user: <unknown>, NetBIOS MAC:
00:0c:29:0d:37:e2 (VMware)
| smb2-security-mode:
|   2.02:
|_    Message signing enabled but not required
| smb2-time:
|   date: 2019-11-05 04:45:57
|_  start_date: N/A

TRACEROUTE
HOP RTT     ADDRESS
1   4.00 ms www.goodshopping.com (10.10.10.19)
```

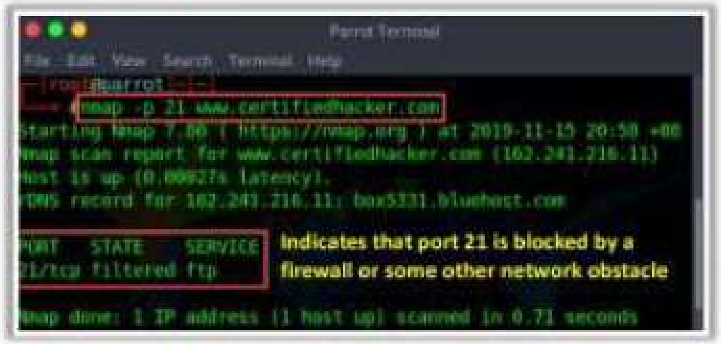
Open port 445

SMB details

FTP and TFTP Enumeration

FTP Enumeration

- FTP transfers data in plain text between the sender and receiver, which can lead to critical information, such as **usernames and passwords**, **being exposed to attackers**
- Attackers use **Nmap** to scan and enumerate open port 21 by running FTP services and further use the information to launch various attacks, such as **FTP bounce**, **FTP brute force**, and **packet sniffing**

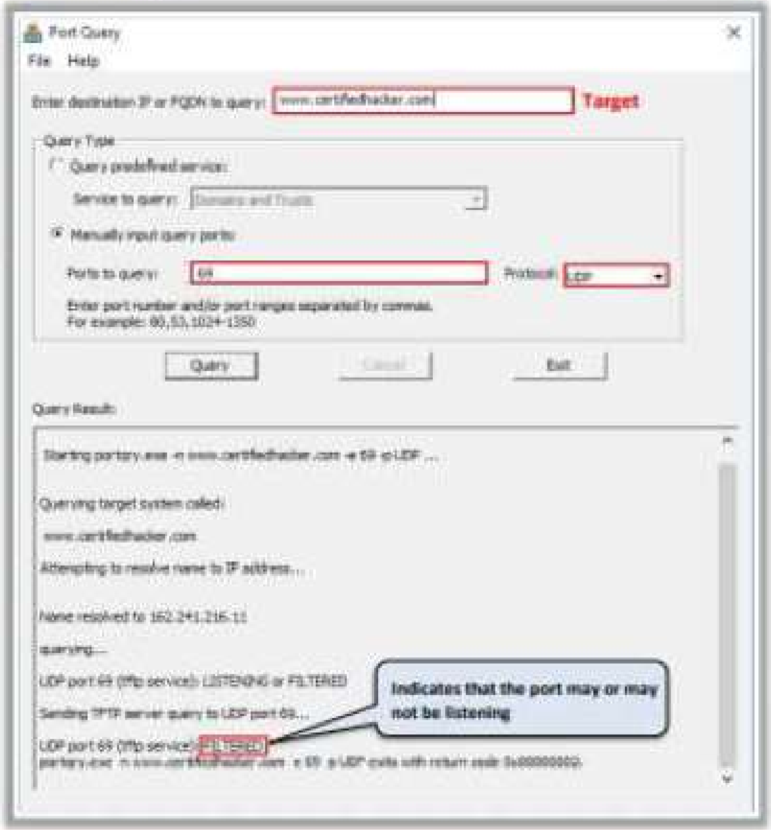


```
┌─[root@parrot]─[~]
└──╼ #nmap -p 21 www.certifiedhacker.com
Starting Nmap 7.80 ( https://nmap.org ) at 2019-11-15 20:50 +08
Nmap scan report for www.certifiedhacker.com (162.241.216.11)
Host is up (0.00027s latency).
rDNS record for 162.241.216.11: box5331.bluehost.com

PORT    STATE    SERVICE
21/tcp filtered ftp

Nmap done: 1 IP address (1 host up) scanned in 0.71 seconds
```

Indicates that port 21 is blocked by a firewall or some other network obstacle

TFTP Enumeration

- Attackers perform TFTP enumeration using tools, such as **PortQry** and **Nmap**, to extract information, such as **running TFTP services** and files stored on the remote server
- Using this information, attackers can gain unauthorized access to the target system, steal important files, and upload malicious script to launch further attacks



Enter destination IP or FQDN to query: www.certifiedhacker.com

Target

○ Query predefined service:

Service to query: Domains and Trusts

● Manually input query ports:

Ports to query: 69

Protocol: UDP

Enter port number and/or port ranges separated by commas.
For example: 80,53,1024-1350

Query Result:

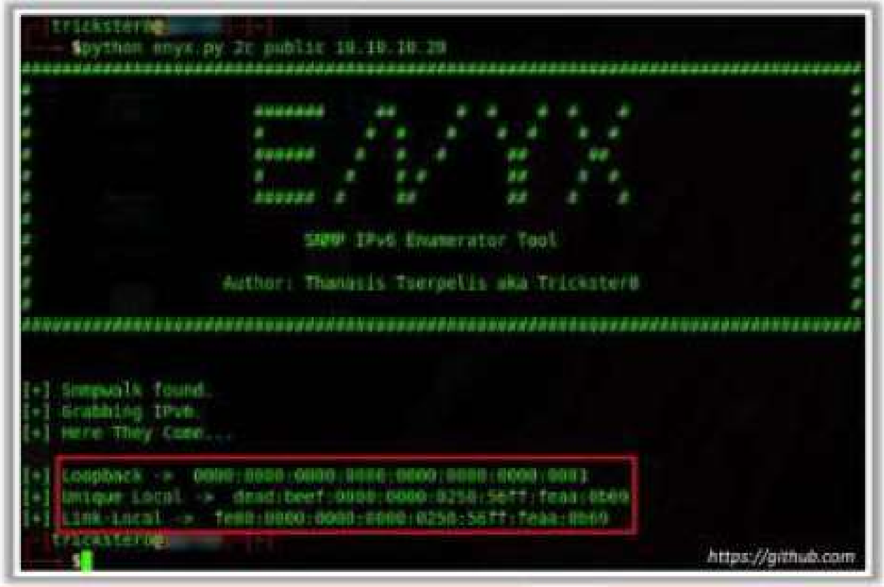
```
Starting portqry.exe -n www.certifiedhacker.com -e 69 -p UDP ...

Querying target system called:

 www.certifiedhacker.com

Attempting to resolve name to IP address...

Name resolved to 162.241.216.11

querying...

UDP port 69 (tftp service): LISTENING or FILTERED

Sending TFTP server query to UDP port 69...

UDP port 69 (tftp service): FILTERED
portqry.exe -n www.certifiedhacker.com -e 69 -p UDP exits with return code 0x00000002.
```

Indicates that the port may or may not be listening

IPv6 Enumeration

- IPv6 is an addressing protocol that **provides identification to computer systems**, including their location information and further assists in routing traffic from one system to the other across the network
- Attackers perform IPv6 enumeration using various tools, such as Enyx and IPv6 Hackit, on target hosts to **obtain their IPv6 addresses** and further scan the enumerated IP addresses to detect various security problems

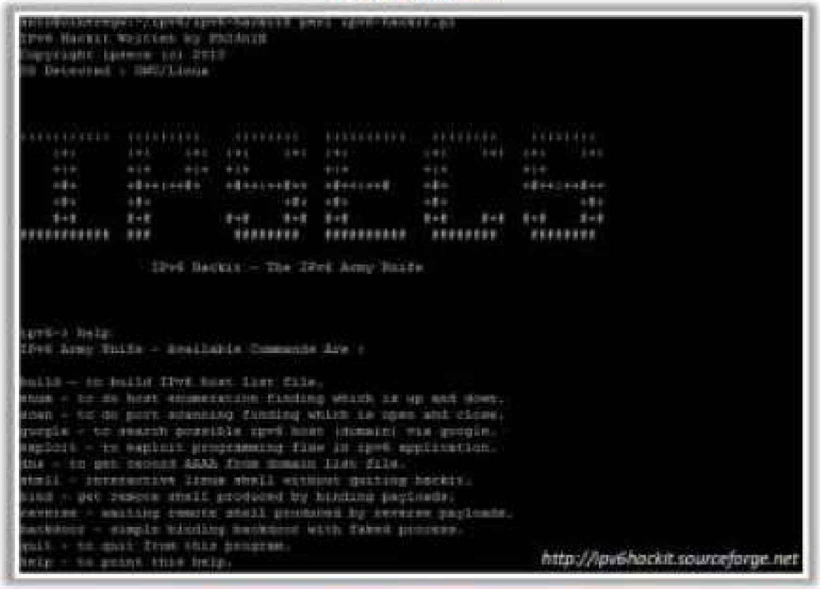
Enyx



```
┌─trickster@[redacted]
└──╼ $python enyx.py 2c public 10.10.10.20
################################################################################
#                                                                              #
#                                                                              #
#                       SNMP IPv6 Enumerator Tool                              #
#                                                                              #
#                 Author: Thanasis Tserpelis aka Trickster0                    #
#                                                                              #
################################################################################


[+] Snmpwalk found.
[+] Grabbing IPv6.
[+] Here They Come...

[+] Loopback ->  0000:0000:0000:0000:0000:0000:0000:0001
[+] Unique Local ->  dead:beef:0000:0000:0250:56ff:feaa:0b69
[+] Link Local ->  fe80:0000:0000:0000:0250:56ff:feaa:0b69
┌─trickster@[redacted]
└──╼ $
```

https://github.com

IPv6 Hackit



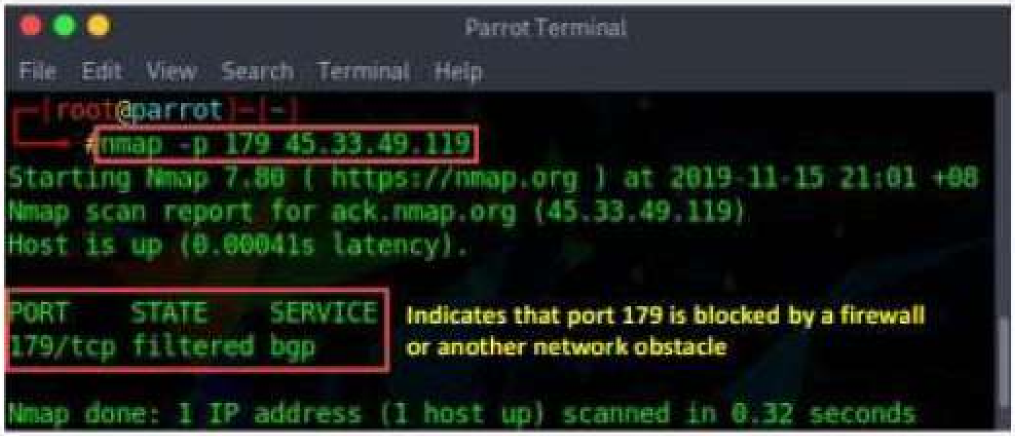
```
[illegible]
IPv6 Hackit Written by [illegible]
Copyright ipsecs (c) 2012
OS Detected : GNU/Linux


                IPv6 Hackit - The IPv6 Army Knife


ipv6-> help
IPv6 Army Knife - Available Commands Are :

build -- to build IPv6 host list file.
enum - to do host enumeration finding which is up and down.
scan - to do port scanning finding which is open and close.
google - to search possible ipv6 host (domain) via google.
exploit - to exploit programming flaw in ipv6 application.
dns - to get record AAAA from domain list file.
shell - interactive linux shell without quiting hackit.
bind - get remote shell produced by binding payloads.
reverse - waiting remote shell produced by reverse payloads.
backdoor - simple binding backdoor with faked process.
quit - to quit from this program.
help - to print this help.
```

http://ipv6hackit.sourceforge.net

BGP Enumeration

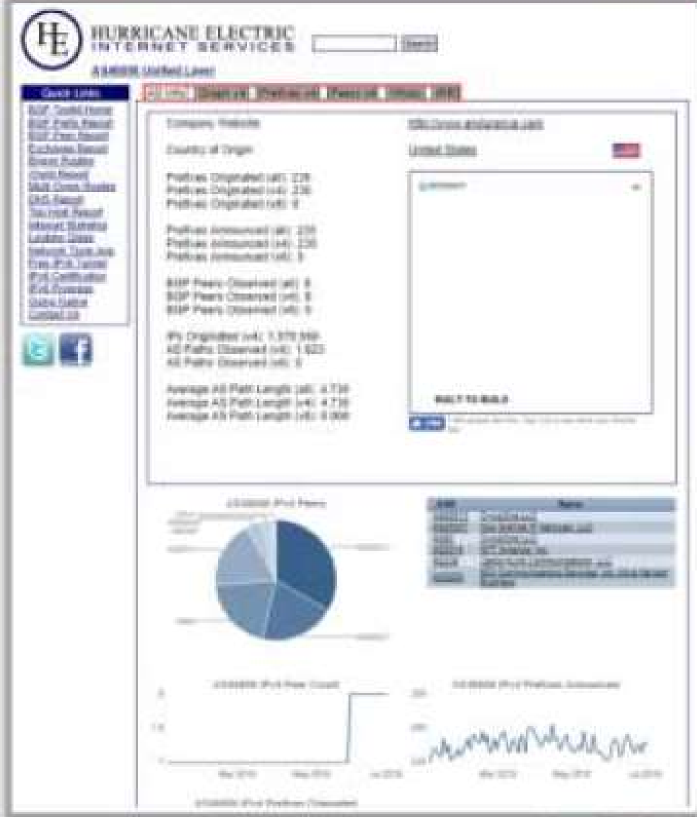
179

- Border Gateway Protocol (BGP) is a routing protocol used to **exchange routing and reachability information** between different autonomous systems (AS) present on the Internet
- Attackers perform BGP enumeration using tools, such as **Nmap** and **BGP Toolkit**, to discover the IPv4 prefixes announced by the AS number and routing path followed by the target
- Attackers use this information to launch various attacks, such as **man-in-the-middle attack**, **BGP hijacking attack**, and **DoS attack** against the target

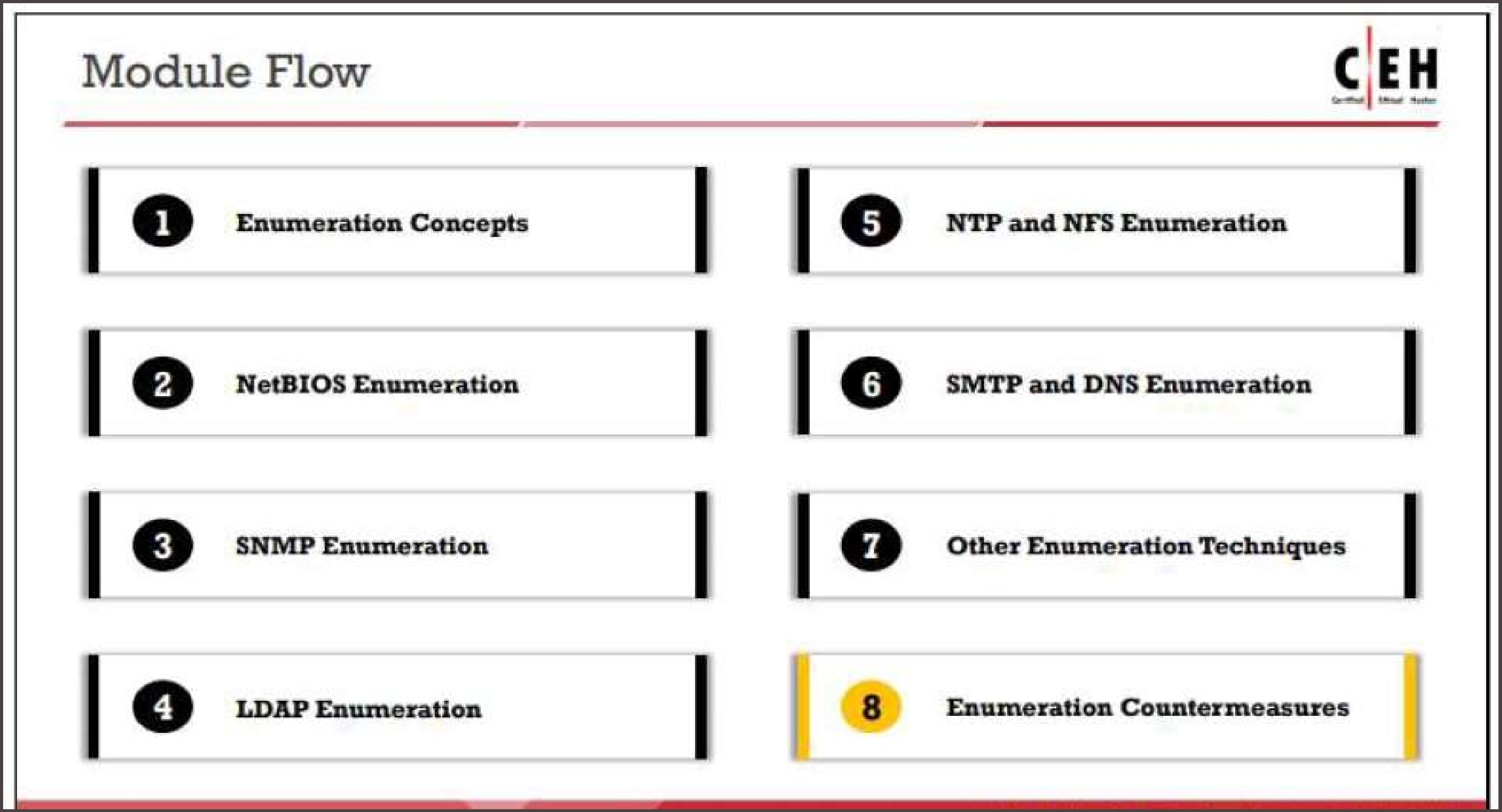


```
┌─[root@parrot]─[~]
└──╼ #nmap -p 179 45.33.49.119
Starting Nmap 7.80 ( https://nmap.org ) at 2019-11-15 21:01 +08
Nmap scan report for ack.nmap.org (45.33.49.119)
Host is up (0.00041s latency).

PORT     STATE    SERVICE
179/tcp filtered bgp

Nmap done: 1 IP address (1 host up) scanned in 0.32 seconds
```

Indicates that port 179 is blocked by a firewall or another network obstacle

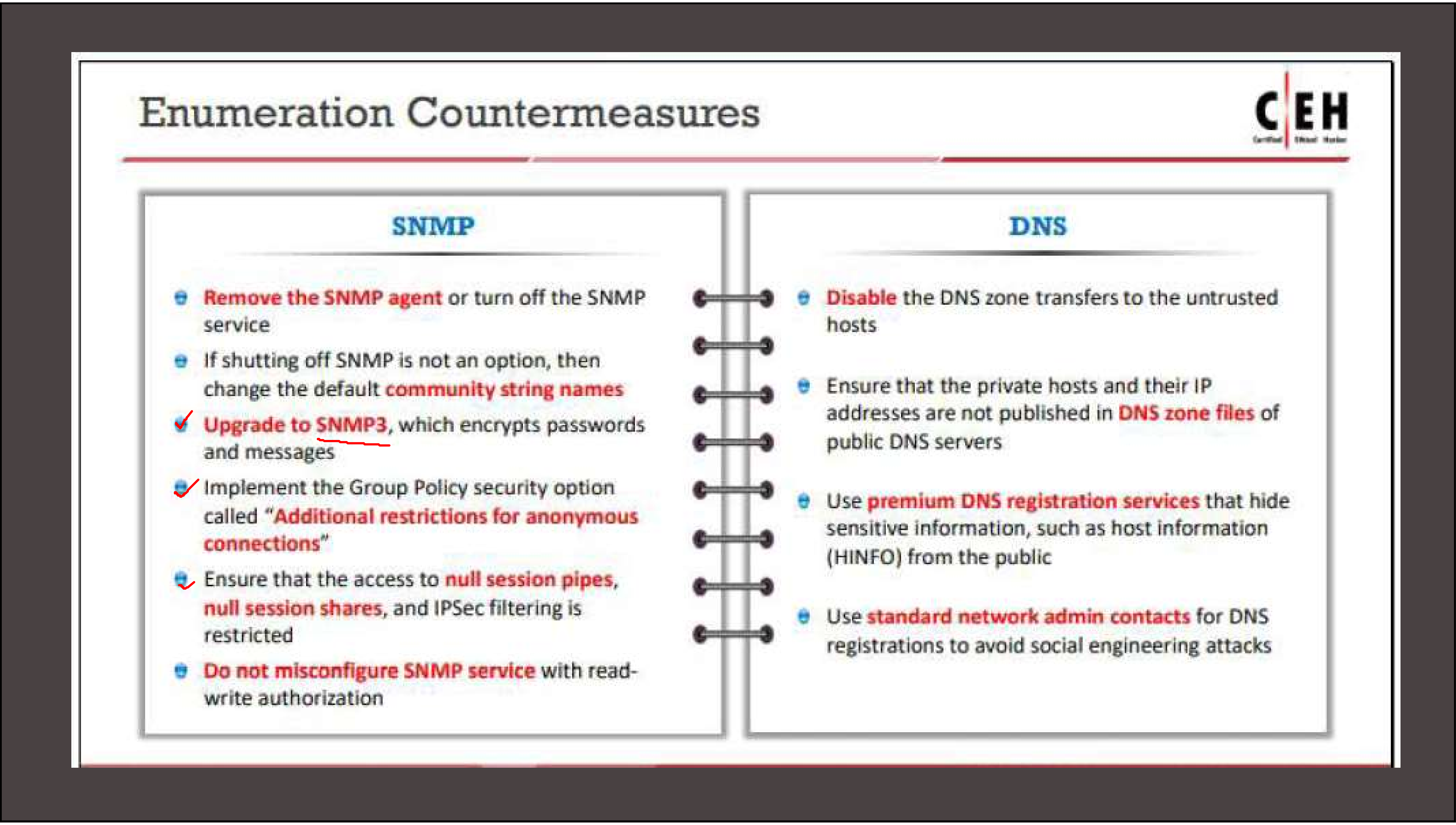


https://bgp.he.net



Module Flow

1. Enumeration Concepts
2. NetBIOS Enumeration
3. SNMP Enumeration
4. LDAP Enumeration
5. NTP and NFS Enumeration
6. SMTP and DNS Enumeration
7. Other Enumeration Techniques
8. Enumeration Countermeasures



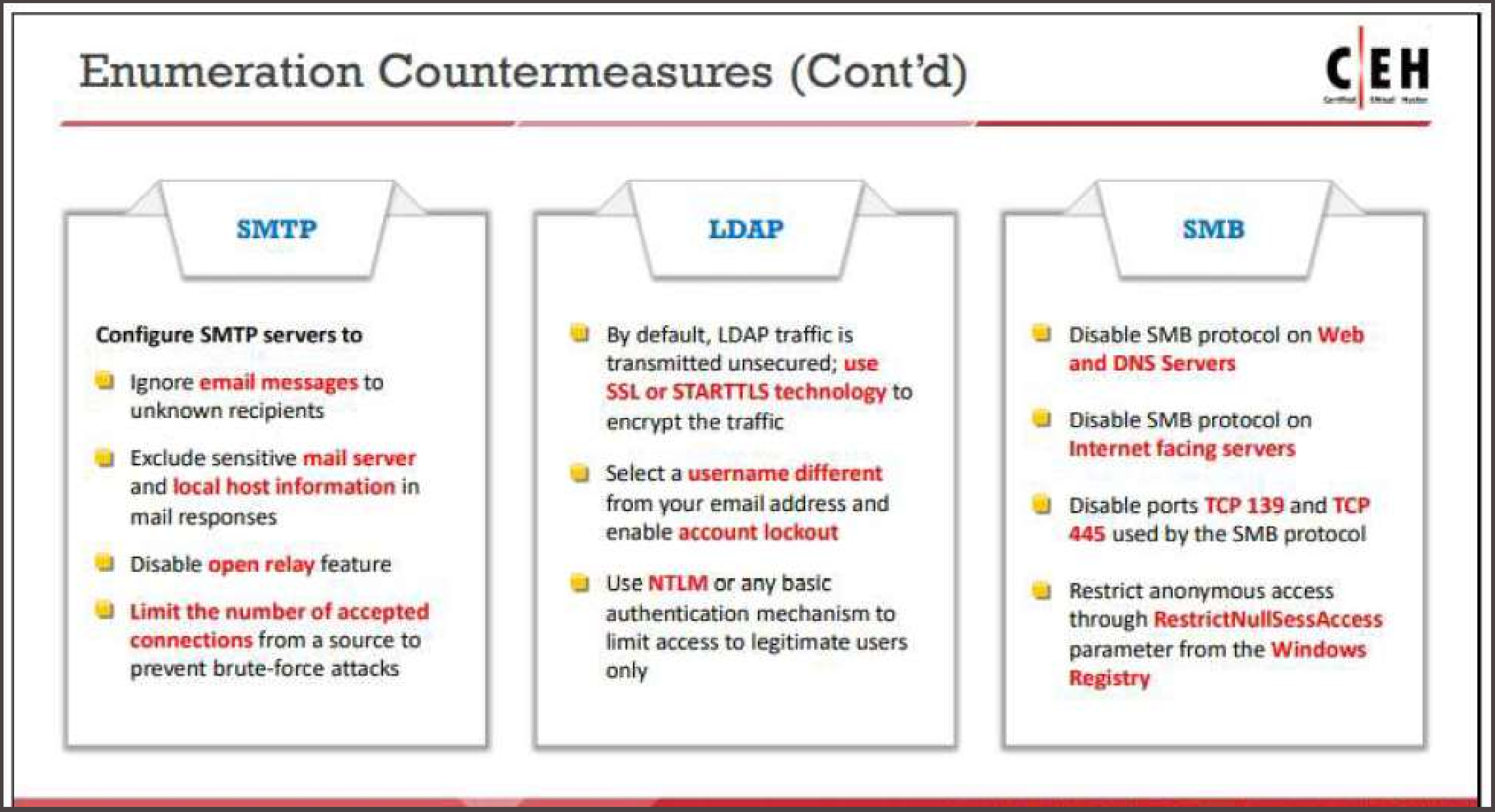
Enumeration Countermeasures

SNMP

- **Remove the SNMP agent** or turn off the SNMP service
- If shutting off SNMP is not an option, then change the default **community string names**
- **Upgrade to SNMP3**, which encrypts passwords and messages
- Implement the Group Policy security option called "**Additional restrictions for anonymous connections**"
- Ensure that the access to **null session pipes**, **null session shares**, and IPSec filtering is restricted
- **Do not misconfigure SNMP service** with read-write authorization

DNS

- **Disable** the DNS zone transfers to the untrusted hosts
- Ensure that the private hosts and their IP addresses are not published in **DNS zone files** of public DNS servers
- Use **premium DNS registration services** that hide sensitive information, such as host information (HINFO) from the public
- Use **standard network admin contacts** for DNS registrations to avoid social engineering attacks



Enumeration Countermeasures (Cont'd)

SMTP

Configure SMTP servers to

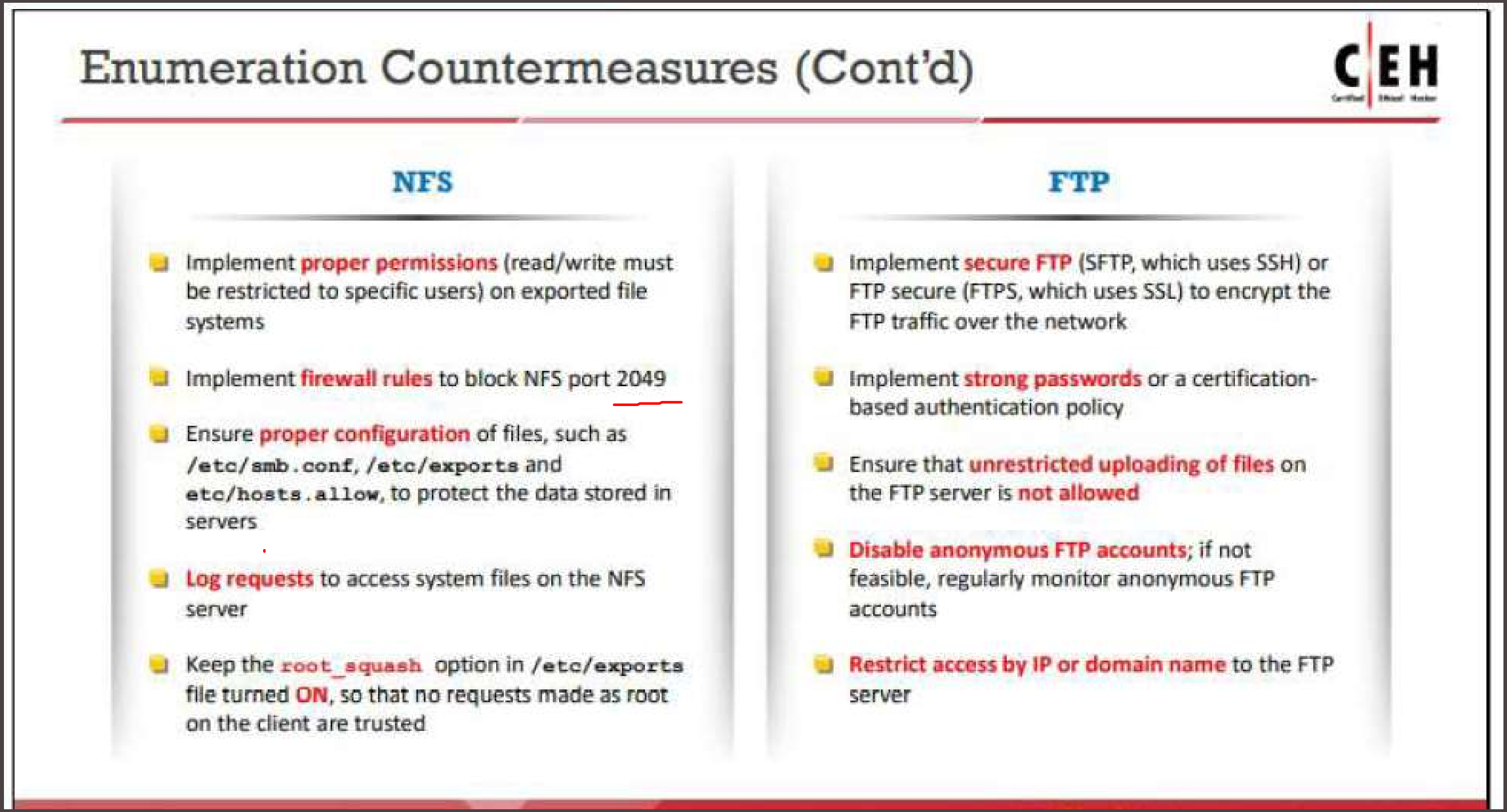
- Ignore **email messages** to unknown recipients
- Exclude sensitive **mail server** and **local host information** in mail responses
- Disable **open relay** feature
- **Limit the number of accepted connections** from a source to prevent brute-force attacks

LDAP

- By default, LDAP traffic is transmitted unsecured; **use SSL or STARTTLS technology** to encrypt the traffic
- Select a **username different** from your email address and enable **account lockout**
- Use **NTLM** or any basic authentication mechanism to limit access to legitimate users only

SMB

- Disable SMB protocol on **Web and DNS Servers**
- Disable SMB protocol on **Internet facing servers**
- Disable ports **TCP 139** and **TCP 445** used by the SMB protocol
- Restrict anonymous access through **RestrictNullSessAccess** parameter from the **Windows Registry**



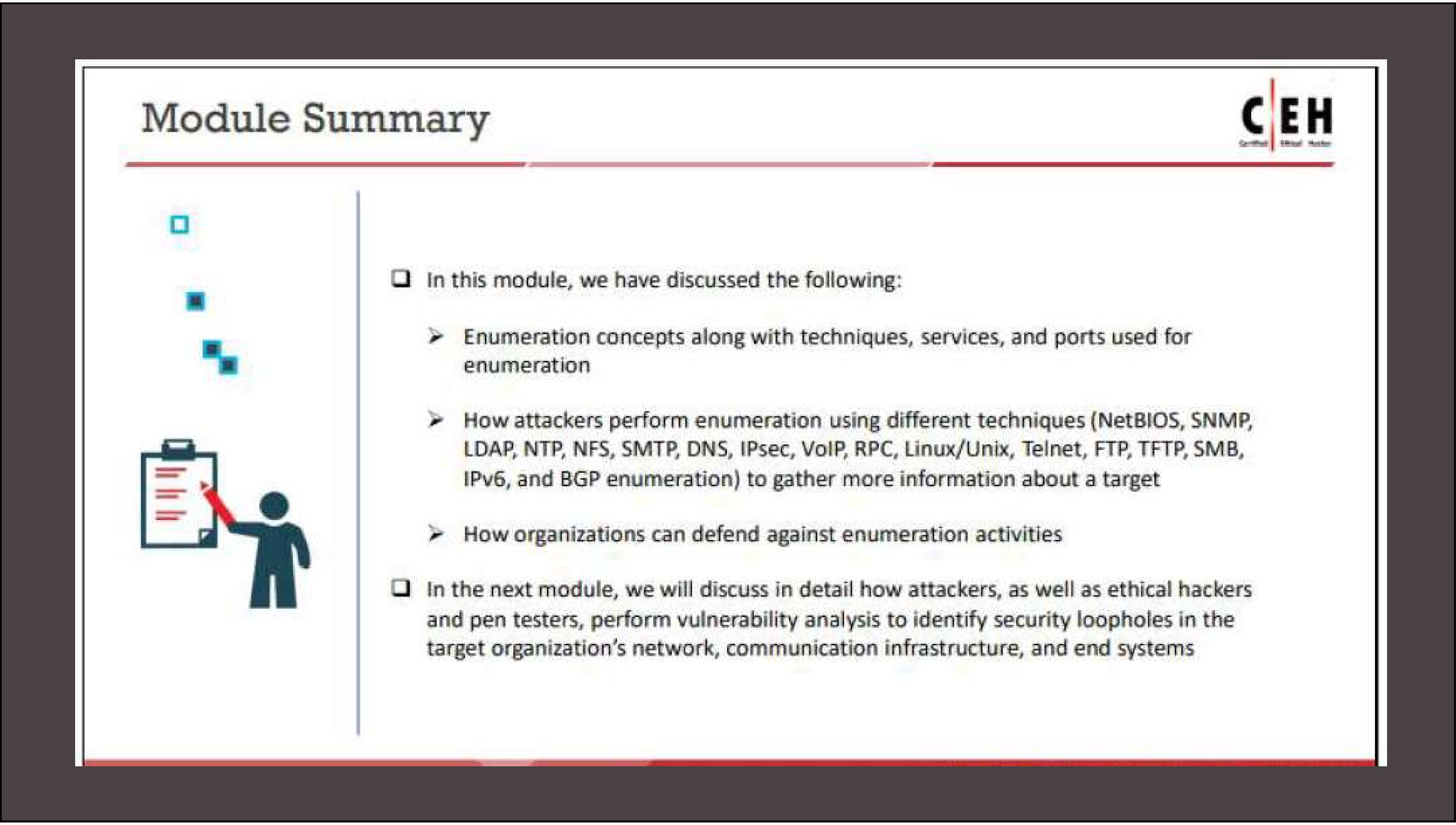
Enumeration Countermeasures (Cont'd)

NFS

- Implement **proper permissions** (read/write must be restricted to specific users) on exported file systems
- Implement **firewall rules** to block NFS port 2049
- Ensure **proper configuration** of files, such as `/etc/smb.conf`, `/etc/exports` and `etc/hosts.allow`, to protect the data stored in servers
- **Log requests** to access system files on the NFS server
- Keep the `root_squash` option in `/etc/exports` file turned **ON**, so that no requests made as root on the client are trusted

FTP

- Implement **secure FTP** (SFTP, which uses SSH) or FTP secure (FTPS, which uses SSL) to encrypt the FTP traffic over the network
- Implement **strong passwords** or a certification-based authentication policy
- Ensure that **unrestricted uploading of files** on the FTP server is **not allowed**
- **Disable anonymous FTP accounts**; if not feasible, regularly monitor anonymous FTP accounts
- **Restrict access by IP or domain name** to the FTP server



Module Summary

C|EH

❑ In this module, we have discussed the following:

- ➢ Enumeration concepts along with techniques, services, and ports used for enumeration
- ➢ How attackers perform enumeration using different techniques (NetBIOS, SNMP, LDAP, NTP, NFS, SMTP, DNS, IPsec, VoIP, RPC, Linux/Unix, Telnet, FTP, TFTP, SMB, IPv6, and BGP enumeration) to gather more information about a target
- ➢ How organizations can defend against enumeration activities

❑ In the next module, we will discuss in detail how attackers, as well as ethical hackers and pen testers, perform vulnerability analysis to identify security loopholes in the target organization's network, communication infrastructure, and end systems