# Enumerating & Footprinting Services

LordOsiris

## Linux Remote Management Protocols

### SSH - 22

- Enumeration & Footprinting the Service — SSH — 22
  - Nmap scripts
    - nmap -p22 <ip> -sC — Send default nmap scripts for SSH
    - nmap -p22 <ip> -sV — Retrieve version
    - nmap -p22 <ip> --script ssh2-enum-algos — Retrieve supported algorithms
    - nmap -p22 <ip> --script ssh-hostkey --script-args ssh_hostkey=full — Retrieve weak keys
    - nmap -p22 <ip> --script ssh-auth-methods --script-args="ssh.user=root" — Check authentication methods
  - msf> use scanner/ssh/ssh_enumusers — Username Enumeration Metasploit
  - nc -vn <IP> 22 — Enumeration using Banner Grabbing.
  - ssh <user>@<FQDN/IP> -o PreferredAuthentications=password — Enforce password-based authentication.
  - ssh -i private.key <user>@<FQDN/IP> — Log in to the SSH server using private key.
  - ssh <user>@<FQDN/IP> — Log in to the SSH server using the SSH client.
  - https://github.com/rapid7/ssh-badkeys/tree/master/authorized ... -f — Known badkeys
  - ssh-audit.py [-1246pbcdjvt8] <host> — Automated ssh-audit

### RSYNC ---> 873/TCP

- Rsync — 873/tcp
  - Scanning for Rsync using Nmap
    - sudo nmap -sV -p 873 127.0.0.1
  - Probing for Accessible Shares
    - nc -nv 127.0.0.1 873
  - Enumerating an Open Share
    - rsync -av --list-only rsync://127.0.0.1/dev

### R-Services ---> 512/513/514

- R-Services — 512 / 513 / 514
  - Logging in Using Rlogin
    - rlogin 10.0.17.2 -l lordosiris
  - Listing Authenticated Users Using Rwho
    - rwho
  - Listing Authenticated Users Using Rusers
    - rusers -al 10.0.17.5

## Windows Remote Management Protocols

### WMI ---> 135

- WMI — 135
  - using WMIexec.py
    - the program wmiexec.py from the Impacket toolkit can be used for footprinting WMI.
    - /usr/share/doc/python3-impacket/examples/wmiexec.py CryptIt3:"P455w0rD!"@10.129.201.248 "hostname"

### WinRM ---> 5985/5986

- WinRM — 5985 / 5986
  - Nmap WinRM footprinting
    - nmap -sV -sC 10.129.201.248 -p5985,5986 --disable-arp-ping -n
  - using Evil-winrm
    - we can use the tool called evil-winrm, another penetration testing tool designed to interact with WinRM
    - evil-winrm -i 10.129.201.248 -u CryptIt3 -p P455w0rD!

### RDP ---> 3389

- RDP — 3389
  - Nmap Scan
    - nmap -sV -sC 10.129.201.248 -p3389 --script rdp*
  - RDP Security Check
    - A Perl script named rdp-sec-check.pl has also been developed by Cisco CX Security Labs that can unauthentically identify the security settings of RDP servers based on the handshakes.
    - ./rdp-sec-check.pl 10.129.201.248
  - Using xfreerdp to initiate RDP session
    - xfreerdp /u:cryptit3 /p:"P455w0rd!" /v:10.129.201.248

## Host Based Enumeration

### SMB ---> 137 / 138 / 139 / 445

- 137 / 138 / 139 / 445 — SMB — Enumeration & Footprinting the Service
  - Metasploit Search exploit
    - msf> search type:exploit platform:windows target:2008 smb
    - msf> searchsploit microsoft smb
  - Obtain Information
    - Dump interesting information
      - enum4linux -a [-u "<username>" -p "<passwd>"] <IP>
      - enum4linux-ng -A [-u "<username>" -p "<passwd>"] <IP>
      - nmap --script "safe or smb-enum-*" -p 445 <IP>
    - Connect to the rpc
      - rpcclient -U "" -N <IP>
      - rpcclient //machine hfn -U domain.local/USERNAME%754d87d42adabcca32bdb34a876b6ffb --pw-nt-hash
      - rpcclient -U "username%passwd" <IP>
    - Dump user information
      - /usr/share/doc/python3-impacket/examples/samrdump.py -port 139 [[domain/]username[:password]@]<targetName or address>
      - /usr/share/doc/python3-impacket/examples/samrdump.py -port 445 [[domain/]username[:password]@]<targetName or address>
    - Map possible RPC endpoints
      - /usr/share/doc/python3-impacket/examples/rpcdump.py -port 135 [[domain/]username[:password]@]<targetName or address>
      - /usr/share/doc/python3-impacket/examples/rpcdump.py -port 139 [[domain/]username[:password]@]<targetName or address>
      - /usr/share/doc/python3-impacket/examples/rpcdump.py -port 445 [[domain/]username[:password]@]<targetName or address>
  - Enumerate Users, Groups & Logged On Users
    - crackmapexec smb 10.10.10.10 --users [-u <username> -p <password>]
    - crackmapexec smb 10.10.10.10 --groups [-u <username> -p <password>]
    - crackmapexec smb 10.10.10.10 --groups --loggedon-users [-u <usernames> -p <password>]
    - ldapsearch -x -b "DC=DOMAIN,NAME,DC=LOCAL" -s sub "(&(objectclass=user))" -h 10.10.10.10 | grep -i samaccountname: | cut -f 2 -d " "
    - rpcclient -U "" -N 10.10.10.10
    - Impacket - Enumerate local users
      - lookupsid.py -no-pass hostname.local
    - Metasploit - Enumerate local users
      - use auxiliary/scanner/smb/smb_lookupsid
    - Brute Forcing User RIDs
      - samrdump.py 10.129.14.128
    - This info should already being gathered from enum4linux and enum4linux-ng
  - Enum4Linux-ng - Enumeration
    - enum4linux - General enumeration - anonymous session
      - enum4linux -a <target>
    - enum4linux - General enumeration - authenticated session
      - enum4linux -a <target> -u <user> -p <pass>
    - enum4linux - Users enumeration
      - enum4linux -u <user> -p <pass> -U <target>
    - enum4linux - Group and members enumeration
      - enum4linux -u <user> -p <pass> -G <target>
    - enum4linux - Password policy
      - enum4linux -u <user> -p <pass> -P <target>
  - Download files
    - smbclient //<IP>/<share>
    - > mask ""
    - > recurse
    - > prompt
    - > mget *
  - smbclient
    - List shares on a machine using NULL Session
      - smbclient -L
    - Connect to a valid share with username + password
      - smbclient //<target>/<share$> -U username%password
    - List shares on a machine using a valid username + password
      - smbclient -L <target-IP> -U username%password
    - List files on a specific share
      - smbclient //<target>/<share$> -c 'ls' password -U username
    - List files on a specific share folder inside the share
      - smbclient //<target>/<share$> -c 'cd folder; ls' password -U username
    - Download a file from a specific share folder
      - smbclient //<target>/<share$> -c 'cd folder;get desired_file_name' password -U username
    - Copy a file to a specific share folder
      - smbclient //<target>/<share$> -c 'put /var/www/my_local_file.txt .\target_folder\target_file.txt' password -U username
    - Create a folder in a specific share folder
      - smbclient //<target>/<share$> -c 'mkdir .\target_folder\new_folder' password -U username
    - Rename a file in a specific share folder
      - smbclient //<target>/<share$> -c 'rename current_file.txt new_file.txt' password -U username

### IMAP / POP3 ---> 110 / 143 / 993 / 995

- 110 / 143 / 993 / 995 — IMAP / POP3 — Enumeration & Footprinting the Service
  - Enumerating IMAP/POP3 using Nmap
    - sudo nmap -sC -sV -v -p 110,143,993,995 <Target IP>
  - Connecting to IMAP using openssl
    - openssl s_client -connect <Target IP>:imaps
  - Connecting to POP3 using openssl
    - openssl s_client -connect <Target IP>:pop3s
  - If credentials are obtained we can get information using cURL
    - 1. Listing mailboxes
      - curl -k 'imaps://1.2.3.4/' --user user:pass
      - LIST "" "*"
    - 2. Listing messages in a mailbox
      - curl -k 'imaps://1.2.3.4/INBOX?ALL' --user user:pass
      - SELECT INBOX and then SEARCH ALL
    - 3. searching for drafts with password in mail body
      - curl -k 'imaps://1.2.3.4/Drafts;MAILINDEX=1' --user user:pass
      - SELECT Drafts and then FETCH 1 BODY[]
  - Banner grabbing
    - nc -nv <IP> 143
    - openssl s_client -connect <IP>:993 -quiet

### FTP ---> 21

- 21 — FTP — Enumeration & Footprinting the Service
  - NMAP Scripts
    - nmap --script ftp-* -p 21 <ip>
      - ftp-syst
      - ftp-vsftpd-backdoor
      - ftp-vsftp-cve2010-4221
      - ftp-proftpd-backdoor
      - ftp-bounce
      - ftp-libopie
      - ftp-anon
      - ftp-brute
  - Interact with the FTP service on the target.
    - ftp <FQDN/IP>
  - Banner Grabbing
    - nc -nv <FQDN/IP> 21
  - Interact with the FTP service on the target.
    - telnet <FQDN/IP> 21
  - Interact with the FTP service on the target using encrypted connection.
    - openssl s_client -connect <FQDN/IP>:21 -starttls ftp
  - Download all available files on the target FTP server.
    - wget -m --no-passive ftp://<USER>:<Pass>@<target>

### IPMI ---> 623/UDP

- 623 / UDP — IPMI — Enumeration & Footprinting the Service
  - Enumerating using Nmap
    - sudo nmap -sU --script ipmi-version -p 623 ilo.inlanfreight.local
  - Enumerating using Metasploit
    - msf6 > use auxiliary/scanner/ipmi/ipmi_version
  - Metasploit Dumping Hashes
    - msf6 > use auxiliary/scanner/ipmi/ipmi_dumphashes

### MSSQL ---> 1433

- 1433 — MSSQL
  - MSSQL Ping in Metasploit
    - scanner/mssql/mssql_ping
  - Connecting with Mssqlclient.py
    - python3 mssqlclient.py Administrator@10.129.201.248 -windows-auth

### MySQL ---> 3306

- 3306 — MySQL
  - Local
    - Connect to root without password
      - mysql -u root
    - A password will be asked
      - mysql -u root -p
  - Remote
    - mysql -h <Hostname> -u root
    - mysql -h <Hostname> -u root@localhost

### NFS ---> 2049/111

- 2049 / 111 — NFS — Enumeration
  - Show available NFS Shares
    - showmount -e <TargetIP>
  - Mounting NFS Share
    - mount -t nfs <TargetIP>:/ ./<internalfolder mount point>/ -o nolock
  - NMAP
    - sudo nmap -p 111,2049 -sV -sC <TargetIP>
  - Useful nmap scripts
    - List NFS exports and check permissions
      - nfs-ls
    - Like showmount -e
      - nfs-showmount
    - Disk statistics and info from NFS share
      - nfs-statfs
  - Useful metasploit modules
    - Scan NFS mounts and list permissions
      - scanner/nfs/nfsmount

### SNMP ---> 161 / 162 / 10161 / 10162 /udp

- 161 / 162 / 10161 / 10162 /udp — SNMP — Enumeration & Footprinting the Service
  - Connecting using SNMPwalk
    - snmpwalk -v2c -c public 10.129.14.128
  - OneSixtyOne for bruteforcing SNMP community strings
    - onesixtyone -c /opt/useful/SecLists/Discovery/SNMP/snmp.txt <Target IP>
  - braa to brute-force the individual OIDs and enumerate the information
    - braa <community string>@<IP>:.1.3.6.*

### SMTP ---> 25/587

- 25 / 587 — SMTP — Enumeration & Footprinting the Service
  - Nmap
    - sudo nmap 10.129.14.128 -sC -sV -p25
    - nmap -p25 --script smtp-commands 10.10.10.10
    - nmap -p25 --script smtp-open-relay 10.10.10.10 -v
  - smtp-commands

| SMTP Commands to use | |
|---|---|
| | Description |
| AUTH PLAIN | AUTH is a service extension used to authenticate the client. |
| HELO | The client logs in with its computer name and thus starts the session. |
| MAIL FROM | The client names the email sender. |
| RCPT TO | The client names the email recipient. |
| DATA | The client initiates the transmission of the email. |
| RSET | The client aborts the initiated transmission but keeps the connection between client and server. |
| VRFY | The client checks if a mailbox is available for message transfer. |
| EXPN | The client also checks if a mailbox is available for messaging with this command. |
| NOOP | The client requests a response from the server to prevent disconnection due to time-out. |
| QUIT | The client terminates the session. |

  - Banner Grabbing
    - nc -vn <IP> 25
  - Username guessing using smtp-user-enum
    - smtp-user-enum -M VRFY -U footprinting-wordlist.txt -p 25 -w 15 -t 10.129.114.201