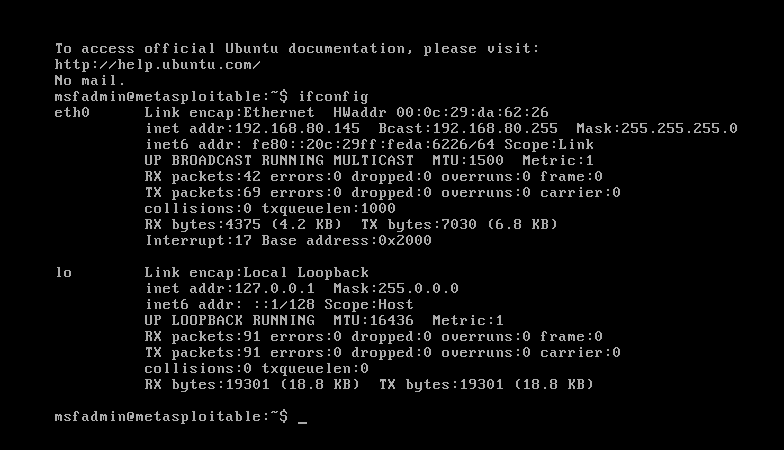
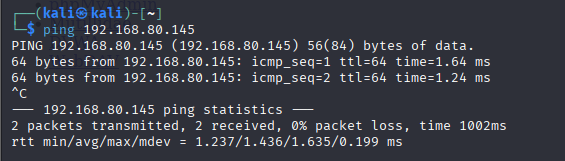
**METASPLOIT VAPT USING KALI LINUX.**

Looked for target ip in Metasploit by command:ifconfig



Checked is both machines are in same network or not by:

Ping Metasploit ip. In linux terminal.



Nmap tool used for checking ports and their version of service by command:

nmap -sC -sV -v -p- --min-rate 10000 ip.

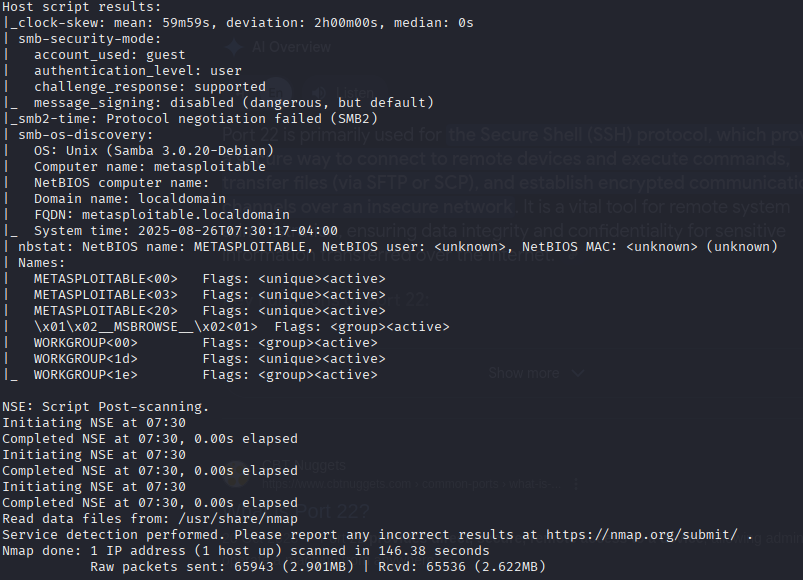
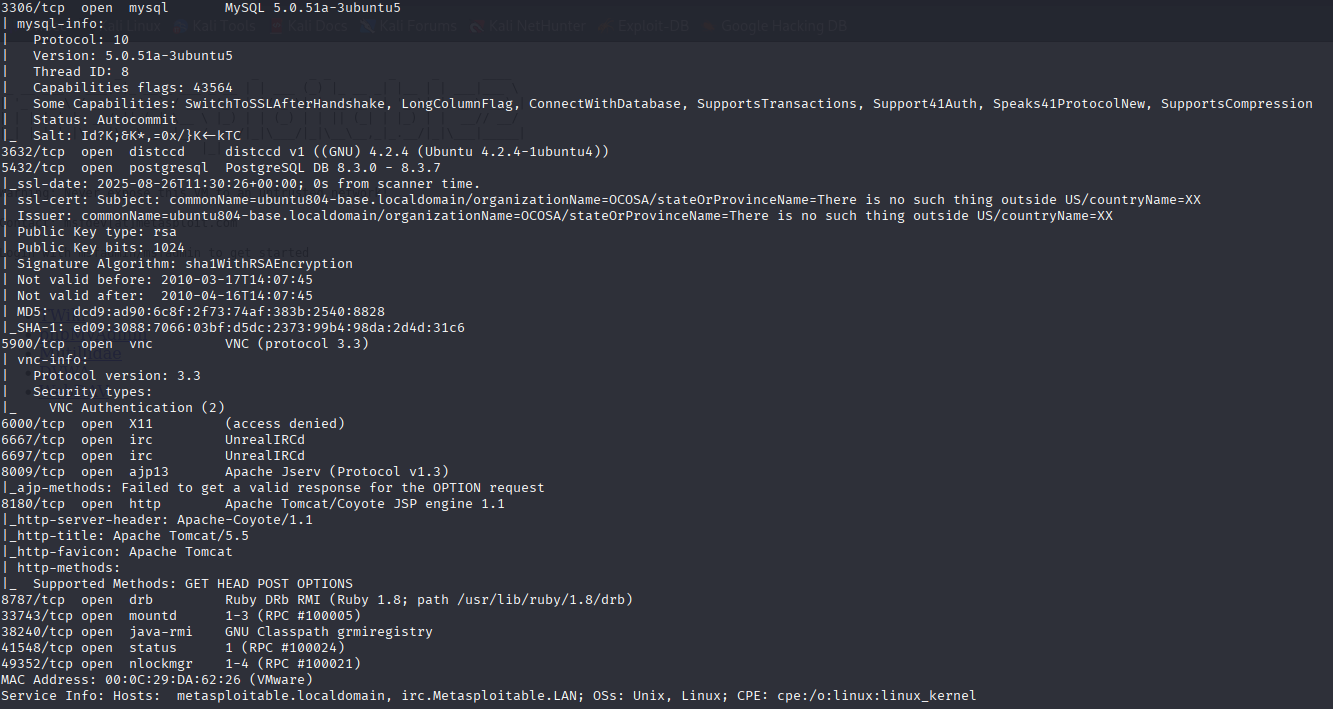
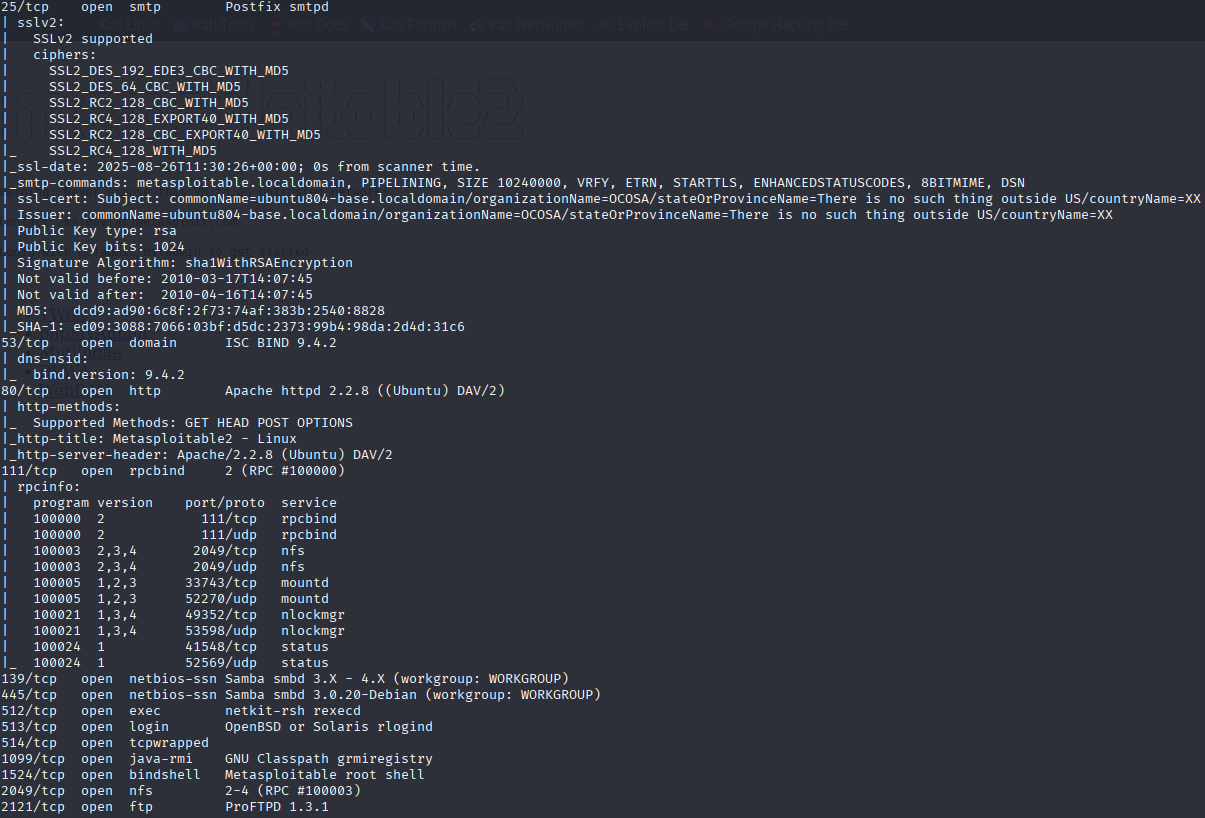
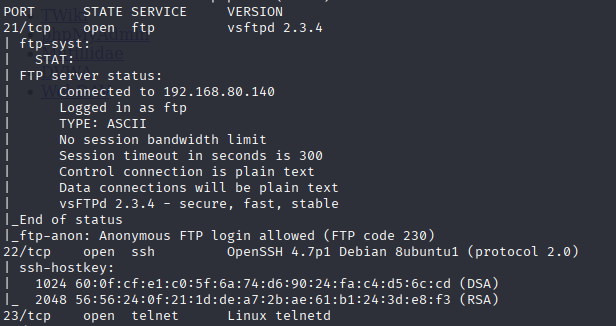
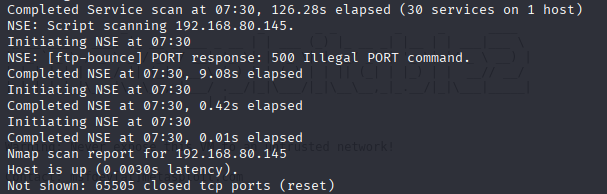
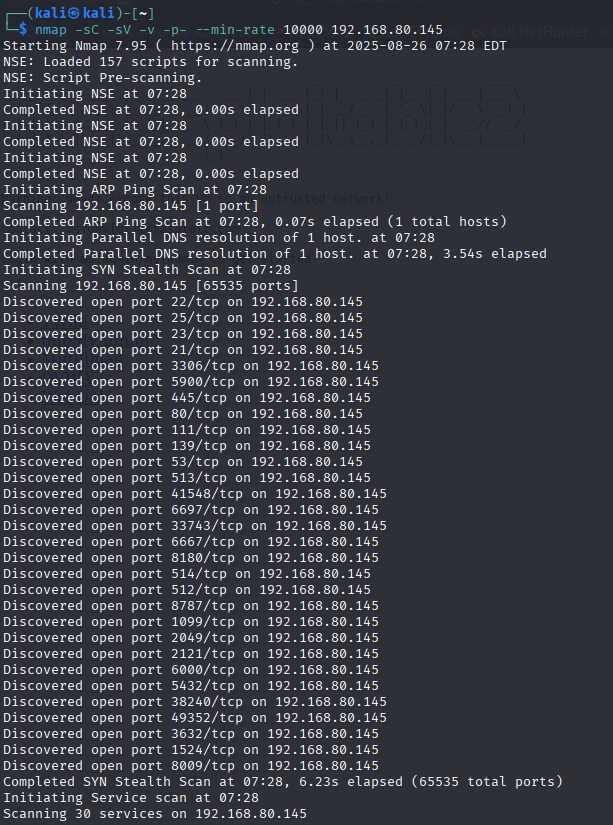
Here **-sC** used for selection of scripts that are considered **safe, non-intrusive, and useful** for basic discovery and information gathering.

**-sV** tag used for used for **service and version detection.**

**-v** tag:it is a verbose mode. It increases the level of detail in the scan output, providing more information about the scan's progress and what Nmap is doing in the background.

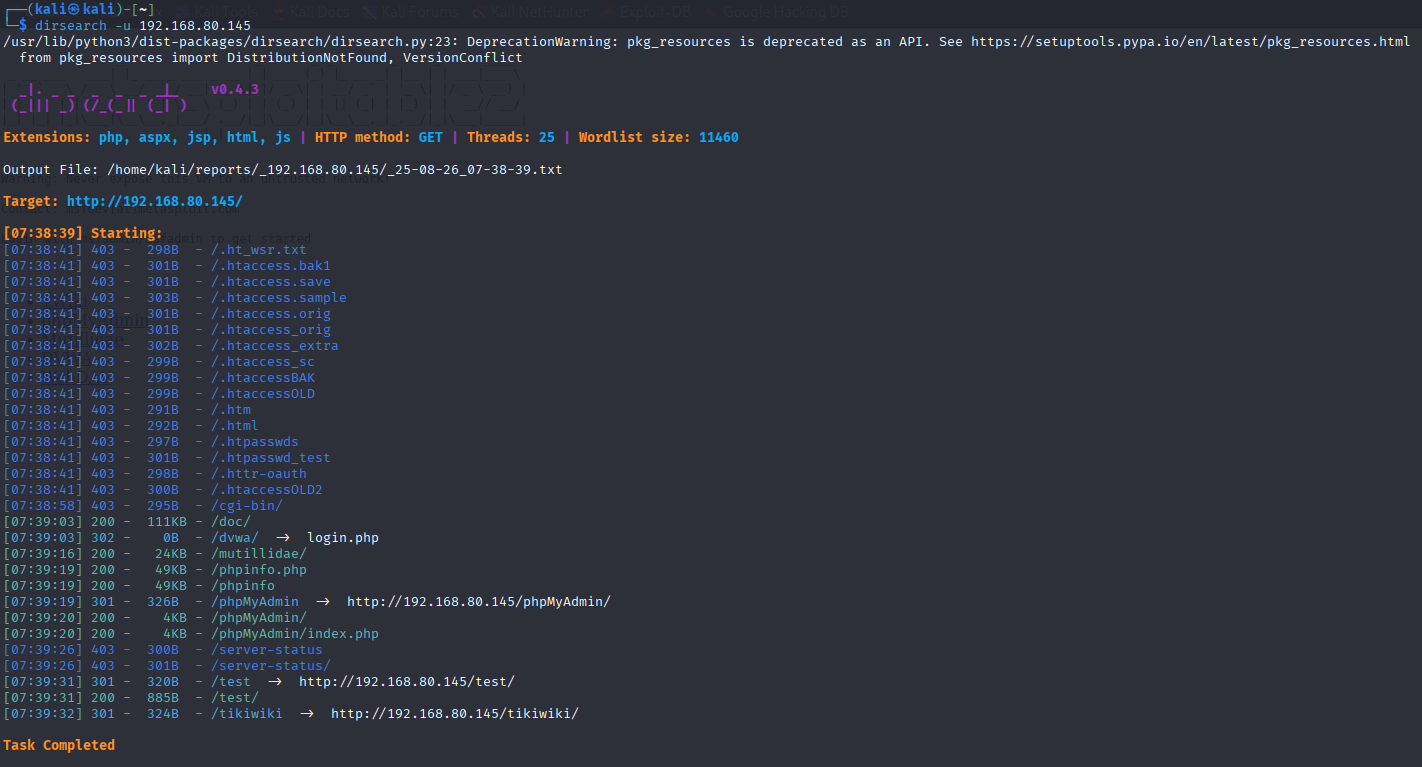
**-p-** tag : to **scan all 65,535 TCP ports** instead of the default 1,000 most common ports.default nmap scans fir 1000 ports of the target machine.

**--min-rate 10000** tag: performance-tuning option that forces the scanner to send packets at a **minimum rate of 10,000 packets per second**.



This is all about nmap tool used for the target ip.

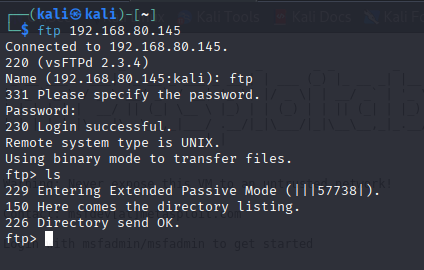
Searched for directories present in the target ip.



1.FTP on port 21:

FTP:is a file transfer protocol used to transfer files between target machine and our local machine.

To get FTP shell : **ftp ip** (in terminal)



Username:ftp

Password:ftp

Also tried cred for ftp

username:anonymous

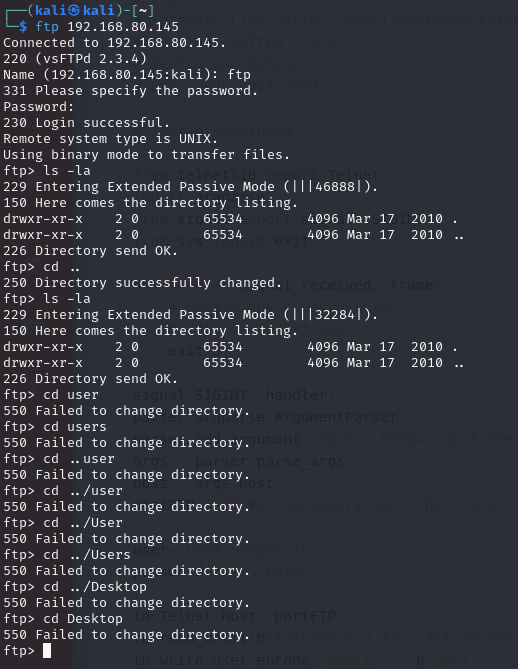
password:

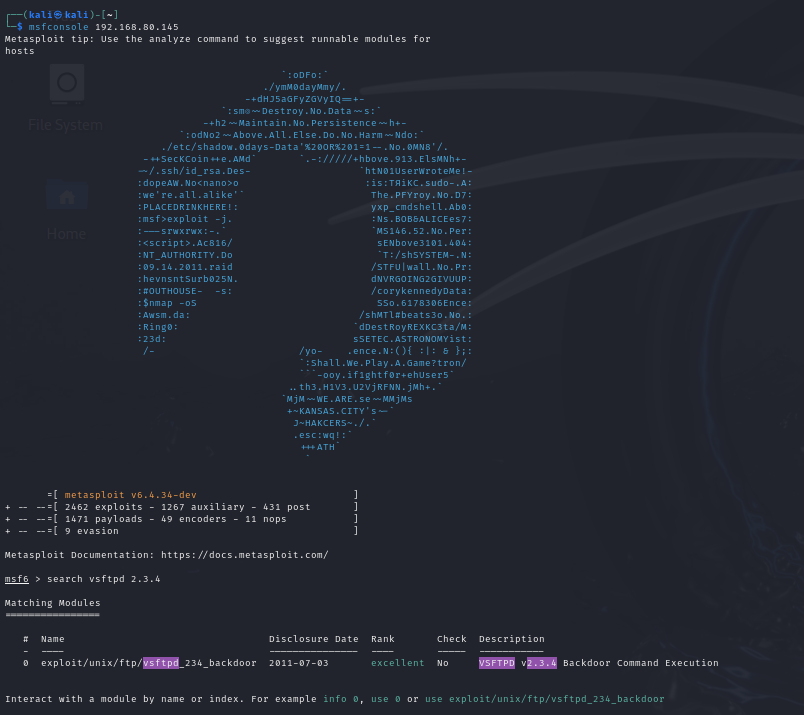
username:ftp

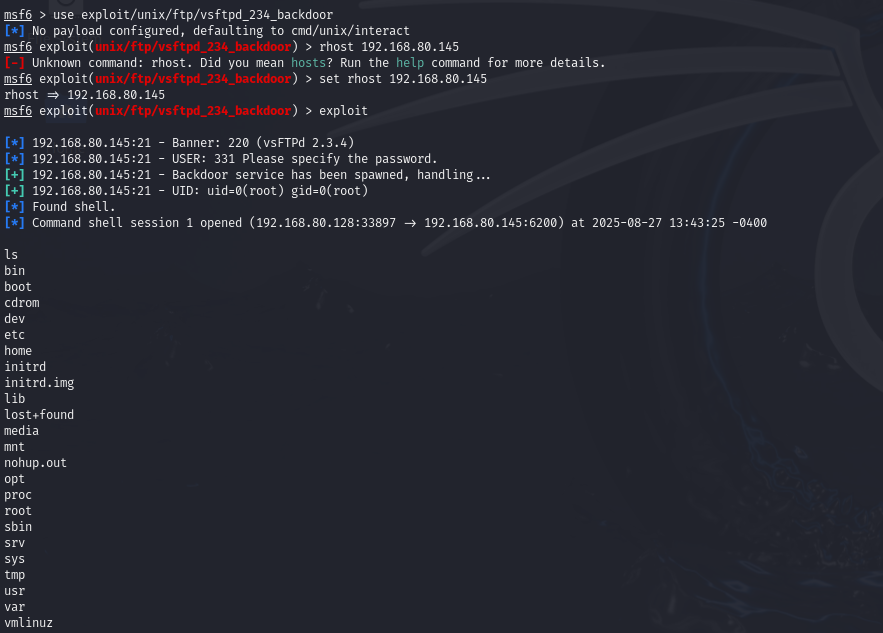
password:

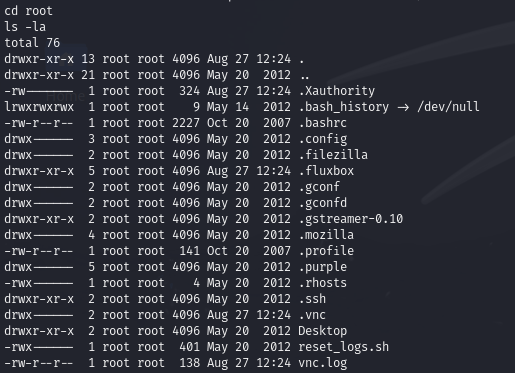
After got ftp shell looked for any confidential files like users,configuration files

Also tried many directories.but no config files or confidential files was present.and also don’t have privilege to put or get any file on [ftp so](ftp://ftp.so) moved on next port that is “ssh”.



Used Metasploit framework to exploit ftp:  






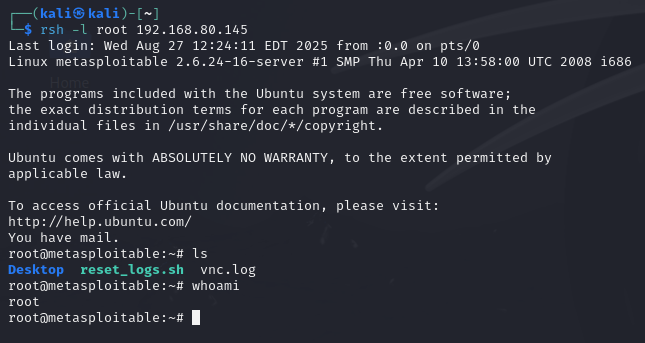




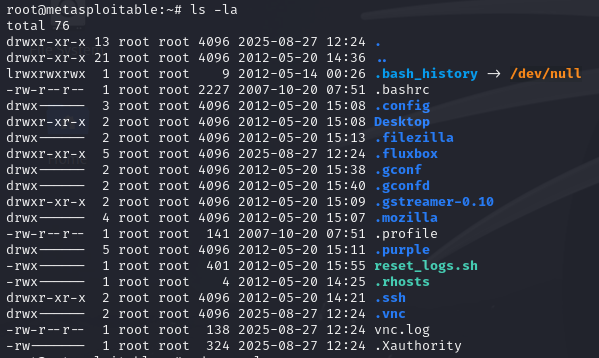
In the context of a .rhosts file, the ++ entry is an extremely dangerous security vulnerability. It's a wildcard that means:

**"Any user from any host is trusted and can log in to this account without a password."**

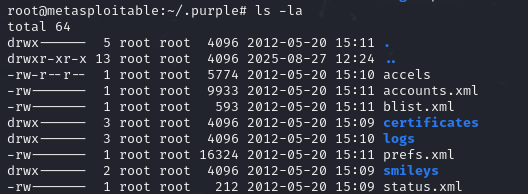
Now tried the login :

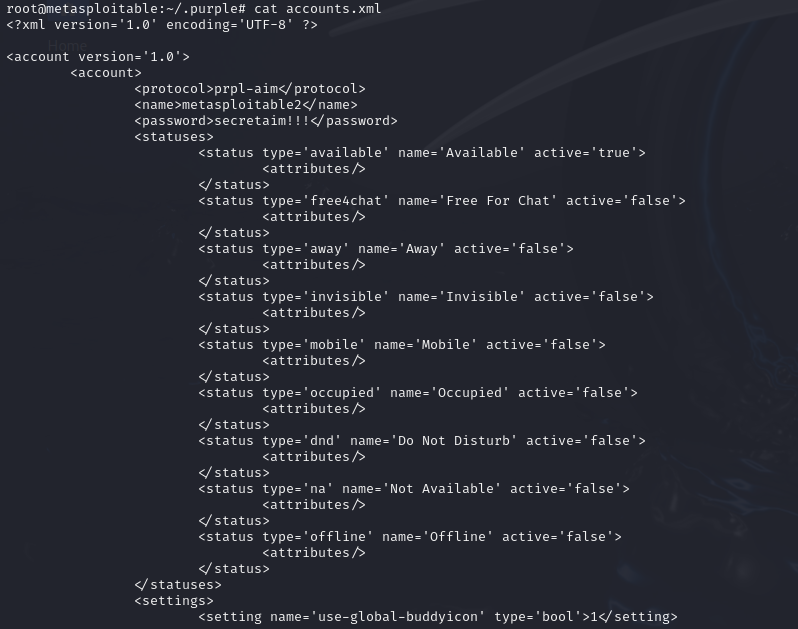


**NOW I AM ROOT OF THE TARGET IP.**







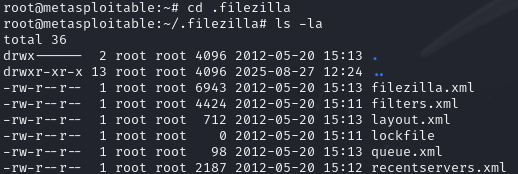


**Extracted Information from the accounts.xml**

The file contains login credentials for several accounts, organized by their respective protocols:

* **AIM (prpl-aim)**
  + **Username**: metasploitable2
  + **Password**: secretaim!!!
  + **Server**: login.messaging.aol.com
  + **Port**: 5190
* **Gadu-Gadu (prpl-gg)**
  + **Username**: metasploitable2
  + **Password**: secret!!@#!
* **Jabber/XMPP (prpl-jabber)**
  + **Username**: metasploitable2@gmail.com/Home
  + **Password**: s3cr3t!
  + **Port**: 5222
* **Novell GroupWise Messenger (prpl-novell)**
  + **Username**: metasploitable2
  + **Password**: grouwis0r!
  + **Port**: 8300
* **IRC (prpl-irc)**
  + **Username**: metasploitable2@irc.ubuntu.com
  + **Password**: invisible!
  + **Port**: 6667
* **MSN (prpl-msn)**
  + **Username**: metasploitable2@metasploit.com
  + **Password**: secrets!
  + **Server**: messenger.hotmail.com
  + **Port**: 1863
* **Yahoo (prpl-yahoo)**
  + **Username**: metasploitable2
  + **Password**: slinkies!!!!
  + **Server**: scs.msg.yahoo.com
  + **Port**: 5050

**cd .filezilla**



**From filezilla.xml**

I got

**Username:** msfadmin

**Password:** msfadmin

They are stored in plain text within the filezilla.xml configuration file.

**From recentservers.xml**

**All Credentials Found**

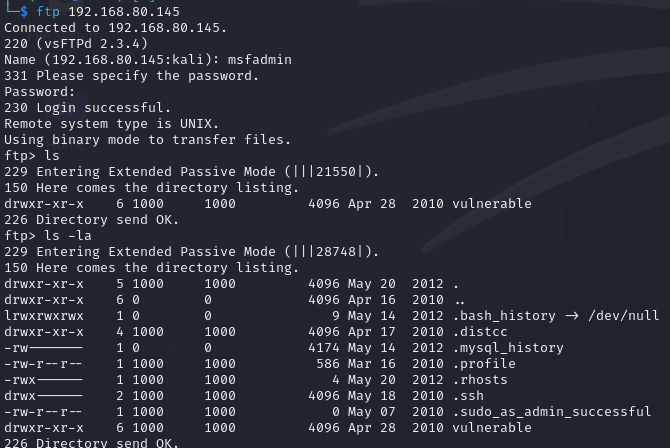
Here are the specific credentials extracted from the file:

* **Server 1**
  + **Host:** localhost
  + **Username:** msfadmin
  + **Password:** msfadmin
* **Server 2**
  + **Host:** localhost
  + **Username:** metasploitable2
  + **Password:** secrets!
* **Server 3**
  + **Host:** ssdf
  + **Username:** ftp
  + **Password:** sdfsdfs@
* **Server 4**
  + **Host:** ftp.openbsd.org
  + **Username:** ftp
  + **Password:** sdfsdfs@

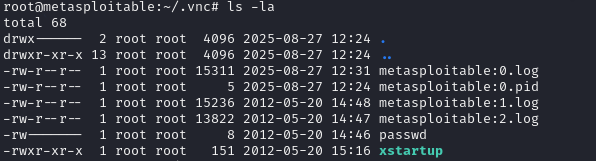
**Security Risk**

Similar to the filezilla.xml file, this recentservers.xml file exposes sensitive login information to anyone who can access the file system. Because the usernames and passwords are not encrypted, an attacker who compromises the host can easily retrieve these credentials and potentially use them to access other systems or services.

Login to server 1:



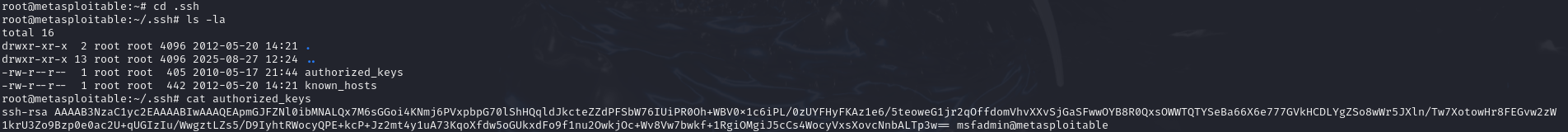
In **cd .vnc**:





This is VNC hashed format passwd it can be decrypted.

**cd .ssh**



From here I got :

This is a **public SSH key.**

This string is the contents of a public key file, typically named id\_rsa.pub. It's used for **public-key cryptography**, a system that uses two mathematically linked keys: a public key and a private key.

**username and hostname** associated to **msfadmin@metasploitable.**

**cat .Xauthority**

****

The cat .Xauthority output contains confidential data in the form of a "magic cookie." This data is a session key used for authentication to the X Window System, which is the graphical interface on Linux and Unix.

**2.SSH(secure shell)(port 22):**

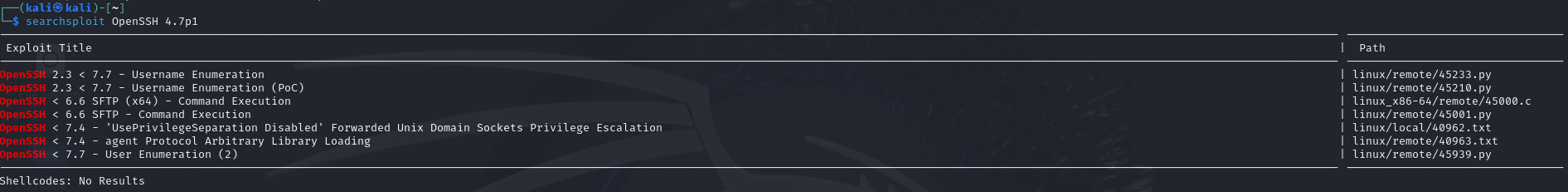
SSH service used to get reverse shell of the target ip.

nc -nv 192.168.80.145 22 command used to get version of port 22 after this command I got “(UNKNOWN) [192.168.80.145] 22 (ssh) open

SSH-2.0-OpenSSH\_4.7p1 Debian-8ubuntu1”

Used command for get exploit of the target version of SSH:

“searchsploit OpenSSH 4.7p1”



Then command to get the exploit locally:

“searchsploit -m linux/remote/45233.py”

Got the exploit in local directory after try to exploit ssh using Metasploit framework:

**msfconsole -q**

**msf6 > use exploit/linux/ssh/45233.py**

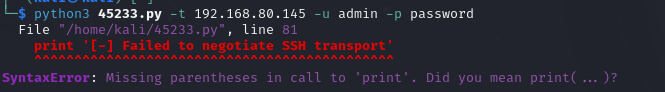
[-] No results from search

[-] Failed to load module: exploit/linux/ssh/45233

AND

Also tried manual python file on target ip

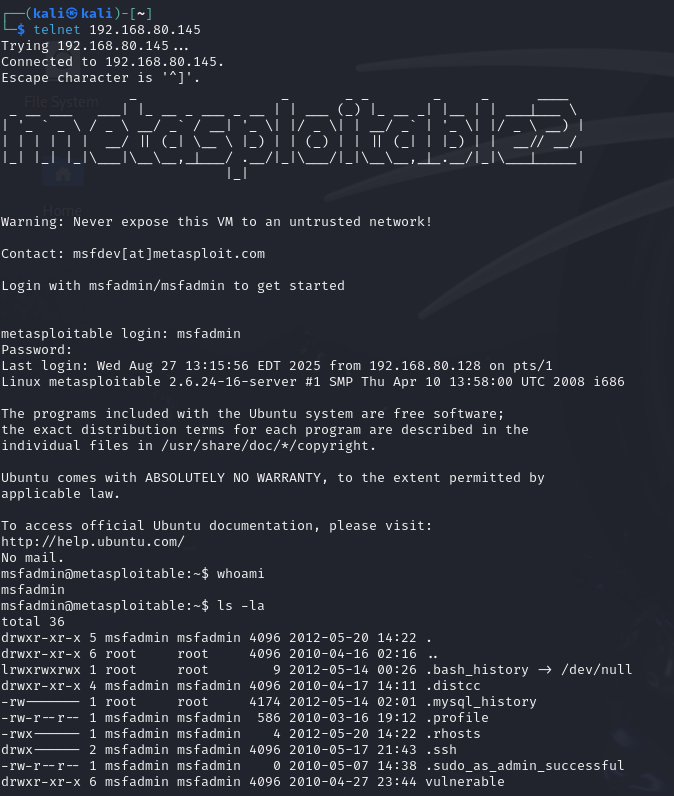
**python3 45233.py -t 192.168.80.145 -u admin -p password**

****

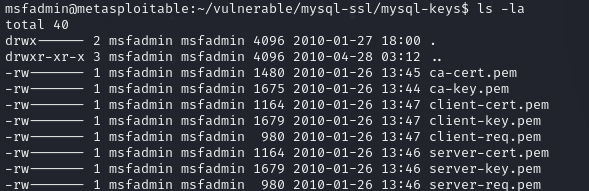
**3.telnet(port 23):**

Telnet is a simple, text-based protocol used to establish a connection to a remote server. It allows a user to control a remote computer as if they were physically logged in.

* Remote Access: It was historically used by network administrators to manage devices like servers, routers, and switches from a different location.
* Checking Network Services: Since it's a simple protocol, it's often used by developers and sysadmins as a quick way to test if a specific port on a server is open and responding. For example, you can use Telnet to connect to a web server on port 80 to see if it's accepting HTTP requests.

****

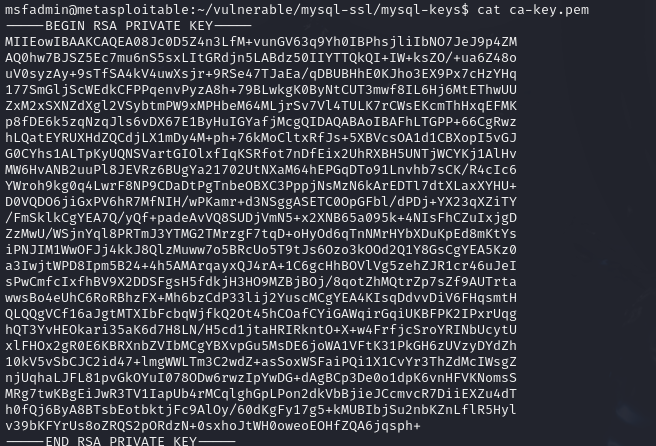
In **vulnerable/** **mysql-ssl/mysql-keys:**

****

**By cat ca-key.pem:**

I got **private key** which is very confidential data.

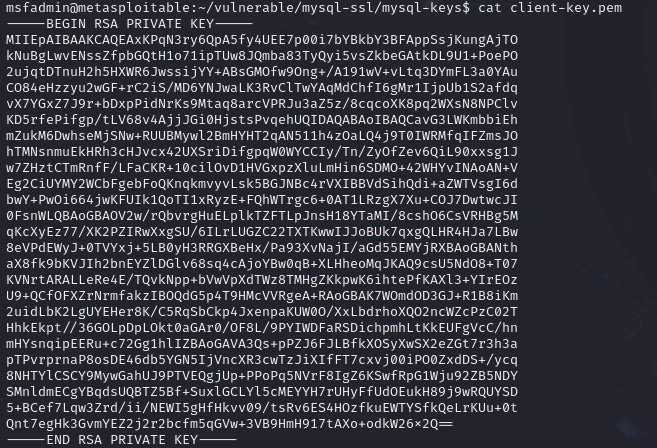
The file content is a **certificate authority (CA) private key** in PEM format. This is the cryptographic key used to sign and issue other certificates (like server and client certificates).

****

we can get the private key in plain text by properly decryption method and steps.

By **cat client-key.pem:**

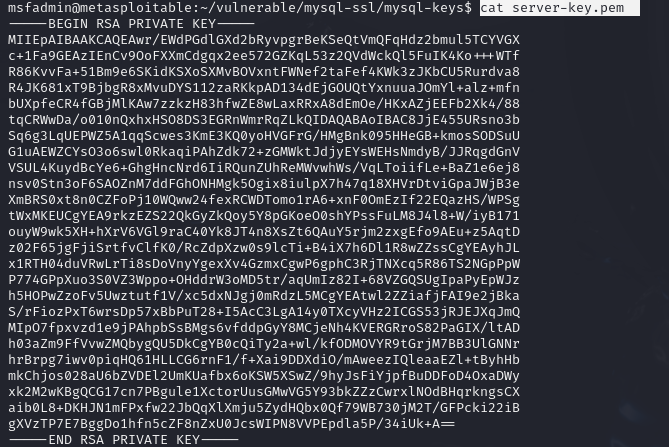
I got **private key** which is very confidential data.



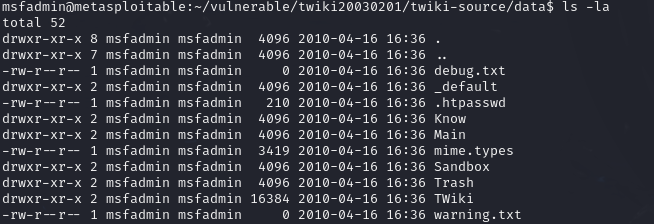
By **cat server-key.pem:**

the file server-key.pem contains **highly confidential data**.

This is a **server-side RSA private key**. It's a critical component in a public key infrastructure (PKI) system, used to secure communications.



In vulnerable/twiki20030201/twiki-source/data





The file is a list of usernames and their corresponding **hashed passwords**. This is a standard .htpasswd file used for HTTP Basic Authentication on web servers.

**The Confidential Data**

The confidential data here is the **hashed passwords** themselves.

* zK.G.uuPi39Qg
* CQdjUgwC6YckI
* h3i.9AzGUn4tQ
* zuUMZlkXvUR6Y
* 2fl31yuNhvMrU
* euHykHV5Q2miA
* pAVoSPpUf3xt2
* EI7XT7IJJV40A

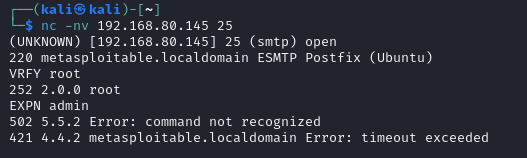
These are not the plain-text passwords, but they are derived from them. This particular format is the output of the outdated crypt() function, which uses a very weak hashing algorithm and a short salt. This makes the hashes easy to crack with modern password-cracking tools.

**4.smtp(port 25):simple mail transfer protocol.**

Purpose of Port 25

* Mail Transfer: Port 25 is primarily used for server-to-server email relay. This is how an email travels from one mail server to another on its way to the recipient.
* Initial Submission: It was historically also used for clients to send email to their mail servers, but this practice has largely been replaced due to security concerns.

Port 25 is a plain-text protocol, meaning **all data, including email content and metadata, is transmitted unencrypted**.



**5. domain name system(dns)(port 53):**

We can do dns tunnelling on this port.

SSH Tunneling:

There are three main types of SSH tunnels: Local, Remote, and Dynamic.

**6.HTTP(hyper text transfer protocol)(port 80):**

Now move to search for all directories in the target ip because it have port no. 80 is open so I opened the ip on browser it was like :



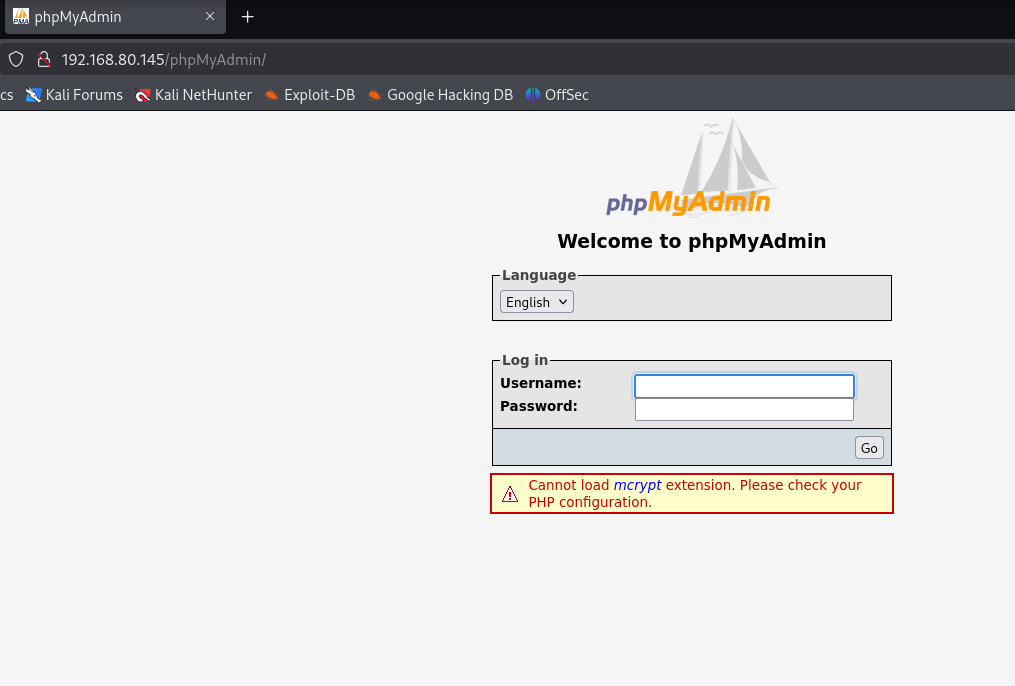
Now got port 80 so dirsearch tool used to get all directories within the machine on port 80.

Port 80 have HTTP service.

So using dirsearch for directory enumeration for target ip.

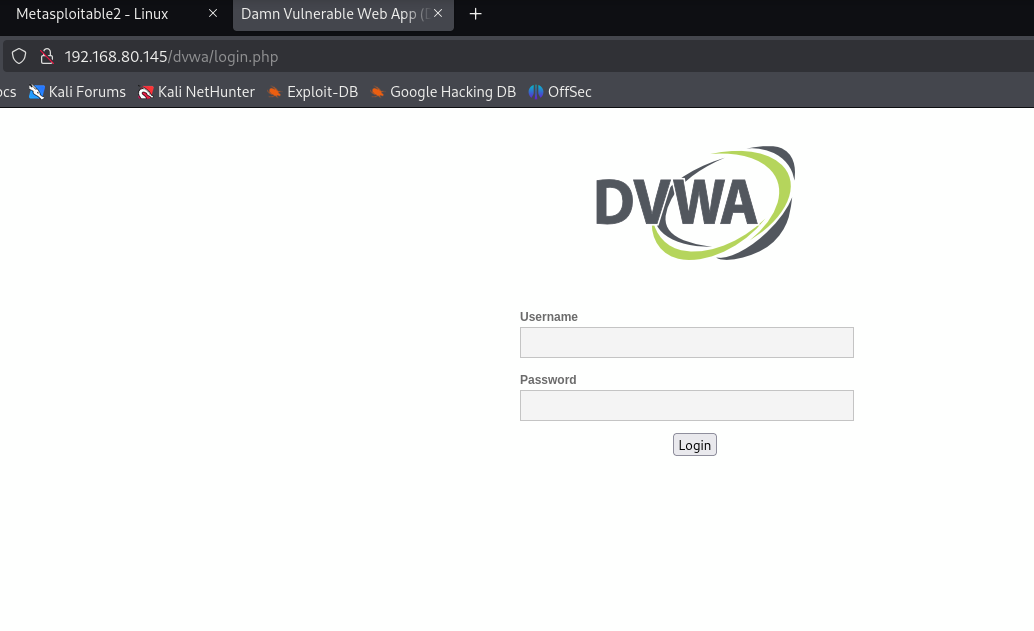


There is phpMyAdmin directory present :



Here we can do sql injection,password bruteforcing if we know any username etc.

Also have directory DVWA here also we can brute force and do sql injection.



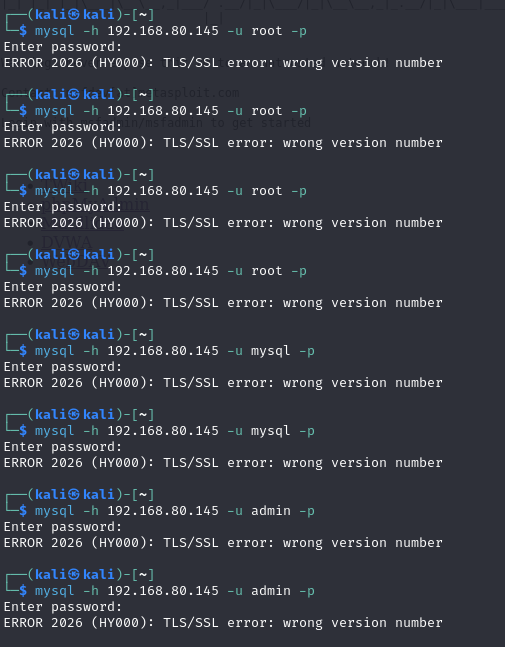
In mutillidae directory this is UI:

This is like a real world web application we can perform attacks here.



**7. mysql(port 3306):**

Tried manual try to get into mysql shell of target ip.

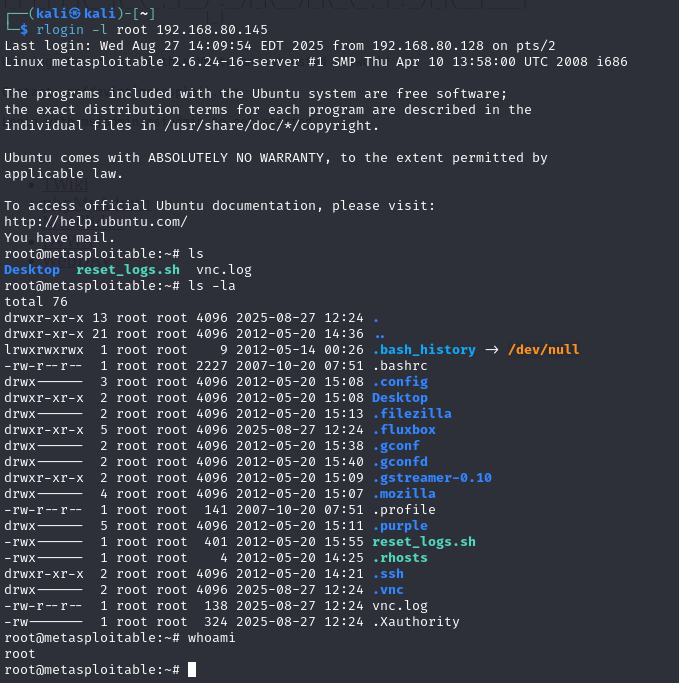
****

**8.rlogin(port 513)(remote login):**

rlogin relies on trust-based authentication (no password, only username).

rlogin -l <username> <target\_IP>

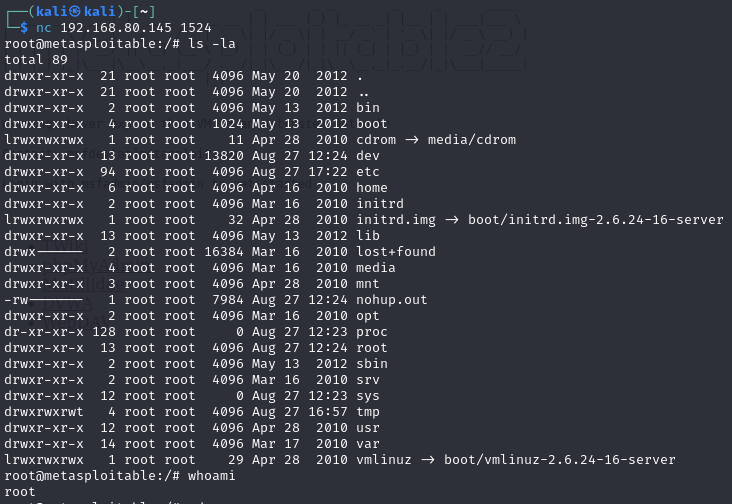
rlogin -l root 192.168.80.145



**9.bindshell(port 1524):**

If it’s a bind shell, interact directly:

nc <target\_IP> 1524



**10.nfs(Network File System)(port 2049):**

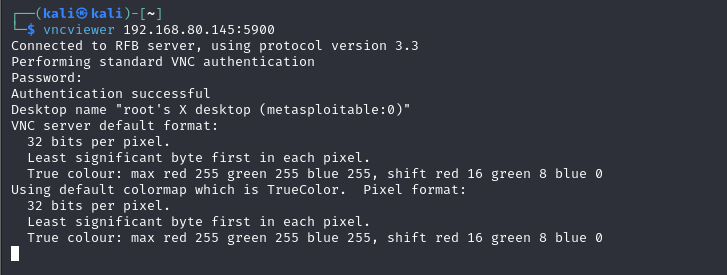
Tried hydra on nfs

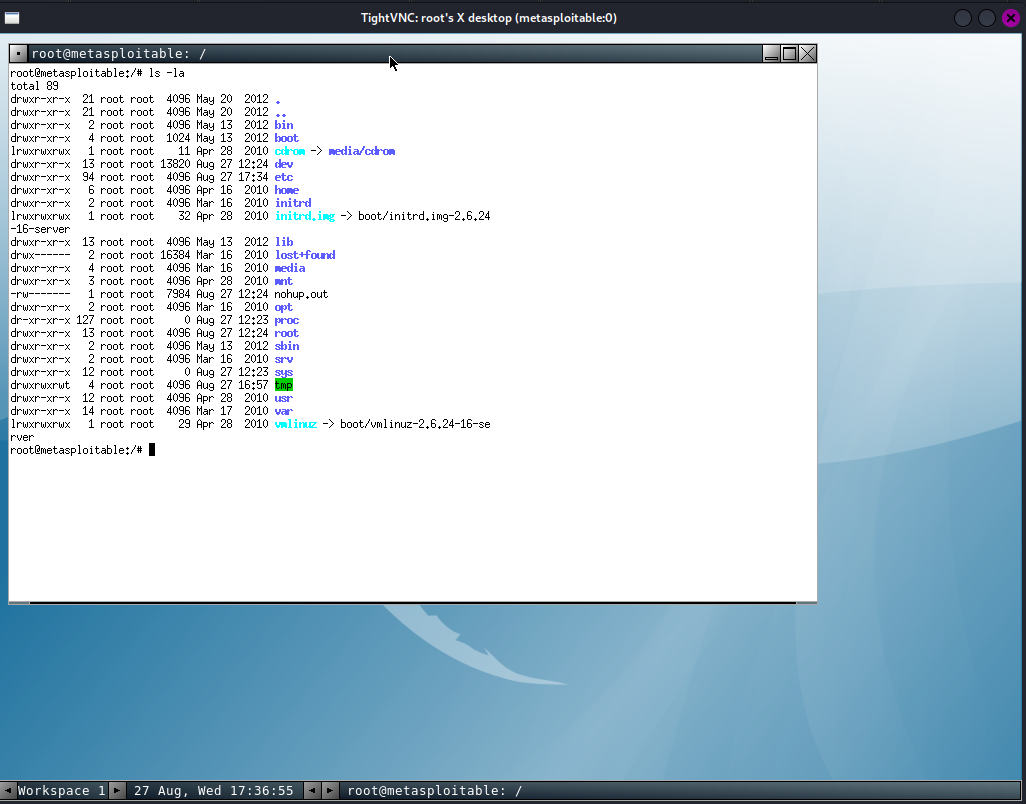
****

**11.VNC(Virtual Network Computing)(port 5900):**

VNC (Virtual Network Computing) is a remote desktop protocol that allows users to control another computer’s graphical interface over a network.

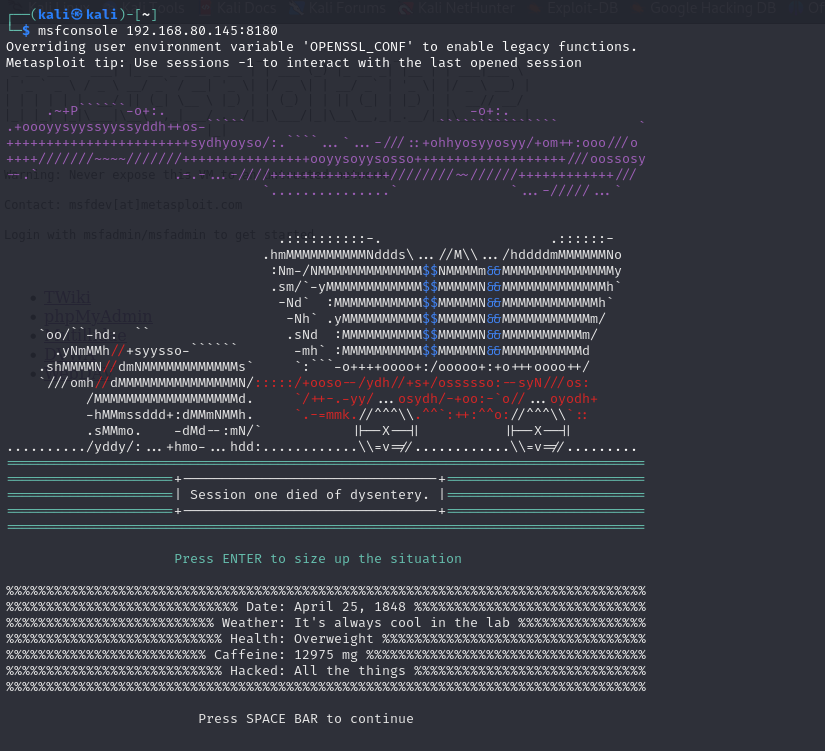
vncviewer ip:port

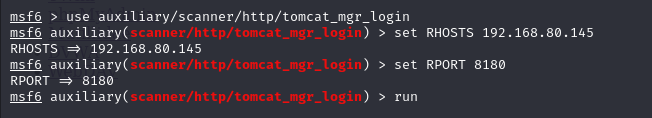
****

****

**12.http(port 8180):**

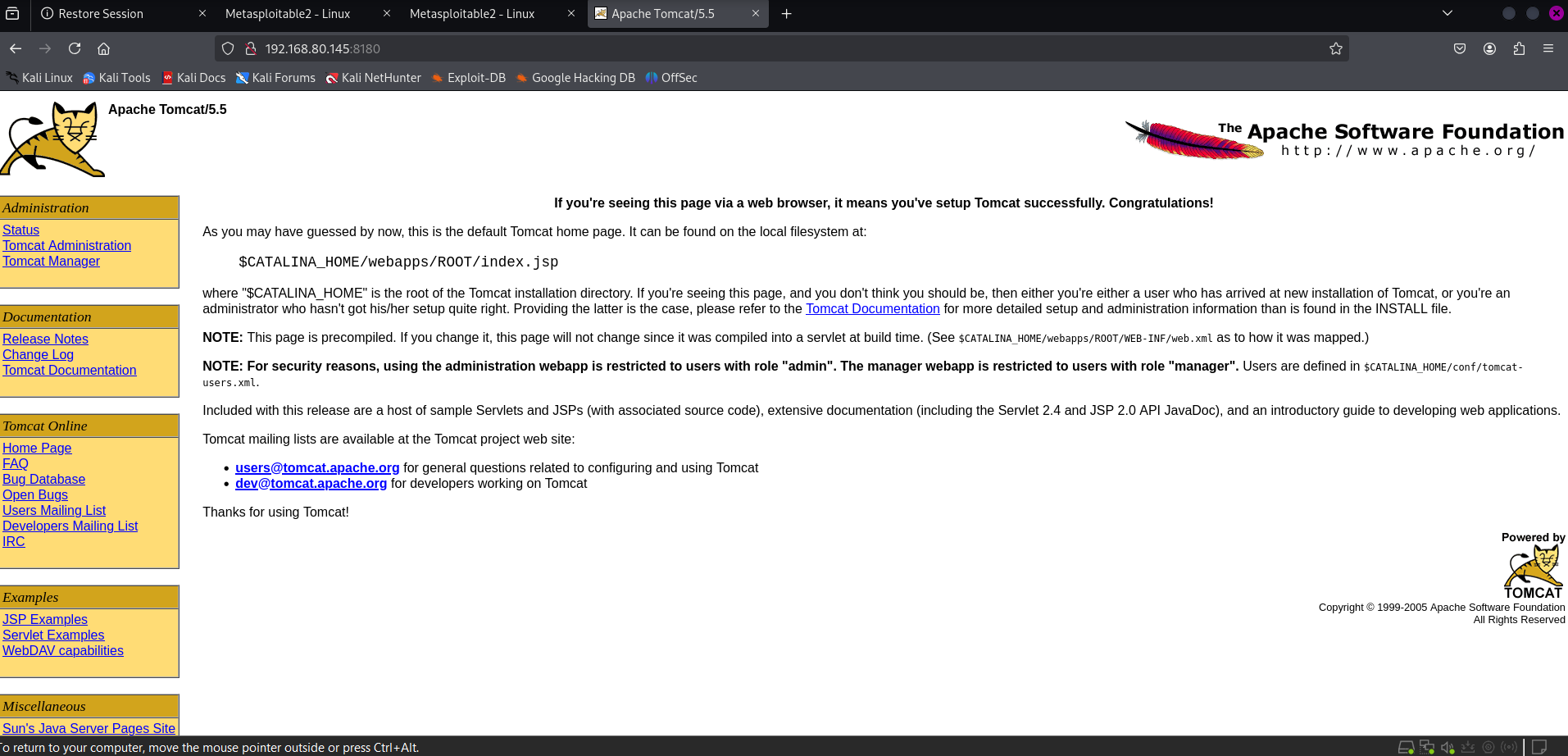
****

****

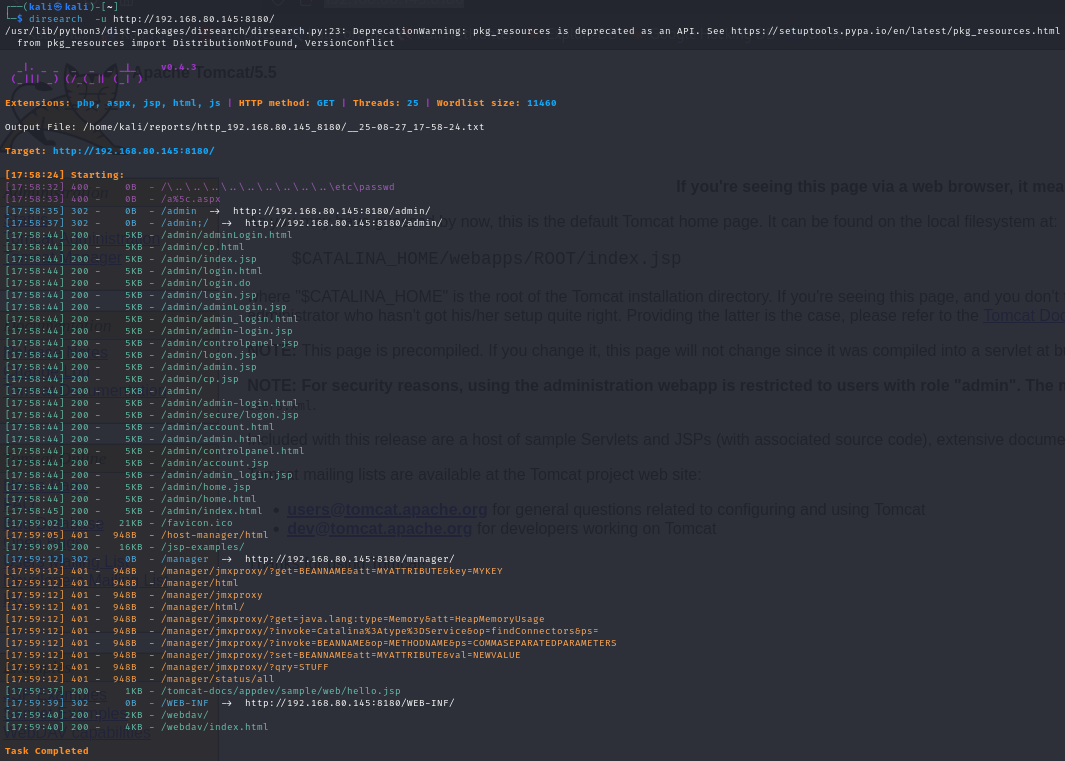
****

****

**Opened in browser**



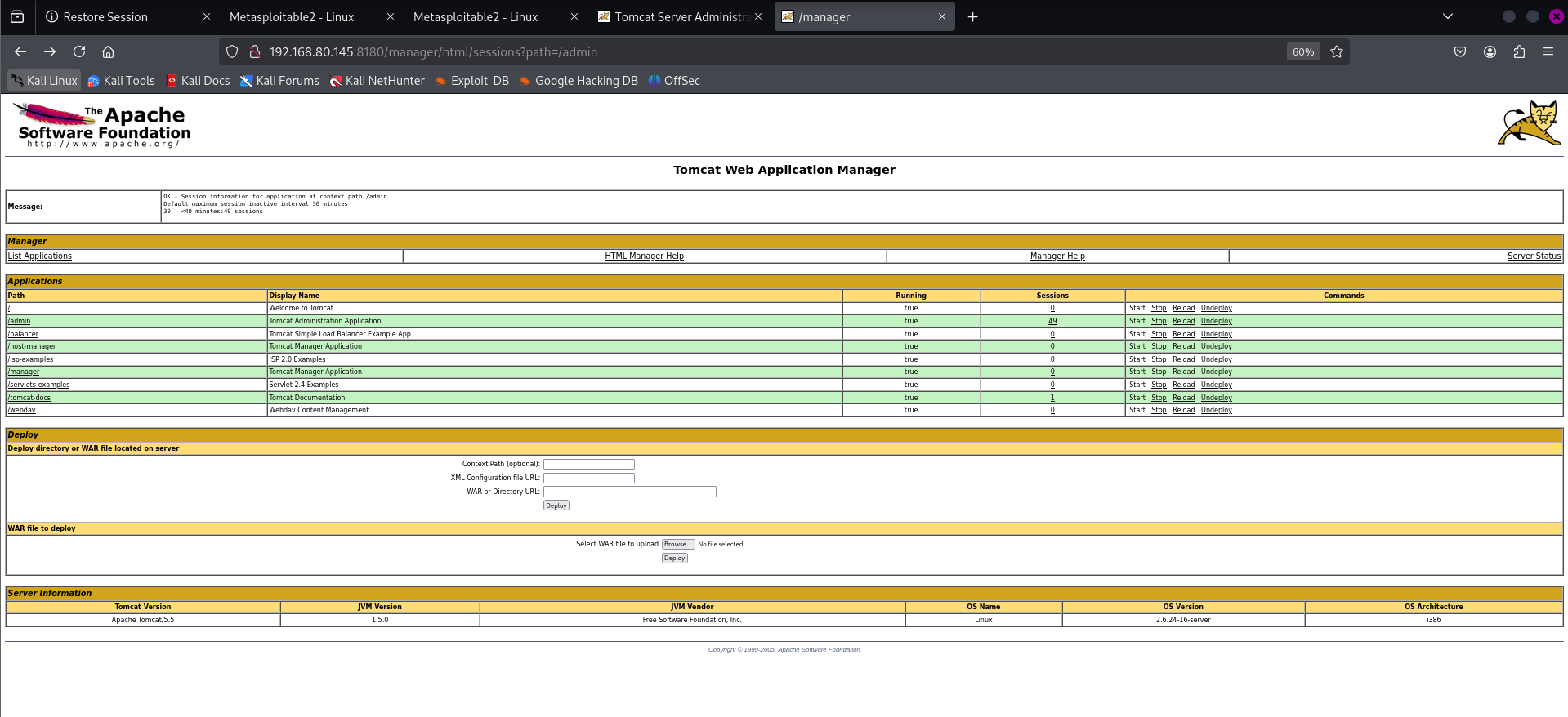
I applied dirsearch on tomcat I got:

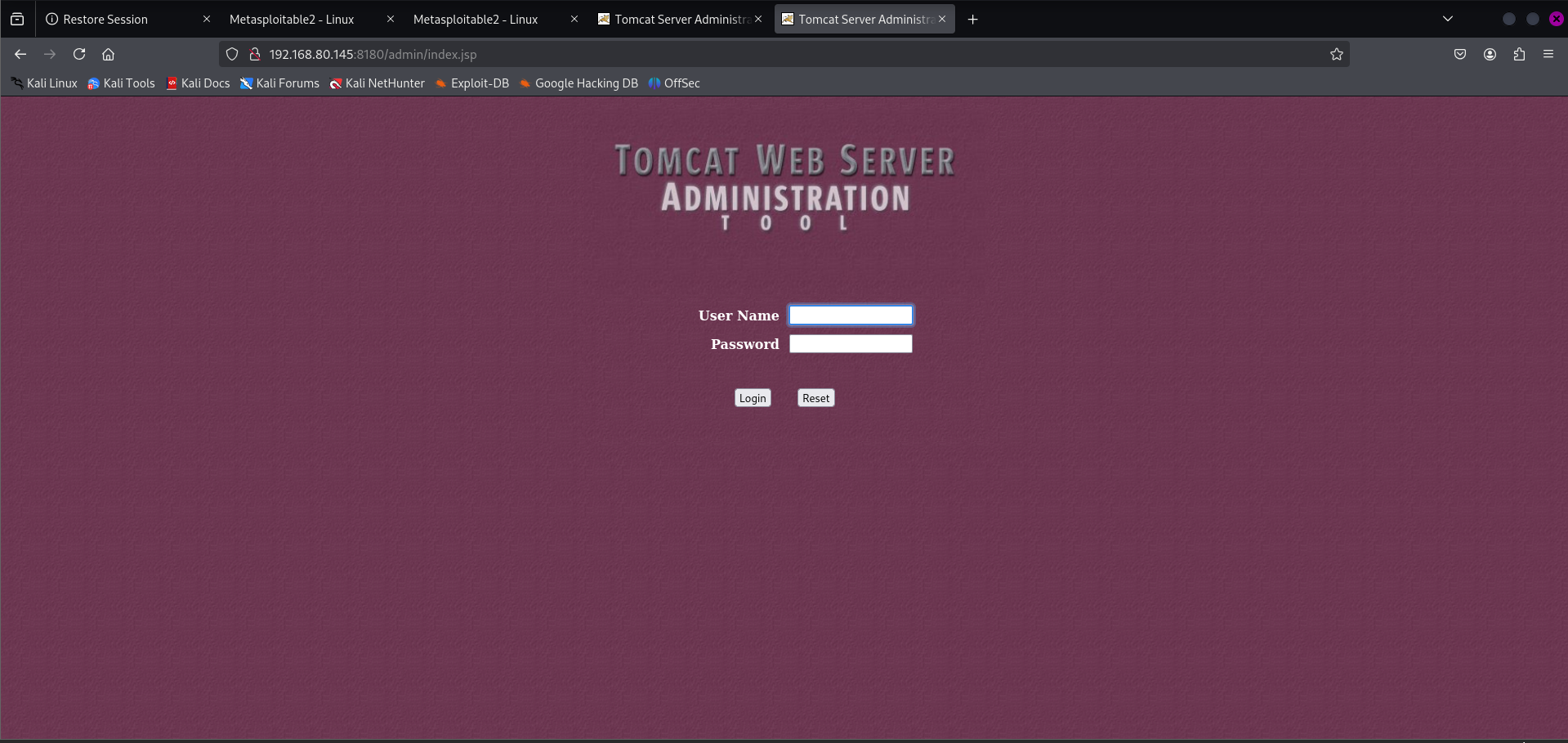


Then in browser went to “<http://192.168.80.145:8180/admin>”:

Then went to :” <http://192.168.80.145:8180/manager/html/sessions?path=/admin>”

Then I am a manager. Here there is war file upload input here we can upload war file to deploy.

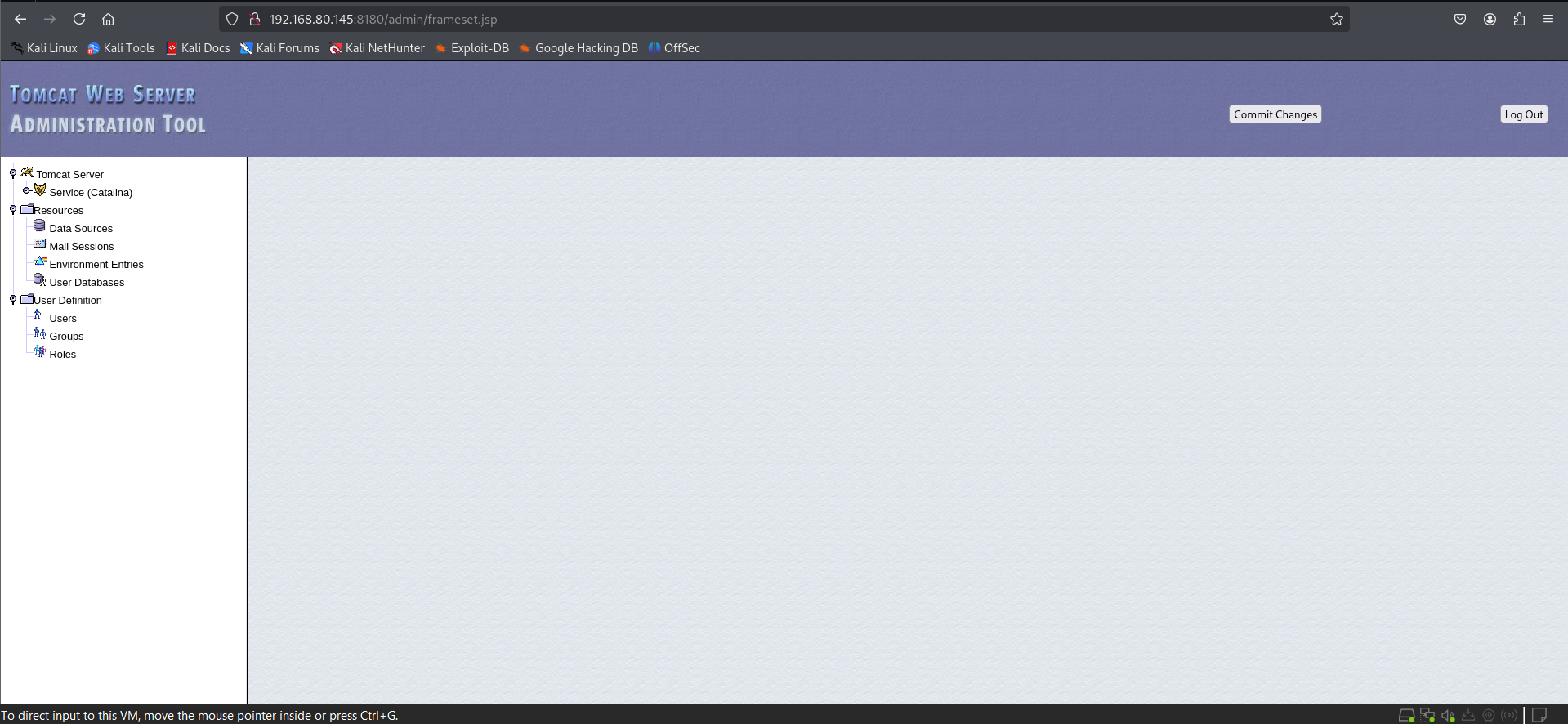




Username:tomcat

Password:tomcat

Now I am logged in as admin. After login I directed to “<http://192.168.80.145:8180/admin/frameset.jsp>”



From server status from manager page I got:



My ip as history.

**13.drb(port 8787):**

The **DRb service** stands for **Distributed Ruby**. It is a **Ruby library and service that allows Ruby programs to communicate with each other over a network**. Essentially, it lets one Ruby program call methods on objects in another Ruby program **as if they were local objects**.

**14.nlockmgr(port 52081):**

The **nlockmgr service** stands for **Network Lock Manager**, which is part of the **NFS (Network File System) infrastructure**. Its main purpose is to **coordinate file locks across a network**, so that multiple clients can safely access shared files without conflicts.