

## Lab3

### 1. Launch a jump host

- Take a screenshot while you are ssh to the jumphost
- Also When you ssh from bastion to the private machine
- Screenshot from the console showing the instances Ips

### 2. Implement the below diagram then

- take a screenshot while you put dns of the load balancer into the browser and it returns a response from the 2 apache instatneces
- Also Screenshot from the console showing the machines BE WS have private Ips
- And finally a screenshot showing the targets of the 2 loadbalancers target groups are healthy

## 1. Launch a jump host

- Take a screen while you are ssh to the jumphost
- Also When you ssh from bastion to the private machine

```
AWS
AWS x Ansible x Centos 3666 x Ubuntu 3888 x Red 3777 x
$$ #####
$$ ## SSH to the public instance
$$ #####
$$
$$ ssh -i "aws_key.pem" ec2-user@107.22.74.106
Register this system with Red Hat Insights: insights-client --register
Create an account or view all your systems at https://red.ht/insights-dashboard
Last login: Wed May 10 15:04:03 2023 from 154.191.89.107
[ec2-user@ip-10-0-0-187 ~]$
[ec2-user@ip-10-0-0-187 ~]$ PS1='PUBLIC_EC2: '
PUBLIC_EC2:
PUBLIC_EC2: ## SSH to the private instance (from inside the public one)
PUBLIC_EC2: #####
PUBLIC_EC2:
PUBLIC_EC2: ssh -i "aws_key.pem_scp_copy" ec2-user@10.0.21.77
Register this system with Red Hat Insights: insights-client --register
Create an account or view all your systems at https://red.ht/insights-dashboard
Last login: Wed May 10 15:04:11 2023 from 10.0.0.187
[ec2-user@ip-10-0-21-77 ~]$
[ec2-user@ip-10-0-21-77 ~]$ ## Install Apache Web Server
[ec2-user@ip-10-0-21-77 ~]$ sudo yum install httpd -y &> httpd_install_log_file
[ec2-user@ip-10-0-21-77 ~]$ sudo systemctl enable --now httpd
Created symlink /etc/systemd/system/multi-user.target.wants/httpd.service → /usr/lib/sy
stemd/system/httpd.service.
[ec2-user@ip-10-0-21-77 ~]$ sudo systemctl status httpd
● httpd.service - The Apache HTTP Server
   Loaded: loaded (/usr/lib/systemd/system/httpd.service; enabled; vendor preset: di
   Active: active (running) since Wed 2023-05-10 15:09:43 UTC; 9s ago
     Docs: man:httpd.service(8)
   Main PID: 11225 (httpd)
    Status: "Total requests: 0; Idle/Busy workers 100/0;Requests/sec: 0; Bytes served>
      Tasks: 213 (limit: 5704)
     Memory: 23.4M
        CPU: 71ms
    CGroup: /system.slice/httpd.service
           └─11225 /usr/sbin/httpd -DFOREGROUND
[ec2-user@ip-10-0-21-77 ~]$ exit
logout
Connection to 10.0.21.77 closed.
PUBLIC_EC2: exit
logout
Connection to 107.22.74.106 closed.
$$
```

## ○ Screenshot from the console showing the instances Ips

**Instances (1/2) Info**

Find instance by attribute or tag (case-sensitive)

Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone
lab1,2_public_instance	i-08c78bb809e080a06	Running	t2.micro	2/2 checks passed	No alarms	us-east-1a
lab1,2_private_instance	i-0a2acae25eb46f076	Running	t2.micro	2/2 checks passed	No alarms	us-east-1a

**Instance: i-0a2acae25eb46f076 (lab1,2\_private\_instance)**

**Details** | Security | Networking | Storage | Status checks | Monitoring | Tags

**Instance summary Info**

Instance ID	Public IPv4 address	Private IPv4 addresses
i-0a2acae25eb46f076 (lab1,2_private_instance)	-	10.0.21.77
IPv6 address	Instance state	Public IPv4 DNS
-	Running	-

**Addition: using user data section to enter an initialization script.**

### User data - optional Info

Enter user data in the field.

```
#!/bin/bash

yum install nginx -y &> $HOME/nginxInstallLog

systemctl enable --now nginx

setsebool -P httpd_can_network_connect 1
```

**EC2 Management Console and 53 more pages - Personal - Microsoft Edge Dev**

**Instances (1/5) Info**

Find instance by attribute or tag (case-sensitive)

Name	Instance ID
lab3-public-ec2-B	i-0999d35e5189caccd

**Instance: i-0999d35e5189caccd (lab3-public-ec2-B)**

**Details** | Security | Networking | Storage | Status checks

**Instance summary Info**

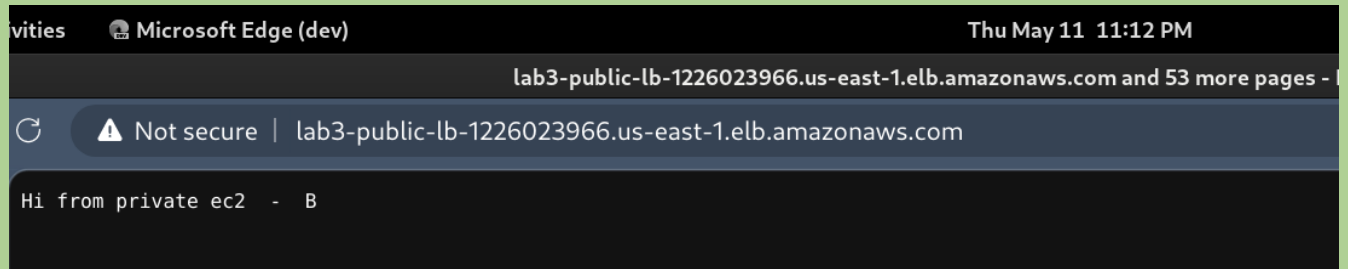
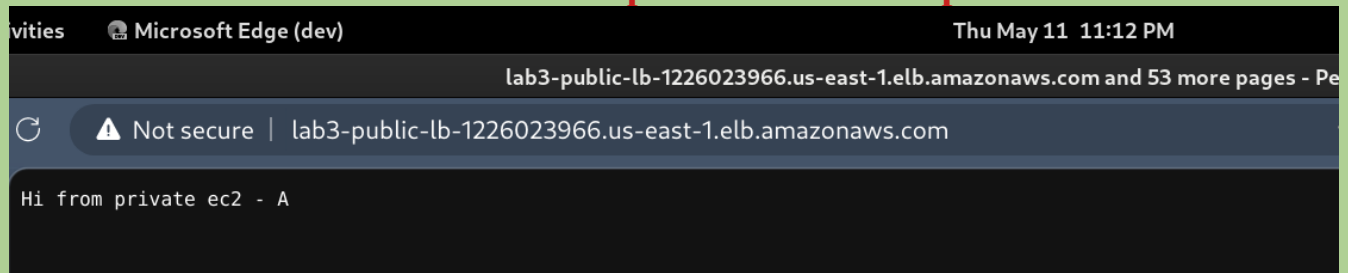
Instance ID	Public IPv4 address
i-0999d35e5189caccd (lab3-public-ec2-B)	3.90.1.211   open address
Private IPv4 addresses	IPv6 address
10.10.2.167	-
Instance state	Public IPv4 DNS
Pending	ec2-3-90-1-211.compute-1.amazonaws.com   open address
Hostname type	Private IP DNS name (IPv4 only)

**Terminal**

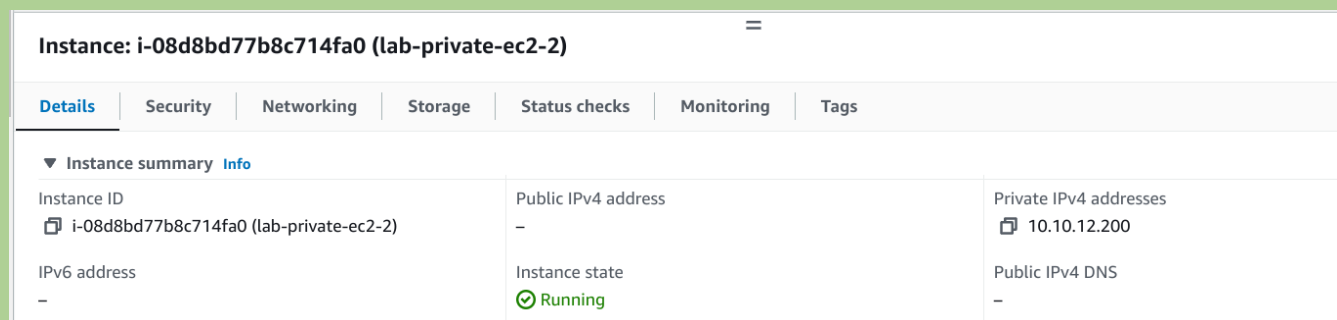
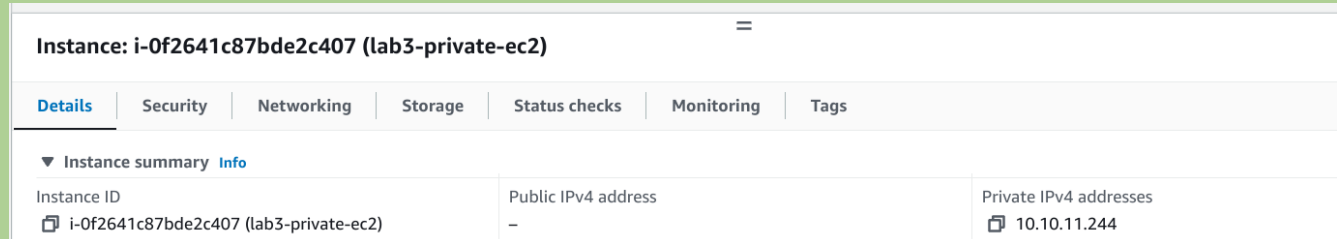
```
ec2-user@ip-10-10-2-167:~$ ssh -i "aws key.pem" ec2-user@3.90.1.211
The authenticity of host '3.90.1.211 (3.90.1.211)' can't be established.
ED25519 key fingerprint is SHA256:eTK0ANKN0wK1K5rd3fveFqSBe4rXD0RbC88HG7GeTOI.
This key is not known by any other names
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '3.90.1.211' (ED25519) to the list of known hosts.
Register this system with Red Hat Insights: insights-client --register
Create an account or view all your systems at https://red.ht/insights-dashboard
[ec2-user@ip-10-10-2-167 ~]$ ls
[ec2-user@ip-10-10-2-167 ~]$ ls /root/
ls: cannot open directory '/root/': Permission denied
[ec2-user@ip-10-10-2-167 ~]$ sudo ls /root
[ec2-user@ip-10-10-2-167 ~]$ ls /home/
[ec2-user@ip-10-10-2-167 ~]$ systemctl status nginx
nginx.service - The nginx HTTP and reverse proxy server
Loaded: loaded (/usr/lib/systemd/system/nginx.service; enabled; vendor preset: disabled)
Active: active (running) since Thu 2023-05-11 21:27:00 UTC; 1min 2s ago
Process: 3139 ExecStartPre=/usr/bin/rm -f /run/nginx.pid (code=exited, status=0/SUCCESS)
lines 1-4...skipping...
nginx.service - The nginx HTTP and reverse proxy server
Loaded: loaded (/usr/lib/systemd/system/nginx.service; enabled; vendor preset: disabled)
Active: active (running) since Thu 2023-05-11 21:27:00 UTC; 1min 2s ago
Process: 3139 ExecStartPre=/usr/bin/rm -f /run/nginx.pid (code=exited, status=0/SUCCESS)
Process: 3148 ExecStartPre=/usr/sbin/nginx -t (code=exited, status=0/SUCCESS)
lines 1-5...skipping...
```

## 2. Implement the below diagram then

- take a screenshot while you put dns of the load balancer into the browser and it returns a response from the 2 apache instanctnes



- Also Screenshot from the console showing the machines BE WS have private Ips



- And finally a screenshot showing the targets of the 2 loadbalancers target groups are healthy

EC2 &gt; Target groups &gt; lab3-public-tg

## lab3-public-tg

Actions ▼

## Details

 arn:aws:elasticloadbalancing:us-east-1:385582076770:targetgroup/lab3-public-tg/baed87826be8a4ae

Target type

Instance

Protocol : Port

HTTP: 80

Protocol version

HTTP1

VPC

[vpc-0d7292c43121b34fd](#) 

IP address type

IPv4

Load balancer

[lab3-public-lb](#) 

Total targets

2

Healthy

 2

Unhealthy

 0

Unused

 0

Initial

 0

Draining

 0

## ► Distribution of targets by Availability Zone (AZ)

Select values in this table to see corresponding filters applied to the Registered targets table below.

## lab3-private-tg

## Details

 arn:aws:elasticloadbalancing:us-east-1:385582076770:targetgroup/lab3-private-tg/ea54fc21ba89fb71

Target type

Instance

Protocol : Port

HTTP: 80

Protocol version

HTTP1

IP address type

IPv4

Load balancer

[lab3-private-lb](#) 

Total targets

2

Healthy

 2

Unhealthy

 0

Unused

 0

## ► Distribution of targets by Availability Zone (AZ)

Select values in this table to see corresponding filters applied to the Registered targets table below.

Targets

Monitoring



Health checks

Attributes

Tags

## Registered targets (2)

 Filter resources by property or value

<input type="checkbox"/>	Instance ID ▼	Name ▼	Port ▼	Zone ▼	Health status ▼	Health status details
<input type="checkbox"/>	<a href="#">i-08d8bd77b8c714fa0</a>	lab-private-ec2-2	80	us-east-1d	 healthy	
<input type="checkbox"/>	<a href="#">i-0f2641c87bde2c407</a>	lab3-private-ec2	80	us-east-1c	 healthy	