CLOUD CONCEPTS

(EC2 Elastic IP and Apache Server on Amazon Linux 2 AMI)

Elastic IP

- With an Elastic IP address, you can mask the failure of an instance or software by rapidly remapping the address to another instance in your account.
- You can only have 5 Elastic IP in your account (you can ask AWS to increase that).
- Overall, try to avoid using Elastic IP:
  - They often reflect poor architectural decisions
  - Instead, use a random public IP and register a DNS name to it

A Load balancer can be used to avoid usage of Public IP.

By default, EC2 comes with Private IP for internal AWS network and Public IP for www.

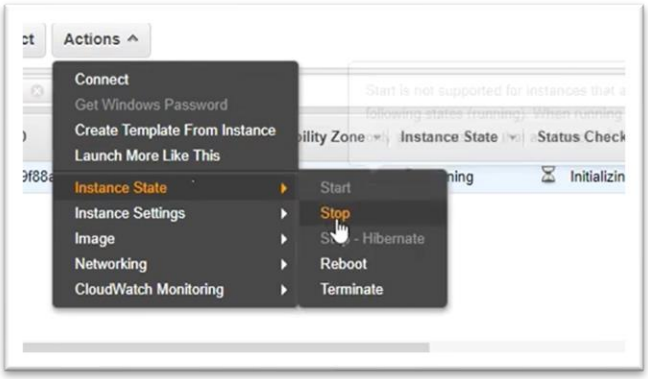
While using SSH to access the server Public IP is used not private IP because we are not on the same network.

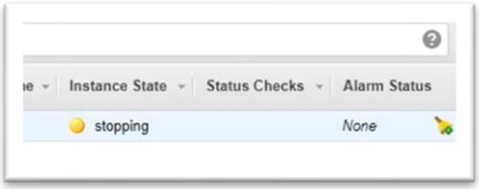
Public IP changes when machine reboots.

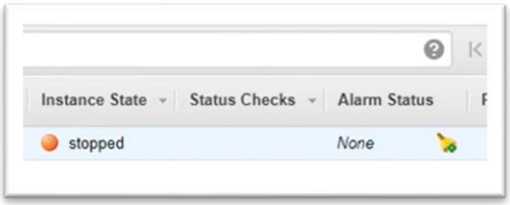
LAB (Elastic IP)

1. Launch the instance or see the IP of the running server. Your server must be connected through your local machine using command line / terminal /power shell or putty (windows only).
2. See the public IP address and save it for references.

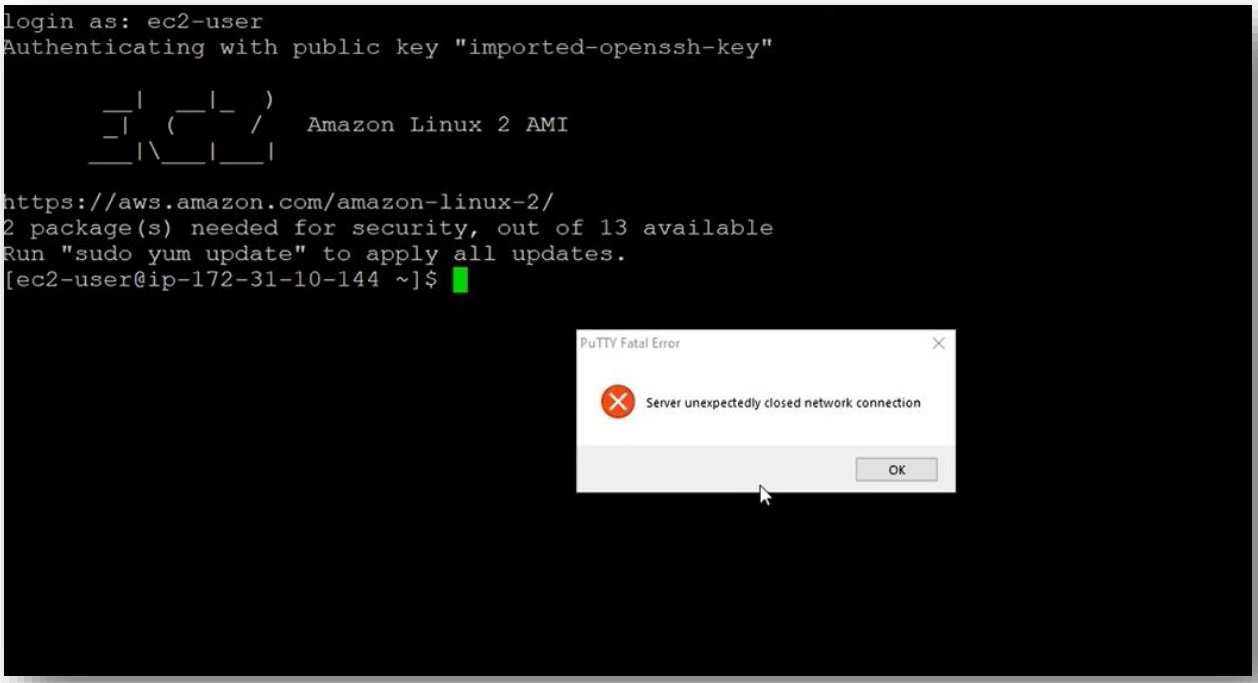
3. Stop the running server and restart it. See the IP address now, it will be changed.



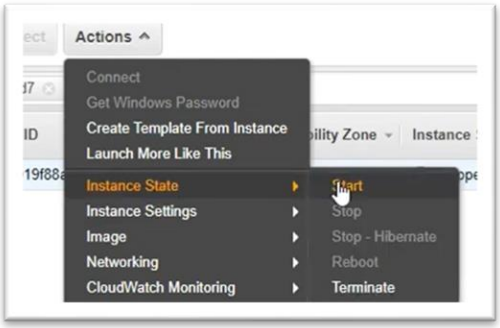


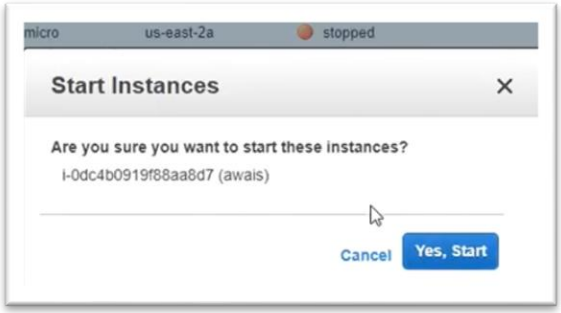


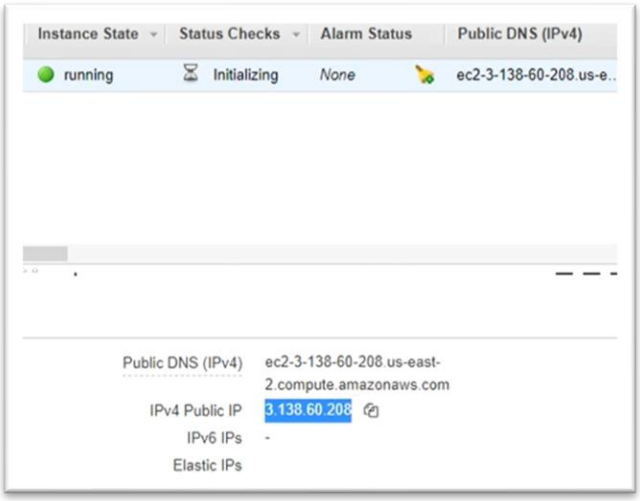
4. When the server is stopped your connection with remote server is disturbed.



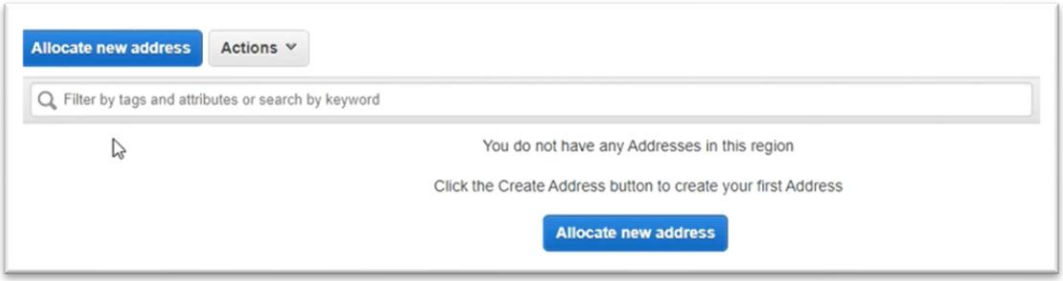
5. Restart the server and see the IP. Public IP is changed now. You must start a new session with new allocated IP.

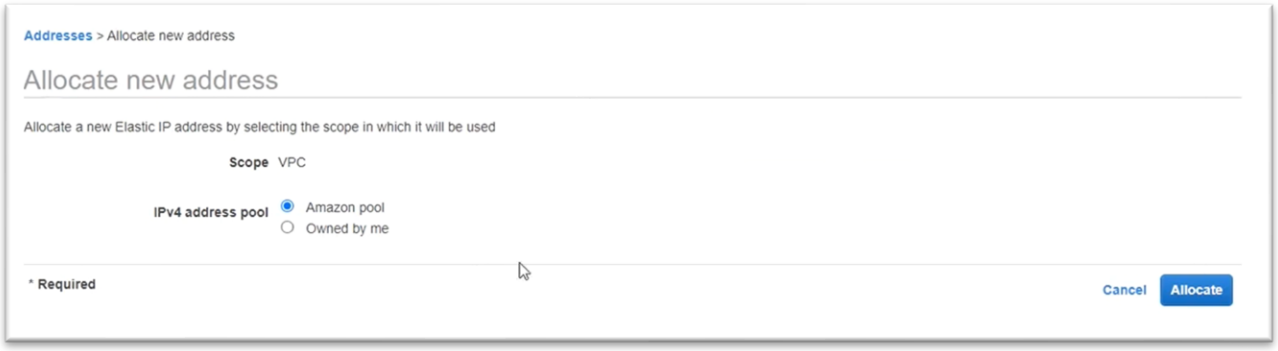




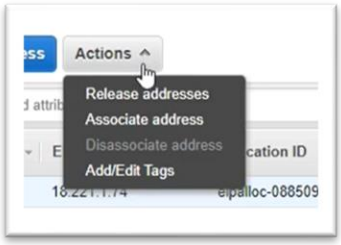


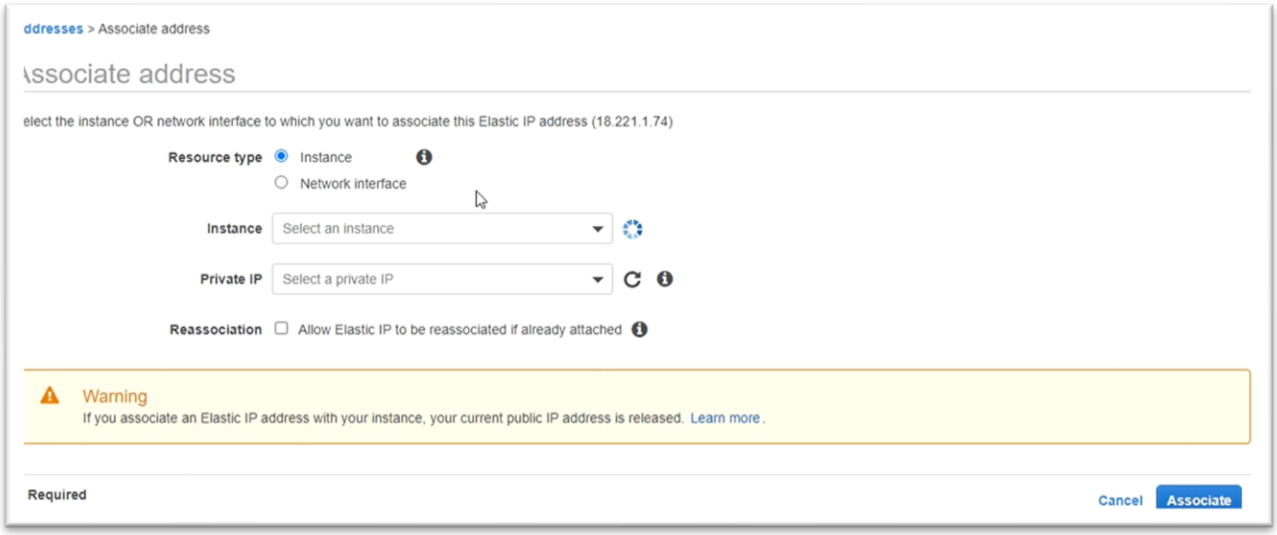
6. Go to Elastic IPs and click on “allocate new address”. Next click “allocate” and close.

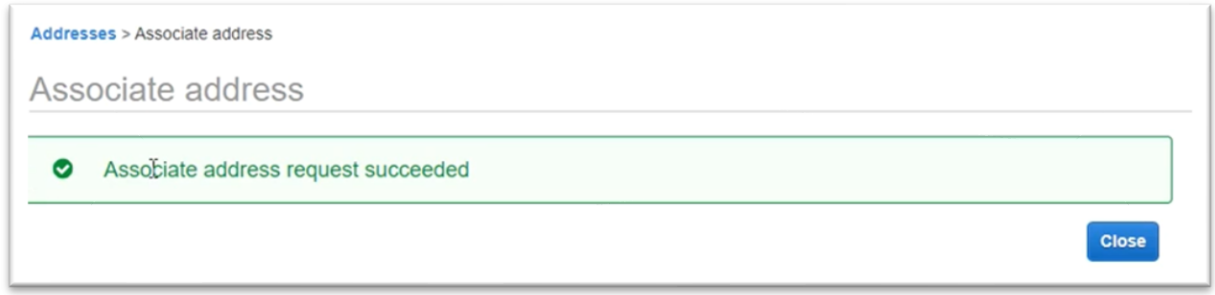




7. You must associate the reserved Elastic IP address with your instance, else billing will be generated. Set the name for EP and select the instance to which this IP is to be associated.
8. Repeat the steps of rebooting the instance and see if the Public IP changes.







DONOT FORGET to detach and release the EP from the server, it will be charged.

LAB (Install Apache Web Server on Linux 2 AMI)

We'll install an Apache Web Server to display a web page We'll create an index.html that shows the hostname of our machine

- **Boot strapping** means launching commands when a machine starts. The script will run once when the instance starts, if there is any error in user data while boot strapping, server has to be launched again.
- **Boot strap script is predefined script in user data.**

It is possible to bootstrap our instances using an EC2 User data script.

EC2 user data is used to automate boot tasks such as:

- Installing updates
- Installing software
- Downloading common files from the internet
- Anything you can think of

The EC2 User Data Script runs with the root user

1. Start the new SSH session and access the running server.

```
Run "sudo yum update" to apply all updates.
[ec2-user@ip-172-31-10-144 ~]$ sudo su -
[root@ip-172-31-10-144 ~]# yum update -y
```

The default username is ec2-user.

2. Use the following command to install the Apache server.

```
[root@ip-172-31-10-144 ~]#
[root@ip-172-31-10-144 ~]# yum install httpd.x86_64 -y
```

httpd is Linux amazon Apache server.

```
)ependency Installed:
  apr.x86_64 0:1.6.3-5.amzn2.0.2                    apr-util.x86_64 0:1.6.1-5.
  apr-util-bdb.x86_64 0:1.6.1-5.amzn2.0.2           generic-logos-httpd.noarch
  httpd-filesystem.noarch 0:2.4.46-1.amzn2          httpd-tools.x86_64 0:2.4.4
  mailcap.noarch 0:2.1.41-2.amzn2                   mod_http2.x86_64 0:1.15.14

:omplete!
[root@ip-172-31-10-144 ~]#
```

3. To run Apache, enter the following command.

```
[root@ip-172-31-10-144 ~]#
[root@ip-172-31-10-144 ~]#
[root@ip-172-31-10-144 ~]# systemctl start httpd.service
```

4. To configure the server, write the following command.

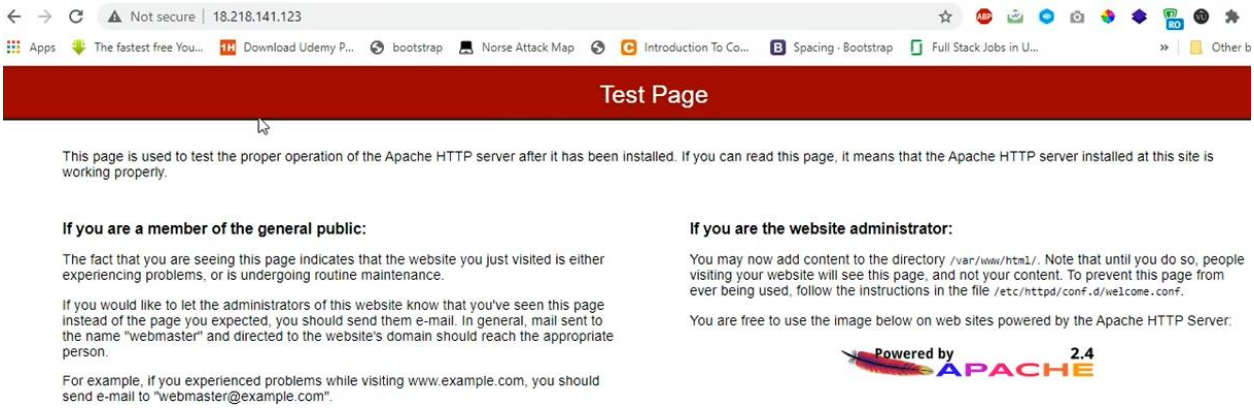
```
[root@ip-172-31-10-144 ~]# systemctl enable httpd.service
Created symlink from /etc/systemd/system/multi-user.target.wants/httpd.service to /usr/lib/syst
em/httpd.service.
[root@ip-172-31-10-144 ~]#
```

5. Default Apache page will be displayed (html script).

```
[root@ip-172-31-10-144 ~]# curl localhost:80
```

```
                                    <p>For example, if you experienced problems while visiting www.
example.com, you should send e-mail to "webmaster@example.com".</p>

                                    <hr />
                                </div>

                                <div class="content-column-right">
                                    <h2>If you are the website administrator:</h2>

                                    <p>You may now add content to the directory <tt>/var/www/html/<
/tt>. Note that until you do so, people visiting your website will see this page, and not your content.
 To prevent this page from ever being used, follow the instructions in the file <tt>/etc/httpd/conf.d/w
elcome.conf</tt>.</p>

                                    <p>You are free to use the image below on web sites powered by
the Apache HTTP Server:</p>

                                    <p align="center"><a href="http://httpd.apache.org/"><img src="
/icons/apache_pb2.gif" alt="[ Powered by Apache ]"/></a></p>

                                </div>
                            </div>
                        </div>
                    </body>
</html>
```

6. Use the Public IP browser to view default Apache page.



9. To see update the page write the command echo *"ANY TEXT HERE"* > /var/www/html/index.html and refresh the page to see the changes.

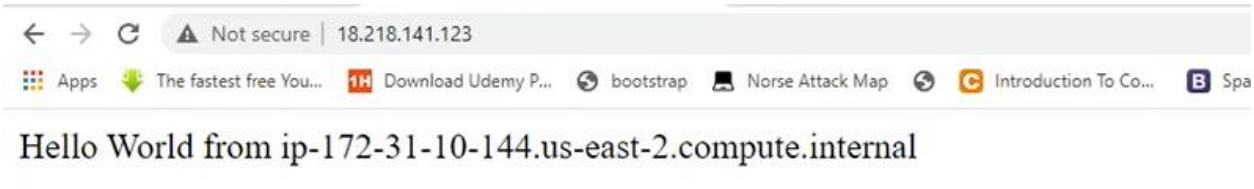
Echo command is to write/display.

> is to write/display text to file at location /var/www/html/index.html.

10. Print the server's name of machine using the following command and refresh the page to see the text on Apache web page.

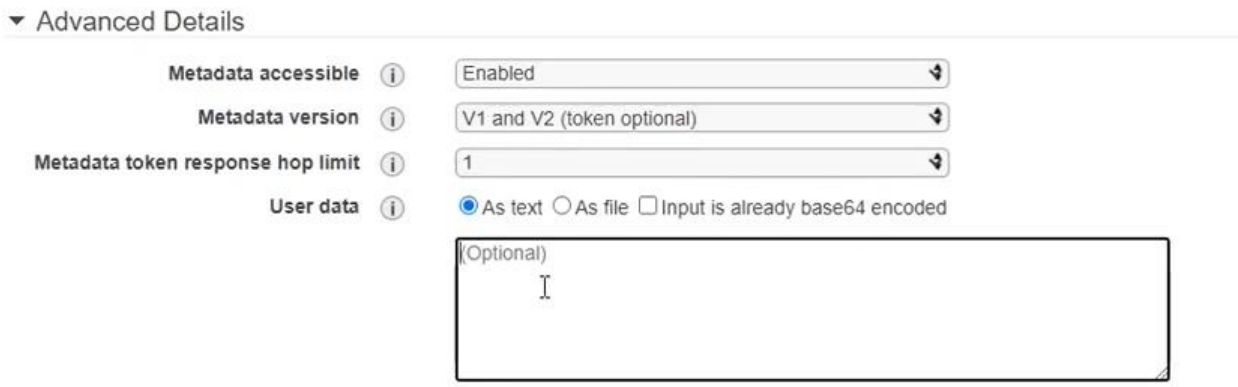
In $(hostname -f), $ is used to store variables. `-f` Option in `hostname` command in Linux display the fully qualified domain name (FQDN) of the host.

```
[root@ip-172-31-10-144 ~]#
[root@ip-172-31-10-144 ~]# echo "Hello World from $(hostname -f)" > /var/www/html/index.html
[root@ip-172-31-10-144 ~]#
```



11. The steps given above are manual, this process can be automated through bash scripting. Bash script will be executed whenever server is launched. To set user data go to advance setting while

configuring the server and paste the Bash script or attach the file.




```
#!/bin/bash

############################################################
##### USE THIS FILE IF YOU LAUNCHED AMAZON LINUX 2 #####
############################################################

# get admin privileges
sudo su

# install httpd (Linux 2 version)
yum update -y
yum install -y httpd.x86_64
systemctl start httpd.service
systemctl enable httpd.service
echo "Hello World from $(hostname -f)" > /var/www/html/index.html
```

Delete the comments in bash script when paste.

12. Tag if multiple servers are running to identify.
13. Create a security group and make sure port 80 is ON and port 22 is optional.

14. After launching the server copy the public IP and browse the page on the browser.