

LETTERKENNY INSTITUTE OF TECHNOLOGY**ASSIGNMENT COVER SHEET**

Lecturer's Name: Ruth Lennon

Assessment Title: Scripting The Deploy Pipeline

Work to be submitted to: Ruth Lennon

Date for submission of work: 20/03/2022

Place and time for submitting work: Blackboard as per submission link

To be completed by the Student

Student's Name: Luis Gonzalez

Class: Scripting The Deploy Pipeline

Subject/Module: Scripting The Deploy Pipeline

Word Count (where applicable): N/A

I confirm that the work submitted has been produced solely through my own efforts.

Student's signature: Luis Gonzalez **Date:** 20/03/2022

Notes

Penalties: The total marks available for an assessment is reduced by 15% for work submitted up to one week late. The total marks available are reduced by 30% for work up to two weeks late. Assessment work received more than two weeks late will receive a mark of zero. [Incidents of alleged plagiarism and cheating are dealt with in accordance with the Institute's Assessment Regulations.]

Plagiarism: Presenting the ideas etc. of someone else without proper acknowledgement (see section L1 paragraph 8).

Cheating: The use of unauthorised material in a test, exam etc., unauthorised access to test matter, unauthorised collusion, dishonest behaviour in respect of assessments, and deliberate plagiarism (see section L1 paragraph 8).

Continuous Assessment: For students repeating an examination, marks awarded for continuous assessment, shall normally be carried forward from the original examination to the repeat examination.

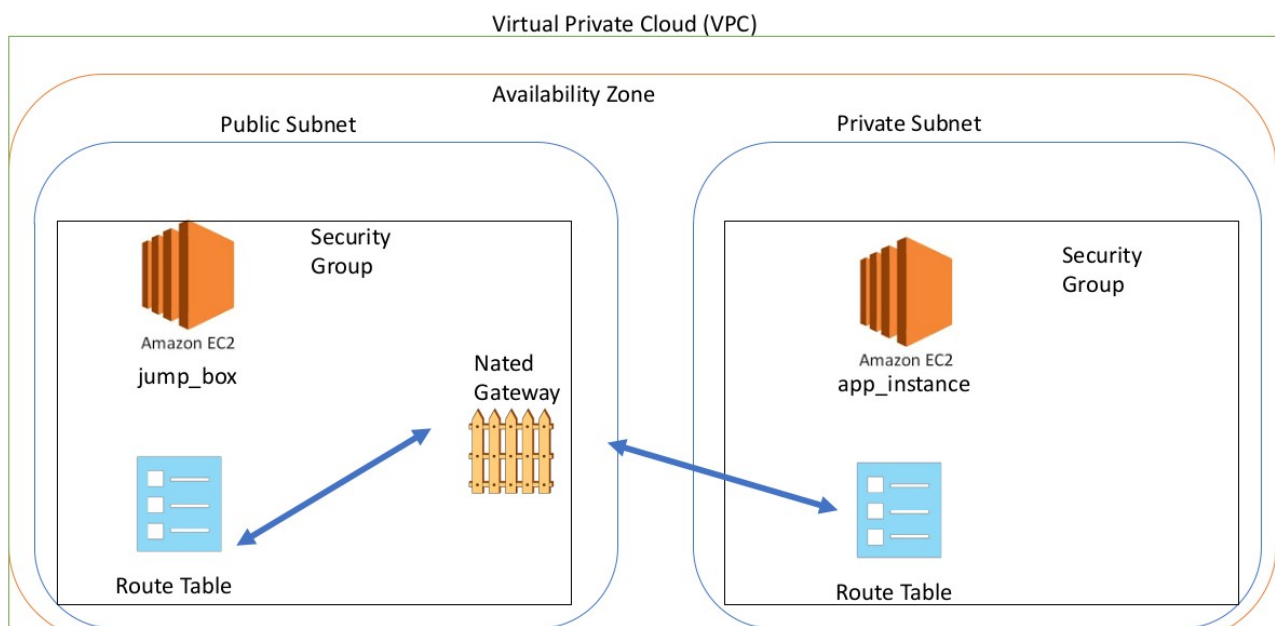
Aims/Description

As per Assignment Question:

Create a Cloud Formation Stack using the designer.

The purpose is to replicate the mini network created previously.

Terraform Demo



Results

1. AWS CloudFormation template was created in Designer
2. Virtual Private Cloud was configured and template was validated
3. Stack was created and instances ran successfully.
4. User was able to remote to public instance by ssh and using key.
5. Within Public Subnet user was able to access Private Subnet.
6. Current file was uploaded to Student's GitHub Repository into CloudFormation
<https://github.com/L00170299/ScriptingTheDeployPipeline>

Conclusions

Student was able to learn and understand basics of CloudFormation in AWS. Template wasn't working at first and there were multiples attempts and failures but each time something new was discovered or understood. Finally pieces started to make sense and with some extra research about how to do small mapping between elements the VPC started to work.

Student tried to follow best practices like add descriptions, tags, avoid security within code, etc.

The idea of implementing a VPC is to protect our infrastructure from possible attacks by hiding it and allow just one instance to be accessed from internet and therefore harden the security in that single instance point.

In this example student used KeyPair file as security measure to access that instance exposed to internet and allow just an specific range of public IP's to get access. KeyPairName must be a key previously generated in AWS and just Name needs to be referenced when running Stack

Appendix

Loading template

Create stack

Prerequisite - Prepare template

Prepare template
Every stack is based on a template. A template is a JSON or YAML file that contains configuration information about the AWS resources you want to include in the stack.

☒ Template is ready

☐ Use a sample template

☐ Create template in Designer

Specify template


A template is a JSON or YAML file that describes your stack's resources and properties.

Template source
Selecting a template generates an Amazon S3 URL where it will be stored.

☐ Amazon S3 URL

☒ Upload a template file

Upload a template file

Choose file 

L00170299_CloudFormation.yaml

JSON or YAML formatted file

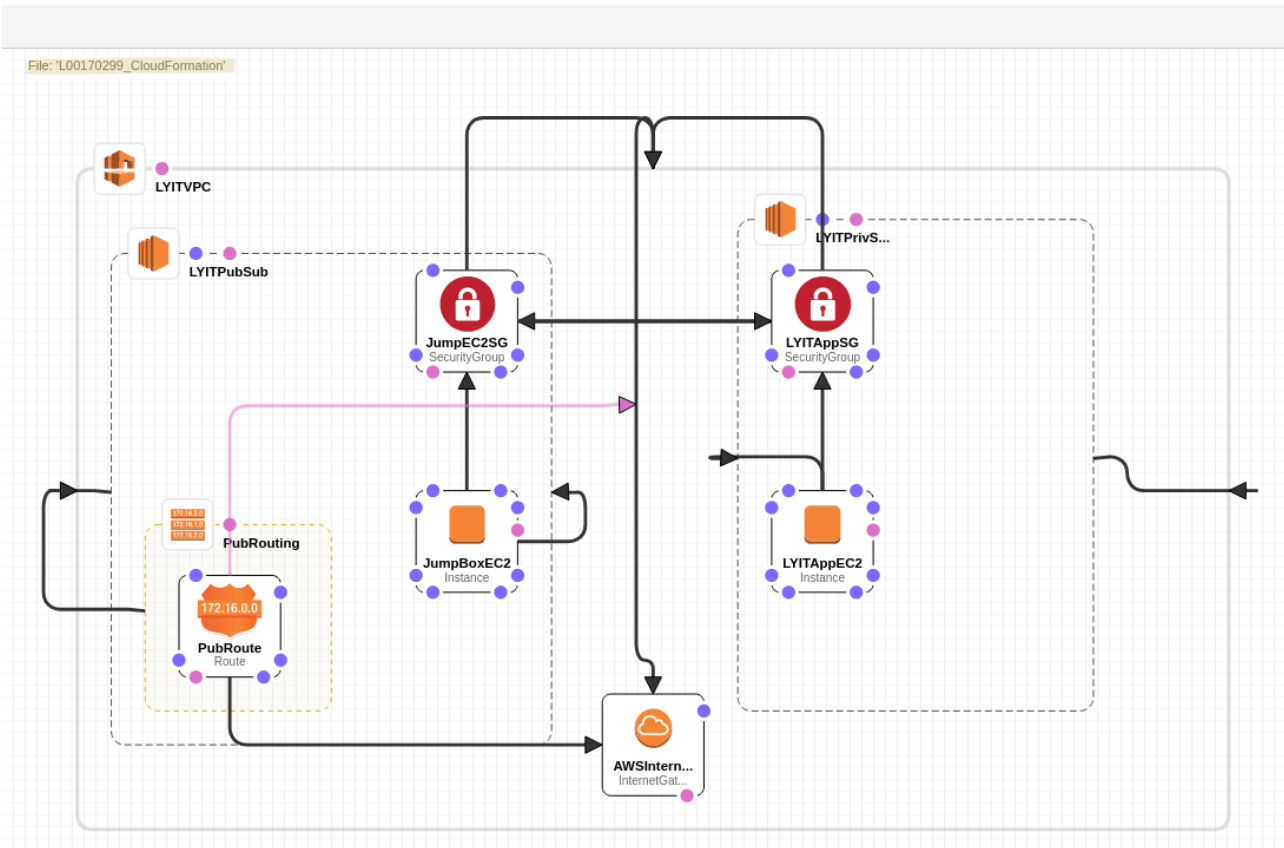
S3 URL: https://s3-eu-west-1.amazonaws.com/cf-templates-1qb6cdt9t6a34-eu-west-1/2022079JBa-L00170299_CloudFormation.yaml

View in Designer

Cancel

Next

VPC diagram from template showing both Subnets



Filling in details before running Stack

Specify stack details

Stack name

Stack name

Stack name can include letters (A-Z and a-z), numbers (0-9), and dashes (-).

Parameters

Parameters are defined in your template and allow you to input custom values when you create or update a stack.

ImageId

InstanceTypeParameter

Enter t2.nano or t2.micro. The default is set as t2.micro

KeyPairName

Enter the name of a keypair name to get access public instance

LYITAdminAddr

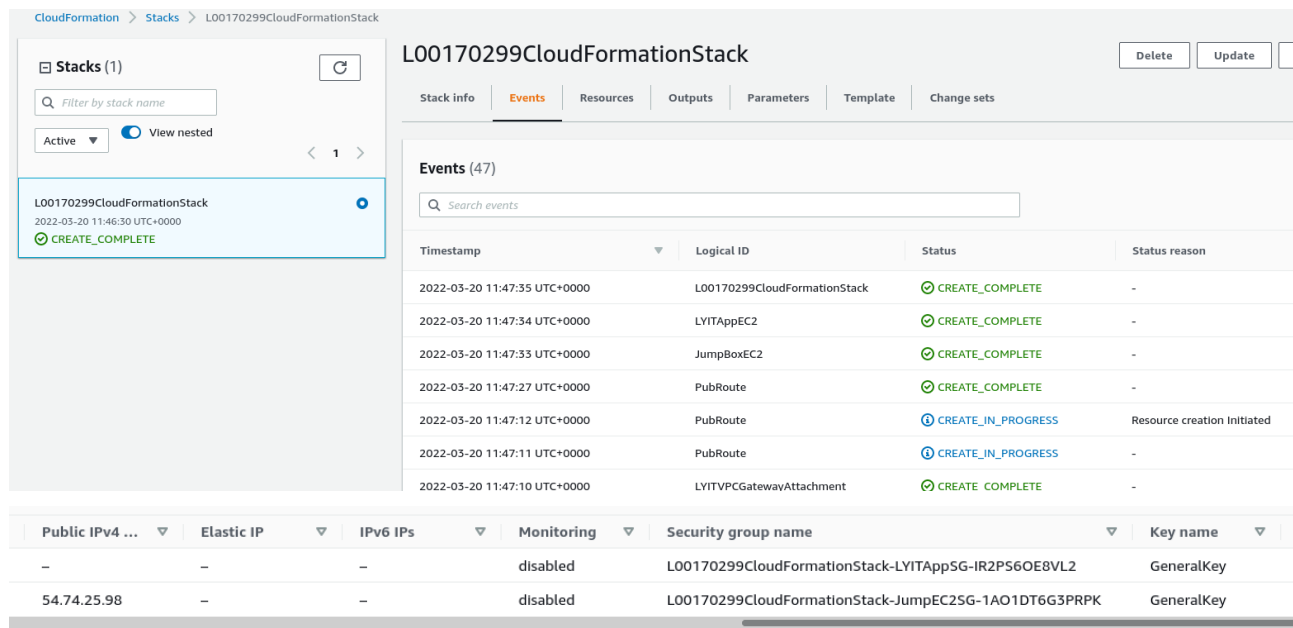
The IP address range that can be used to SSH to the EC2 instances

Cancel

Previous

Next

Stack finished creating VPC and showing both instances



L00170299CloudFormationStack

Stack info | **Events** | Resources | Outputs | Parameters | Template | Change sets

Events (47)

Timestamp	Logical ID	Status	Status reason
2022-03-20 11:47:35 UTC+0000	L00170299CloudFormationStack	CREATE_COMPLETE	-
2022-03-20 11:47:34 UTC+0000	LYITAppEC2	CREATE_COMPLETE	-
2022-03-20 11:47:33 UTC+0000	JumpBoxEC2	CREATE_COMPLETE	-
2022-03-20 11:47:27 UTC+0000	PubRoute	CREATE_COMPLETE	-
2022-03-20 11:47:12 UTC+0000	PubRoute	CREATE_IN_PROGRESS	Resource creation Initiated
2022-03-20 11:47:11 UTC+0000	PubRoute	CREATE_IN_PROGRESS	-
2022-03-20 11:47:10 UTC+0000	LYITVPCGatewayAttachment	CREATE_COMPLETE	-

Public IPv4 ...	Elastic IP	IPv6 IPs	Monitoring	Security group name	Key name
-	-	-	disabled	L00170299CloudFormationStack-LYITAppSG-IR2PS6OE8VL2	GeneralKey
54.74.25.98	-	-	disabled	L00170299CloudFormationStack-JumpEC2SG-1AO1DT6G3PRPK	GeneralKey

Connecting to public IP using KeyFile specified (subnet IP: 192.168.1.80)

```

ec2-user@ip-192-168-1-80:~
[ucabrera@ucabreraos Documents]$ ssh -i "GeneralKey.pem" ec2-user@54.74.25.98
The authenticity of host '54.74.25.98 (54.74.25.98)' can't be established.
ECDSA key fingerprint is SHA256:Bz40ZA/UGDkpZqHZZP4ztTwhHywYXq8KYdzACWAKHr4.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '54.74.25.98' (ECDSA) to the list of known hosts.

  __|  __|_  )
  _| (  /   Amazon Linux 2 AMI
 ---| \___|___|

https://aws.amazon.com/amazon-linux-2/
[ec2-user@ip-192-168-1-80 ~]$ ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 9001
    inet 192.168.1.80 netmask 255.255.255.0 broadcast 192.168.1.255
    inet6 fe80::1:33ff:fe09:6139 prefixlen 64 scopeid 0x20<link>
    ether 02:01:33:09:61:39 txqueuelen 1000 (Ethernet)
    RX packets 552 bytes 472287 (70.5 KiB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 782 bytes 87355 (85.3 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 1000 (Local Loopback)
    RX packets 24 bytes 1944 (1.8 KiB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 24 bytes 1944 (1.8 KiB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

[ec2-user@ip-192-168-1-80 ~]$ ip-192-168-1-80.eu-west-1.compute.internal

```

Accessing private Instance from public instance using same KeyFile (subnet IP: 192.168.0.187)
 **Key needed to be copied first to public one.

```

ec2-user@ip-192-168-0-187:~
[ucabrera@ucabreraos Documents]$ scp -i "GeneralKey.pem" GeneralKey.pem ec2-user@54.74.25.98:/home/ec2-user/
GeneralKey.pem 1dce94284ec01
[ucabrera@ucabreraos Documents]$ ssh -i "GeneralKey.pem" ec2-user@54.74.25.98
Last login: Sun Mar 20 11:53:33 2022 from 176.61.5.87

--|  --|_ )
_| ( / Amazon Linux 2 AMI
---| \---| ---|

Public IPv4 address
~

Instance state
Running

https://aws.amazon.com/amazon-linux-2/
[ec2-user@ip-192-168-1-80 ~]$ ls
GeneralKey.pem
[ec2-user@ip-192-168-1-80 ~]$ ssh -i "GeneralKey.pem" ec2-user@192.168.0.187
The authenticity of host '192.168.0.187 (192.168.0.187)' can't be established.
ECDSA key fingerprint is SHA256:7TW8z9Hon3P+XX+i/ju4HFcmi20ZRwXIKhIah5QD69Y,1.compute.internal
ECDSA key fingerprint is MD5:f4:ad:89:f4:68:39:e7:ba:e3:8a:0e:e3:ec:e4:cf:76.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added '192.168.0.187' (ECDSA) to the list of known hosts.

--|  --|_ )
_| ( / Amazon Linux 2 AMI
---| \---| ---|

IAM Role
~

compute Optimizer for recommendations. | Learn more
https://aws.amazon.com/amazon-linux-2/
[ec2-user@ip-192-168-0-187 ~]$ ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 9001
    inet 192.168.0.187 netmask 255.255.255.0 broadcast 192.168.0.255
    inet6 fe80::5e:14ff:fec6:b24b prefixlen 64 scopeid 0x20<link>
    ether 02:5e:14:c6:b2:4b txqueuelen 1000 (Ethernet)
    RX packets 581 bytes 73718 (71.9 KiB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 796 bytes 87857 (85.7 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 1000 (Local Loopback)
    RX packets 24 bytes 1944 (1.8 KiB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 24 bytes 1944 (1.8 KiB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

[ec2-user@ip-192-168-0-187 ~]$
  
```