

Amazon Rekognition Hands-on: Recognizing celebrities in an image

Trying to follow below example from AWS dev guide

<https://docs.aws.amazon.com/rekognition/latest/dg/celebrities-procedure-image.html>

Cost Analysis:

Amazon S3:

- 5 GB of Standard Storage
- 20,000 Get Requests
- 2,000 Put Requests

Amazon Rekognition:

During the free tier period you can analyze 5,000 images per month for free each, in Group 1 and Group 2 APIs.

Steps:

IAM

- Create a new user from IAM > Users > Add User
- Add user roles: Rekognition FullAccess and S3 Readonly

The screenshot shows the AWS IAM 'Add user' console. The 'Set permissions' section is active, and the 'Attach existing policies directly' tab is selected. A search bar shows 'rekog' and 'Showing 4 results'. The following table lists the policies:

	Policy name	Type	Used as
<input type="checkbox"/>	AmazonRekognitionCustomLabelsFullAccess	AWS managed	None
<input checked="" type="checkbox"/>	AmazonRekognitionFullAccess	AWS managed	None
<input type="checkbox"/>	AmazonRekognitionReadOnlyAccess	AWS managed	None
<input type="checkbox"/>	AmazonRekognitionServiceRole	AWS managed	None

At the bottom, there are buttons for 'Cancel', 'Previous', and 'Next: Tags'.

- Add name tag if you want to (optional). Review and Create user if all looks right.

arch for services, features, blogs, docs, and more

[Alt+S]

Global

Ankit.T

Add user

1

2

3

4

5

Review

Review your choices. After you create the user, you can view and download the autogenerated password and access key.

User details

User name

pocAwsAPIExercise

AWS access type

Programmatic access - with an access key

Permissions boundary

Permissions boundary is not set

Permissions summary

The following policies will be attached to the user shown above.

Type	Name
Managed policy	AmazonRekognitionFullAccess
Managed policy	AmazonS3ReadOnlyAccess

Tags

The new user will receive the following tag

Cancel

Previous

Create user

- Download the csv file or note down Access Key ID and secret key.

Search for services, features, blogs, docs, and more

[Alt+S]

Global

Ankit.T

Add user

1

2

3

4

5

✓ Success

You successfully created the users shown below. You can view and download user security credentials. You can also email users instructions for signing in to the AWS Management Console. This is the last time these credentials will be available to download. However, you can create new credentials at any time.

Users with AWS Management Console access can sign-in at: <https://864540690830.signin.aws.amazon.com/console>

Download .csv

	User	Access key ID	Secret access key
▶	✓ pocAwsAPIExercise	██████████	***** Show

Close

© 2022, Amazon Web Services, Inc. or its affiliates.

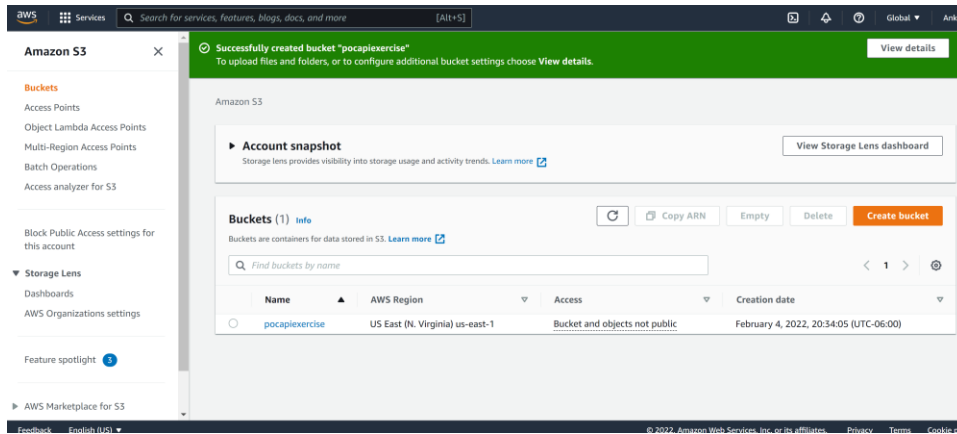
Privacy

Terms

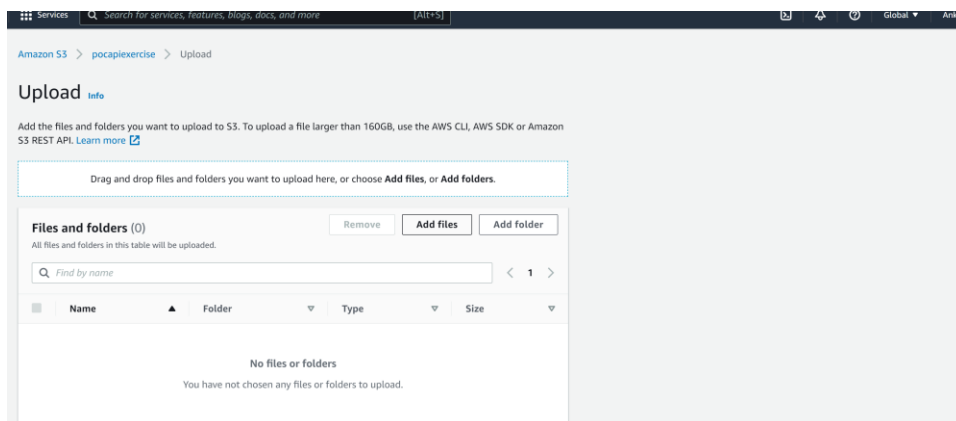
Cookie pr

Amazon S3

- Create an S3 bucket



- Upload a celebrity image in there



AWS CLI

- Start command prompt and set-up AWS CLI to use the user we just created using 'aws configure' command.
- Then run the Rekognition API with bucket name and image name

```
aws rekognition recognize-celebrities --image
"S3Object={Bucket=pocapiexercise,Name=celeb.jpg}"
```

```

c:\>aws configure
AWS Access Key ID [*****ID4U]:
AWS Secret Access Key [*****h+pI]:
Default region name [None]: us-east-1
Default output format [None]:

c:\>aws rekognition recognize-celebrities --image "S3Object={Bucket=pocapiexercise,Name=Natalie-Portman.jpg}"
{
  "CelebrityFaces": [
    {
      "Urls": [
        "www.wikidata.org/wiki/Q37876",
        "www.imdb.com/name/nm0000204"
      ],
      "Name": "Natalie Portman",
      "Id": "25Xf3SZ",
      "Face": {
        "BoundingBox": {
          "Width": 0.40293261408805847,
          "Height": 0.5816137790679932,
          "Left": 0.3083387315273285,
          "Top": 0.14042586088180542
        },
        "Confidence": 99.96854400634766,
        "Landmarks": [
          {
            "Type": "mouthRight",
            "X": 0.607524037361145,
            "Y": 0.5717466473579407
          },
          {
            "Type": "eyeRight",
            "X": 0.6163724660873413,
            "Y": 0.3824973404407501
          },
          {

```

Voila! It recognizes all celebrities in the image. I've tried a few for fun and the result was accurate each time 😊

CloudFormation Hands-on: Launch an Amazon EC2 Web Server

Trying to follow below example from AWS CloudFormation Guide:

<https://docs.aws.amazon.com/AWSCloudFormation/latest/UserGuide/working-with-templates-cfn-designer-walkthrough-createbasicwebserver.html>

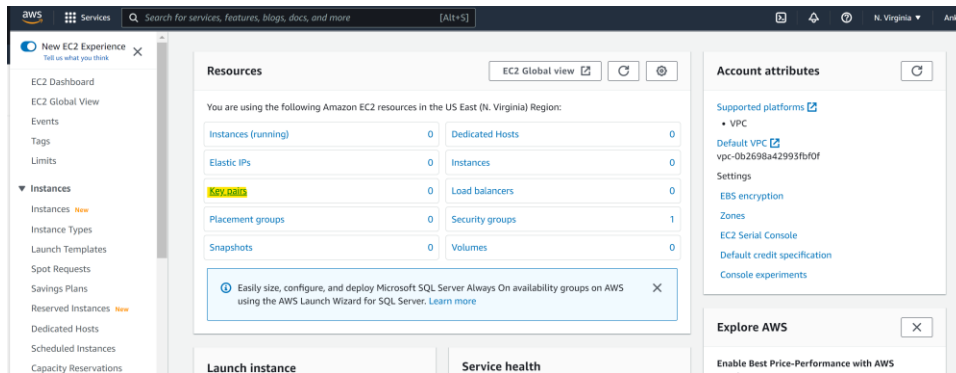
Cost-Analysis:

CloudFormation is a free service; however, you are charged for the AWS resources you include in your stacks at the current rate for each.

Steps:

EC2

If you don't already have a key-pair, create one from the EC2 console.



I've downloaded the PEM format here but you can specify PPK if you're used to that.

Key pair

A key pair, consisting of a private key and a public key, is a set of security credentials that you use to prove your identity when connecting to an instance.

Name

MyEC2KeyPair

The name can include up to 255 ASCII characters. It can't include leading or trailing spaces.

Key pair type [Info](#)

☒ RSA

☐ ED25519

Private key file format

☒ .pem

For use with OpenSSH

☐ .ppk

For use with PuTTY

Tags (Optional)

No tags associated with the resource.

Add tag

You can add 50 more tags.

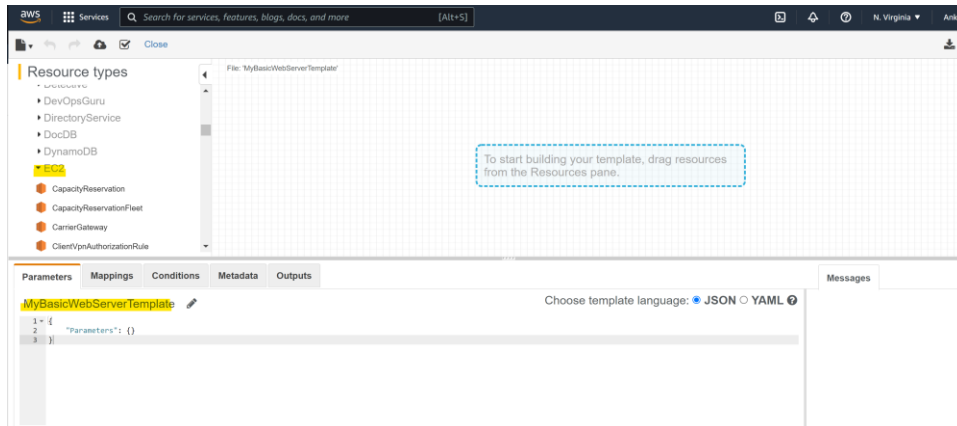
Cancel

Create key pair

CloudFormation

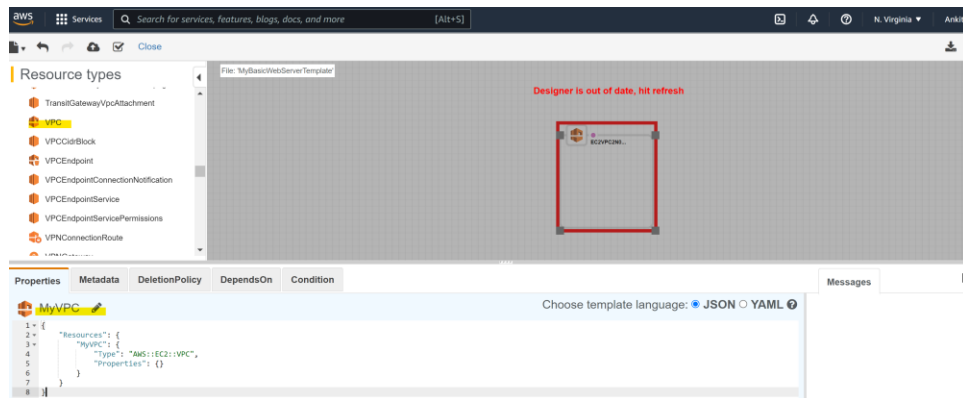
Launch CloudFormation Designer

Change the template name and expand EC2 category

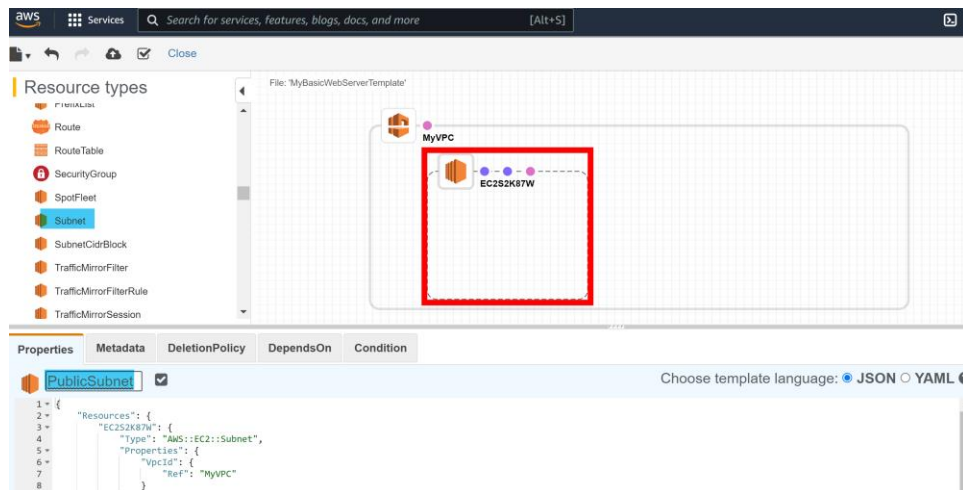


Drag-and-drop VPC onto the canvas and rename it. Hit refresh from top-right corner of the designer.

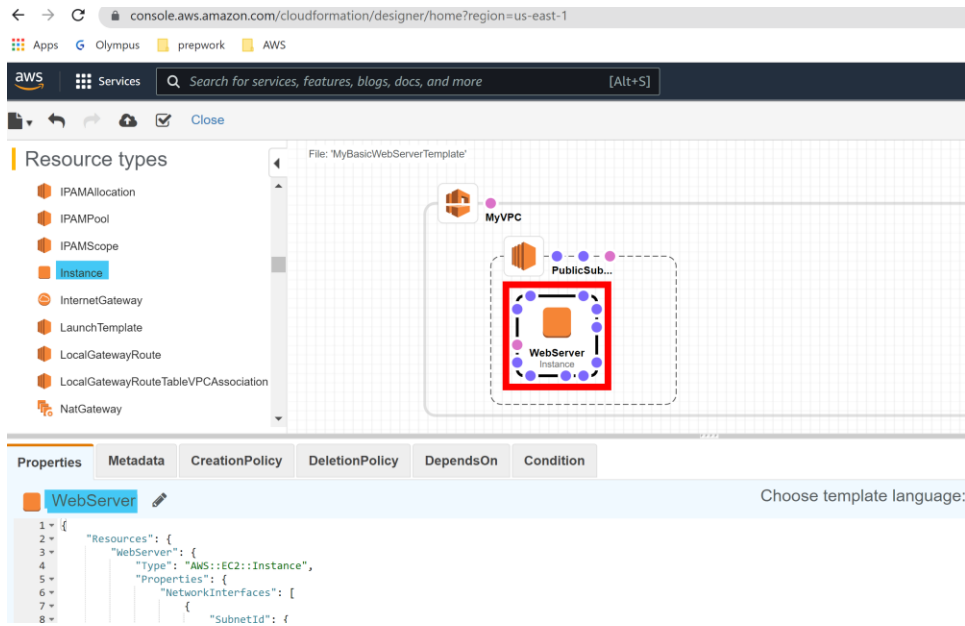
Drag the corners and make it a little bigger.



Drag-and-drop Subnet component from the category panel and rename it.

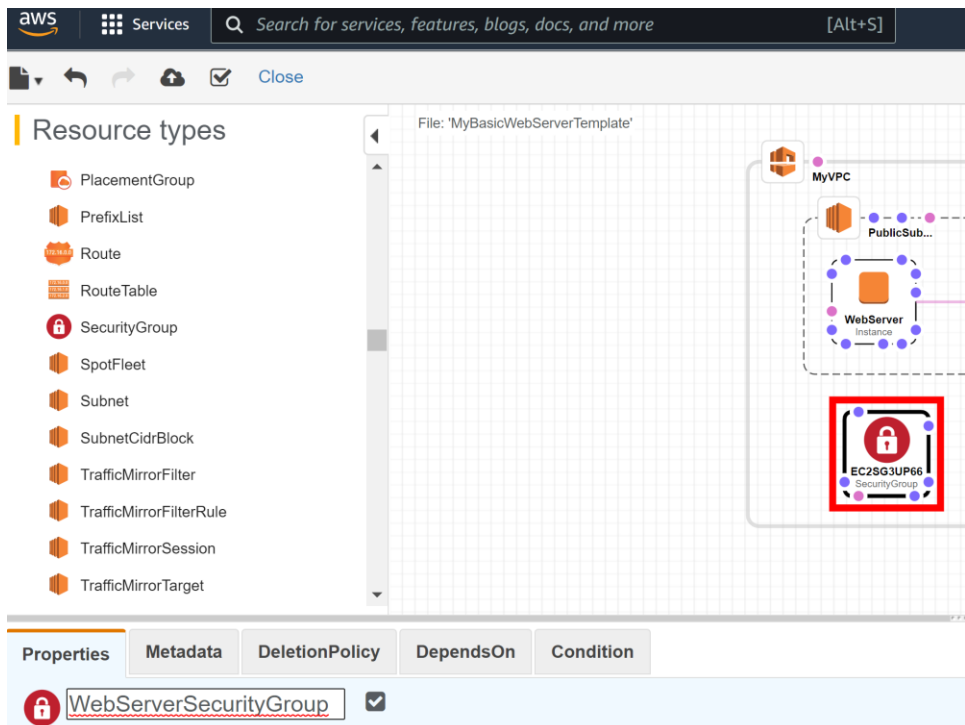


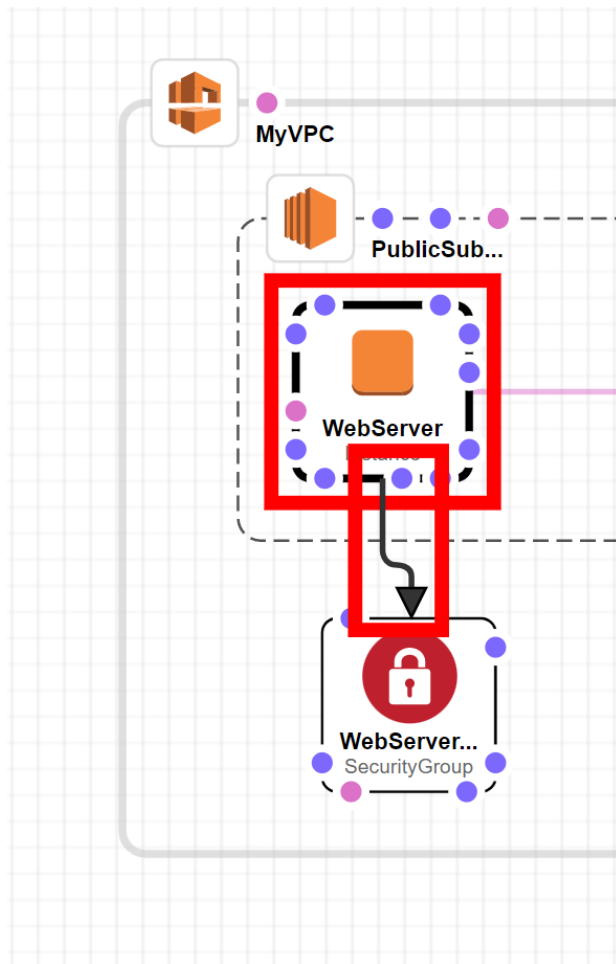
Drag Instance component from the category panel and drop it inside the Subnet and rename it.



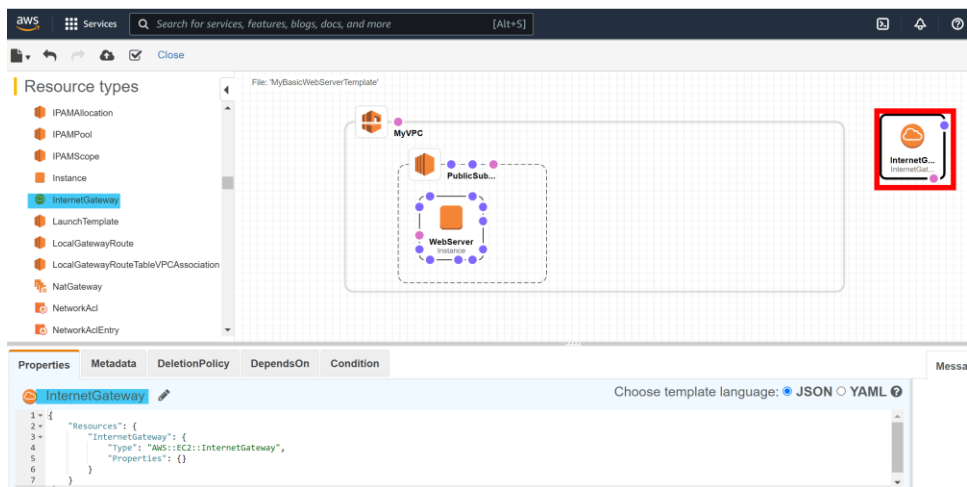
Drag-and-drop Security Group component from the category panel and rename it.

Then connect security group connection from WebServer to the Security group.

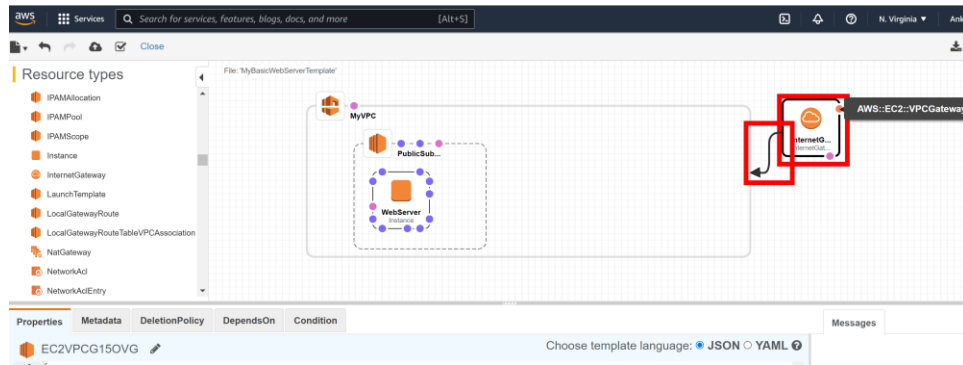




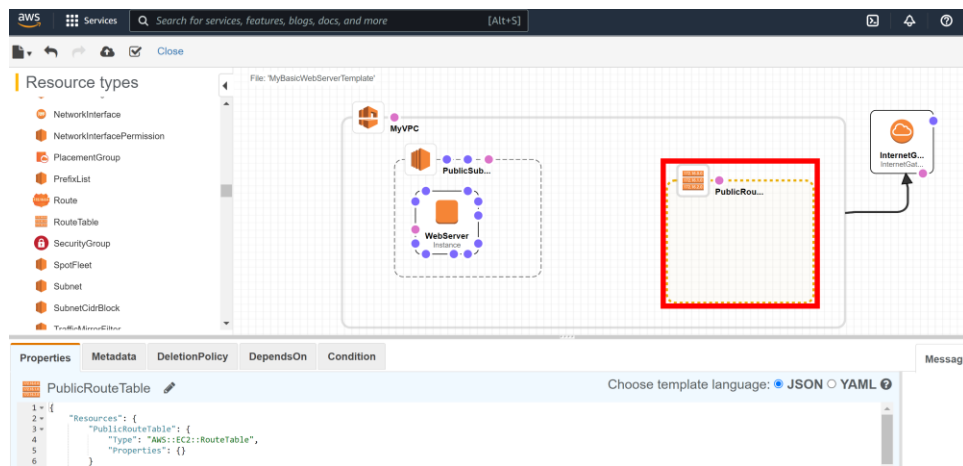
Drag-and-drop Internet Gateway component from the category panel inside the Subnet and rename it.



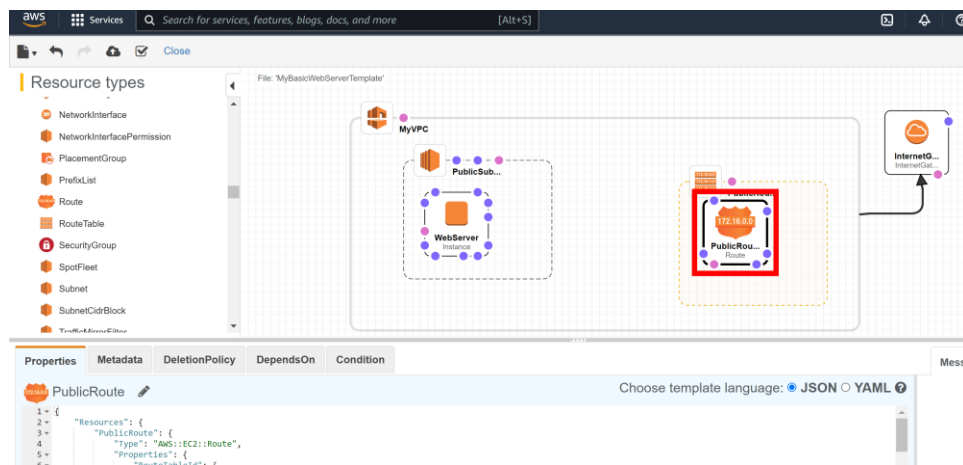
Attach IG to VPC



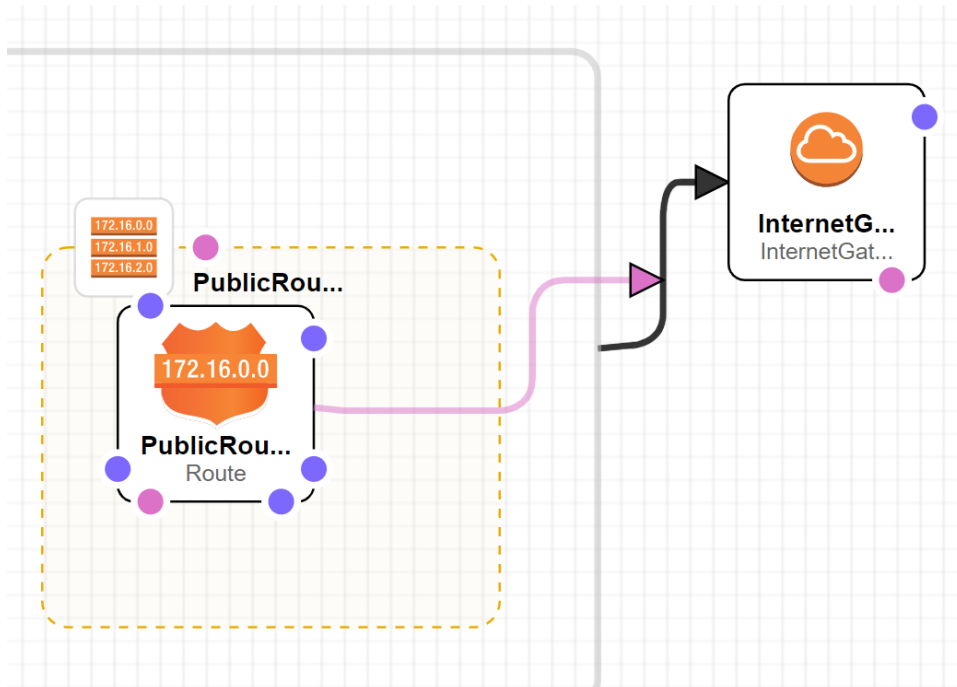
Drag and drop Route table into VPC. Expand the VPC size if it doesn't allow to drop it.



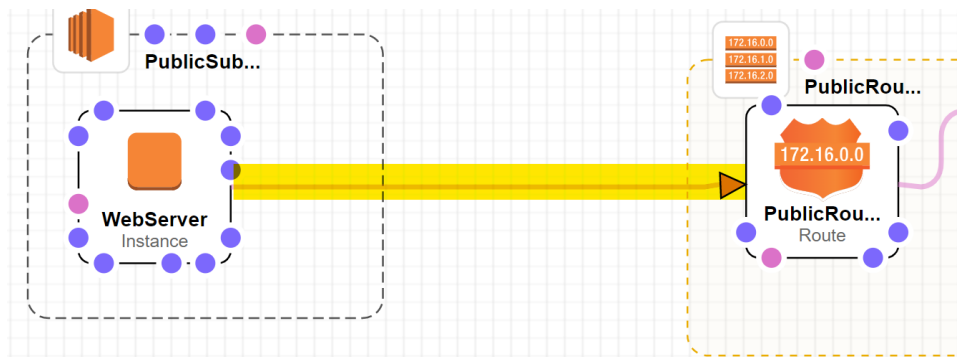
Drop Route component inside Route Table and rename it.



Create a “depends on” link between the route entry and IG

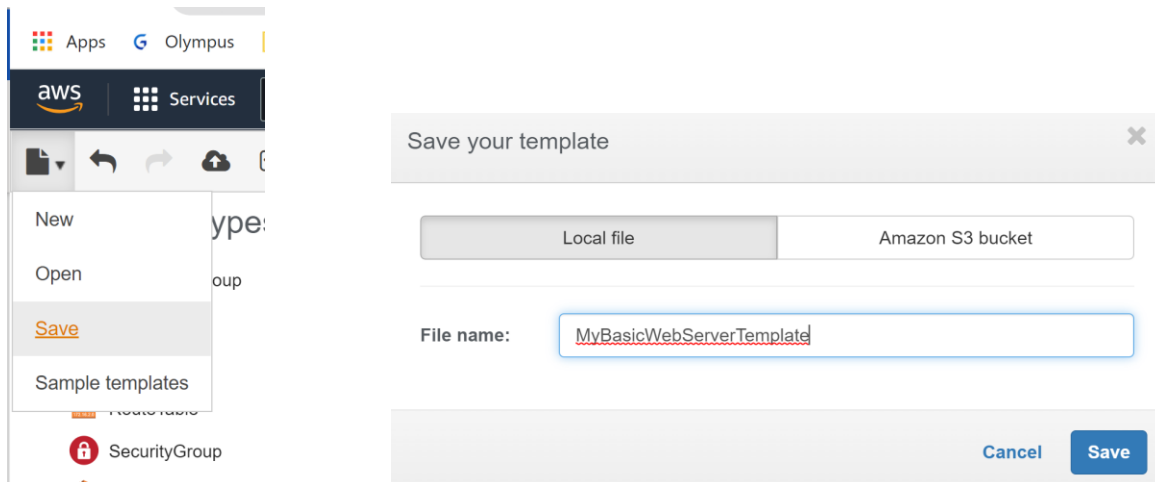


Now create a “depends on” link from WebServer to Route entry.

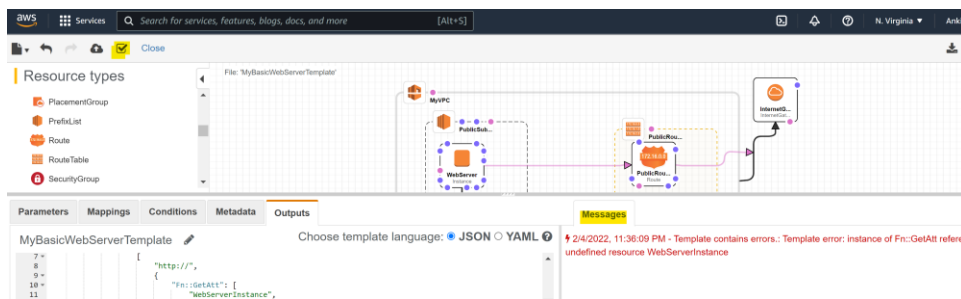


Now, copy over configurations to resources from the article page. Beware that some configs are to be replaced while others are to be appended.

Save the template file to your local now.



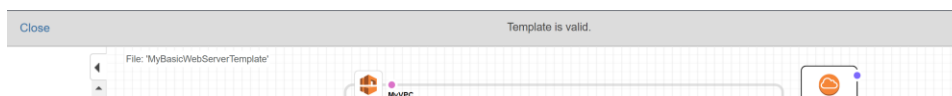
Now click on Validate icon from the toolbar and check if the template is correctly configured.



Looks like we got an error here! Let's try to understand. It says can't find WebServerInstance resource in the function.

Okay, I get it. I had changed named few components different than in the article. I'm going to update that name in the function and run the validation check again.

Hurray! Now it's valid 😊



It's time to provision resources and put the template into action.

Click the cloud icon to create stack. It'll auto create an S3 bucket. Review details and click next.

Services

Search for services, features, blogs, docs, and more

[Alt+S]

Step 1

Specify template

Step 2

Specify stack details

Step 3

Configure stack options

Step 4

Review

Create stack

Prerequisite - Prepare template

Prepare template

Every stack is based on a template. A template is a JSON or YAML file that c

☒ Template is ready

☐ Use a s

Specify template

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

Template source

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

☒ Amazon S3 URL

Amazon S3 URL

https://s3-external-1.amazonaws.com/cf-templates-bf6j317kqj

Amazon S3 template URL

S3 URL: https://s3-external-1.amazonaws.com/cf-templates-bf6zi7r9gxvp4

Specify stack name, select instance type and then select key-pair that we created in first step. Click Next.

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.

InstanceType

WebServer EC2 instance type

KeyName

Name of an EC2 KeyPair to enable SSH access to the instance.

SSHLocation

The IP address range that can be used to access the web server using SSH.

Configure Stack options if you want to. Click Next

Configure stack options

Tags

You can specify tags (key-value pairs) to apply to resources in your stack. You can add up to 50 unique tags for each resource.

Add tag

Permissions

Choose an IAM role to explicitly define how CloudFormation can create, modify, or delete resources in the stack. If you don't choose a role, CloudFormation uses the default role based on your user credentials. [Learn more](#)

IAM role - optional

Choose the IAM role for CloudFormation to use for all operations performed on the stack.

Stack failure options

Behavior on provisioning failure

Specify the roll back behavior for a stack failure. [Learn more](#)

Review details and click Create Stack for AWS to start creating resources. And that's it!

Notification options

No notification options

There are no notification options defined

Stack creation options

Timeout

-

Termination protection

Disabled

► [Quick-create link](#)

Cancel

Previous

Create change set

Create stack