lab

lab title

Monitoring AWS Services
V1.00

Course title

BackSpace Academy
AWS Certified Associate

BackSpace

Table of Contents

Contents

About the Lab

Please note that not all AWS services are supported in all regions. Please use the US-East-1 (North Virginia) region for this lab.

These lab notes are to support the hands on instructional videos of the Monitoring section of the AWS Certified Associate Course.

Please note that AWS services change on a weekly basis and it is extremely important you check the version number on this document to ensure you have the lastest version with any updates or corrections.

Implementing CloudWatch Monitoring Scripts on EC2

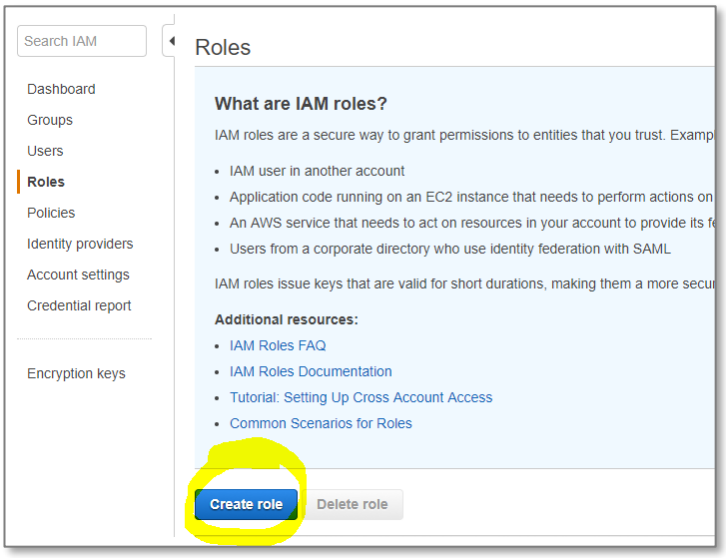
In this section we will create an EC2 server and assign it a role to access the CloudWatch service. We will then install CloudWatch monitoring scripts to produce custom CloudWatch metrics.

Create IAM Role for EC2

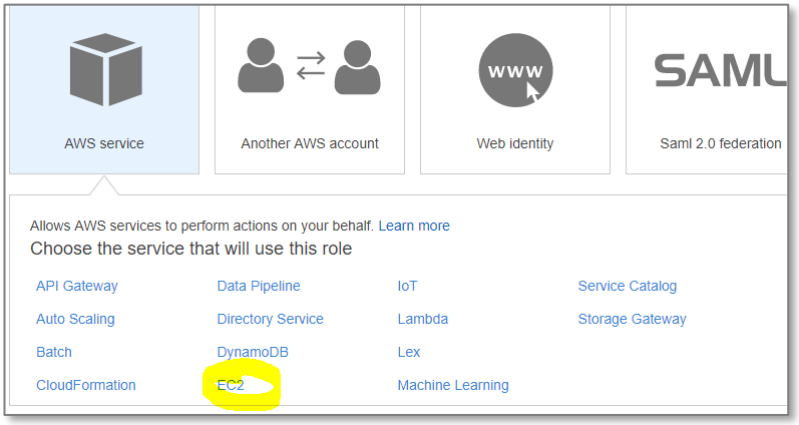
Select “Services” “IAM” to go to the IAM console

Select Roles

Click “Create Role”

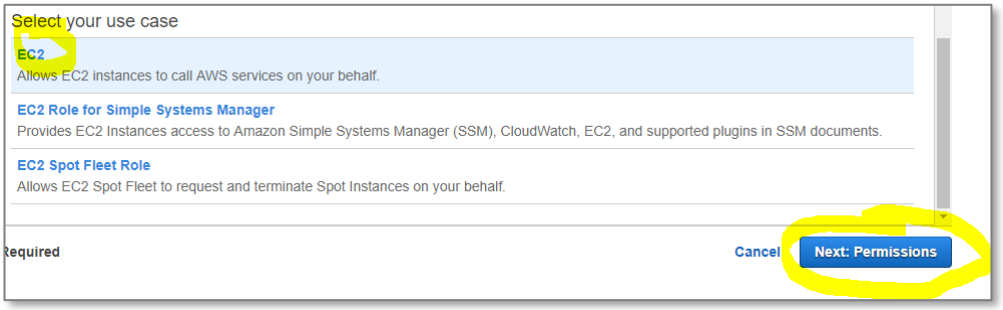


Select EC2

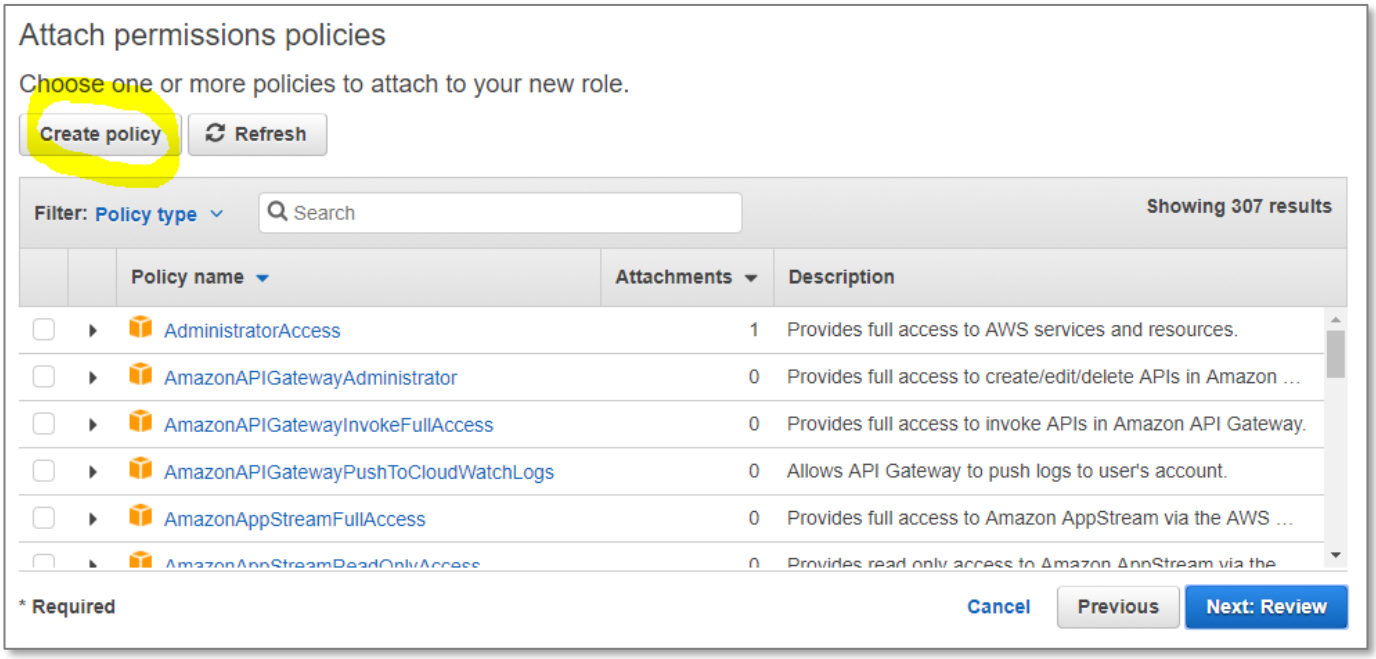


Select EC2 again

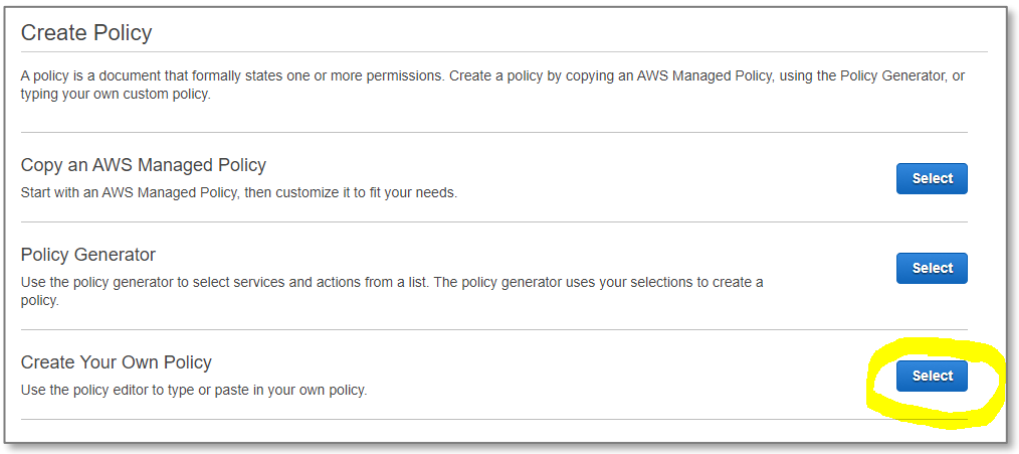
Click “Next: Permissions”



Click “Create Policy”



Select the “Create Your Own Policy”



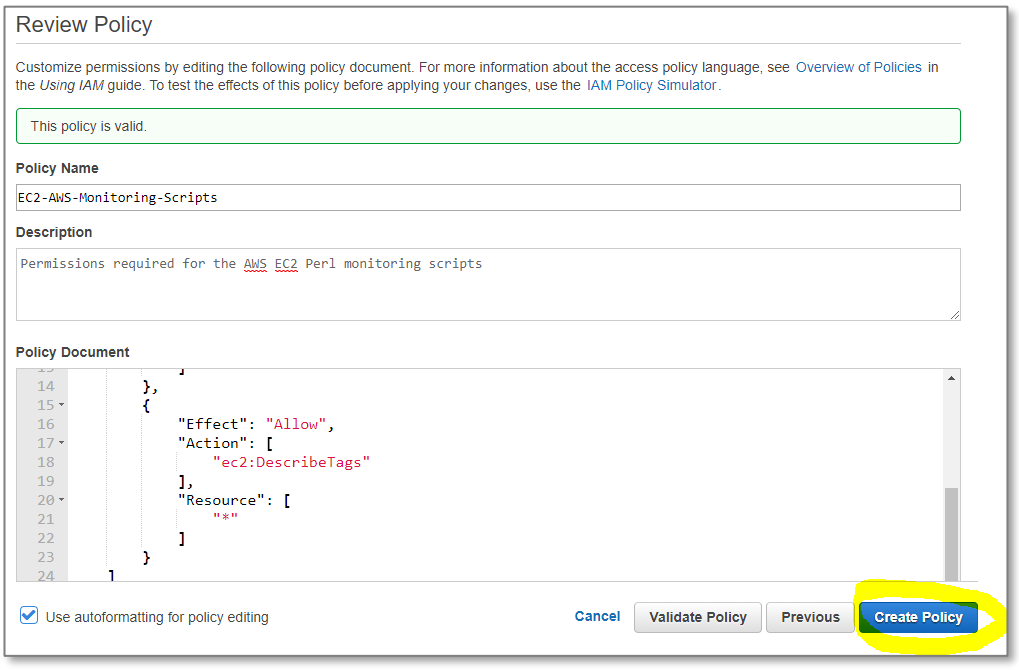
Give the policy a name and description

Paste the following JSON into Policy Document.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "cloudwatch:GetMetricStatistics",
                "cloudwatch:ListMetrics",
                "cloudwatch:PutMetricData"
            ],
            "Resource": [
                "*"
            ]
        },
        {
            "Effect": "Allow",
            "Action": [
                "ec2:DescribeTags"
            ],
            "Resource": [
                "*"
            ]
        }
    ]
}
```

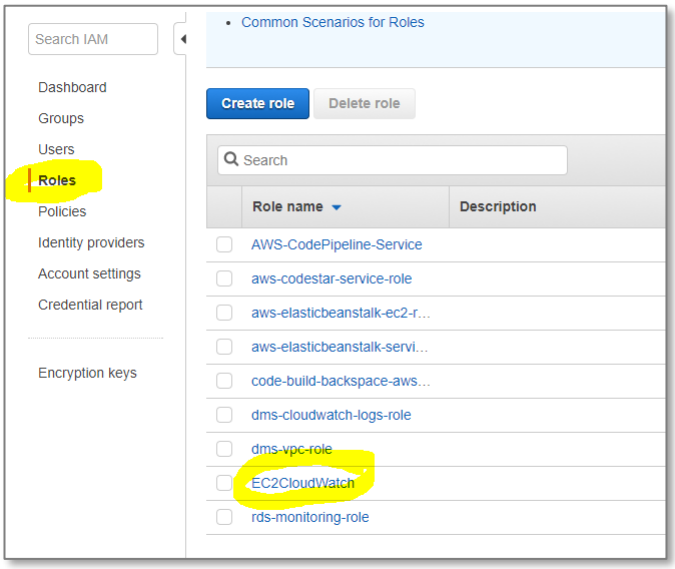
Click “Validate Policy to check it is OK

Click “Create Policy”

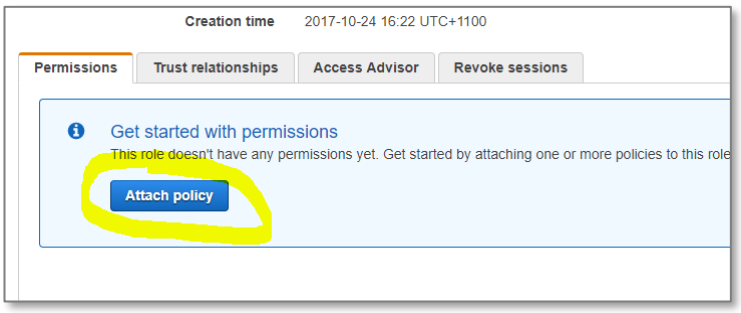


Go back to the Roles page

Click on the role you created



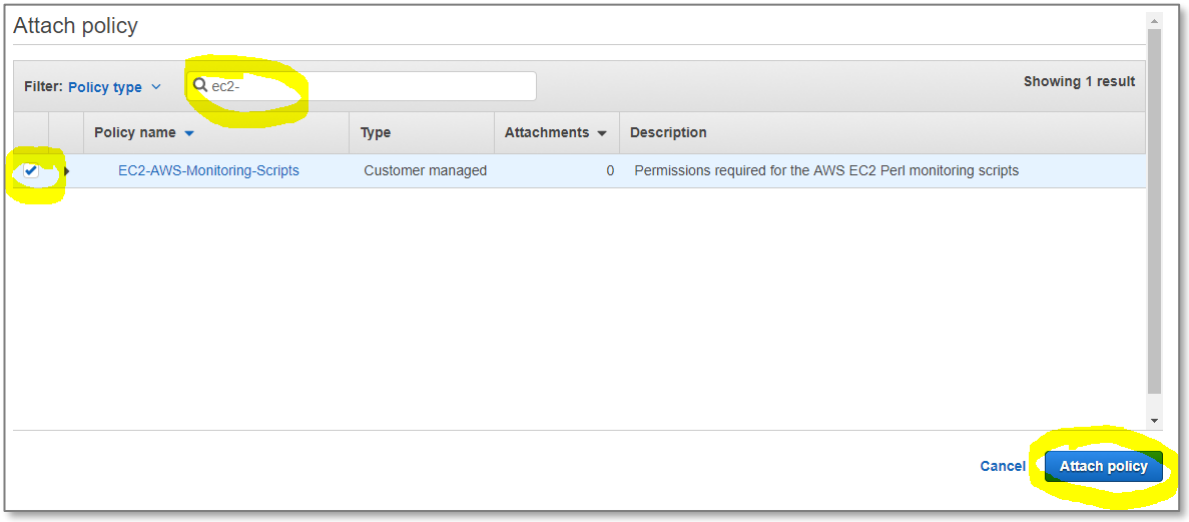
Click “Attach policy”



Search for your policy

Select the policy

Click “Attach policy”

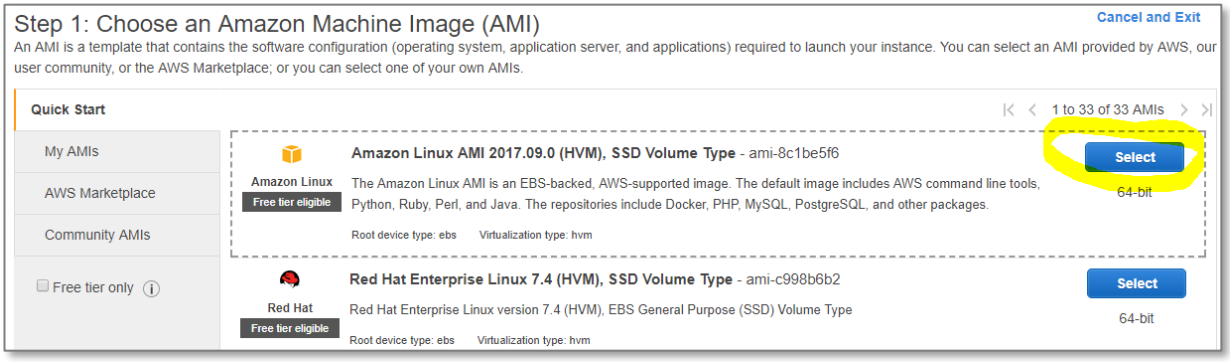


Create EC2 Server

Click on the services menu and select EC2.

Click “Launch Instance”

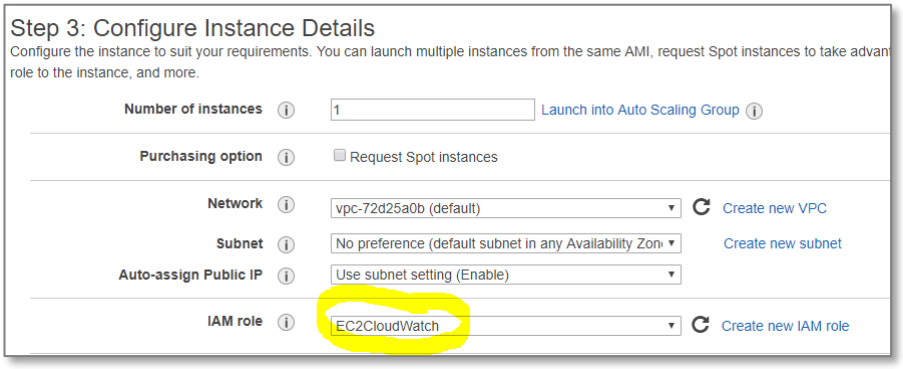
Select the Amazon Linux AMI



Select t2 micro instance type

Click “Next: Configure Instance Details”

Select our EC2 Role

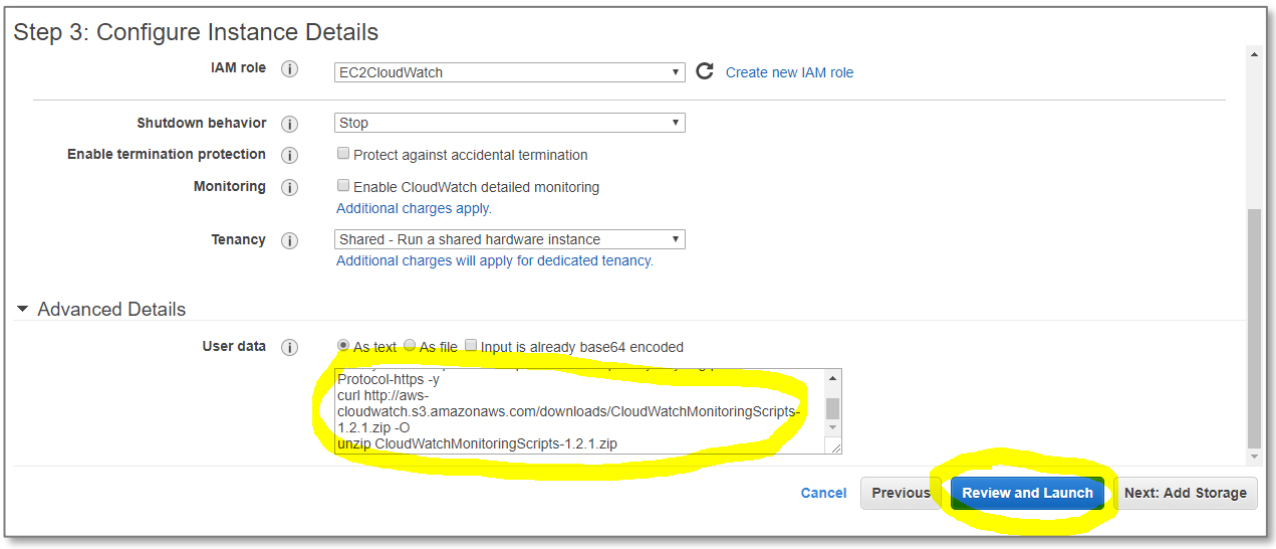


Expand the “Advanced Details”

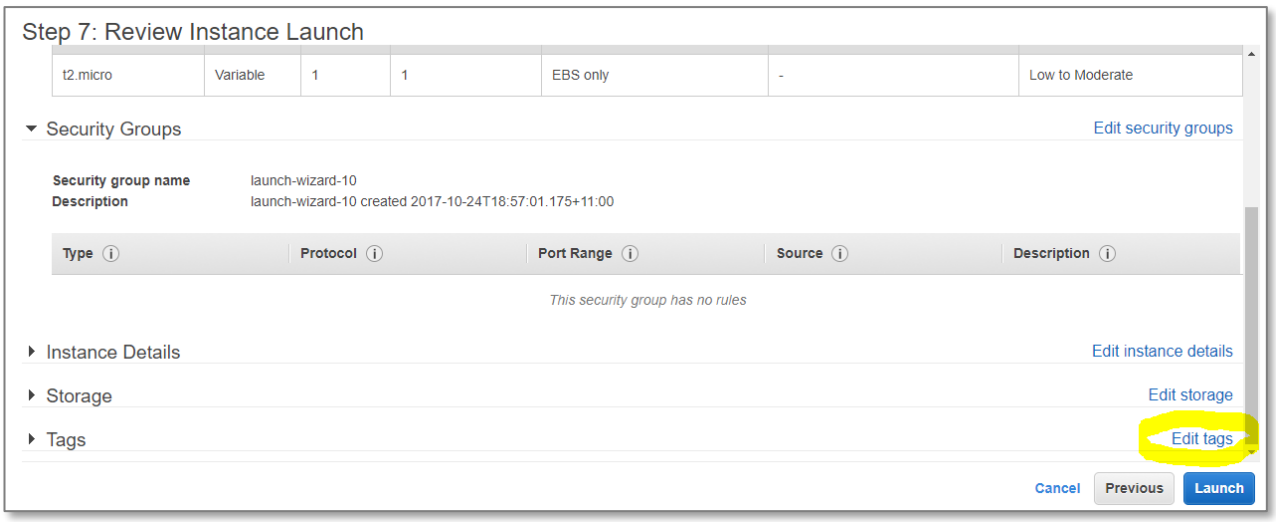
Add the following bash script in “User Data”

```
#!/bin/bash
yum -y update
sudo yum install perl-Switch perl-DateTime perl-Sys-Syslog perl-LWP-Protocol-https -y
curl http://aws-cloudwatch.s3.amazonaws.com/downloads/CloudWatchMonitoringScripts-1.2.1.zip -O
unzip CloudWatchMonitoringScripts-1.2.1.zip
```

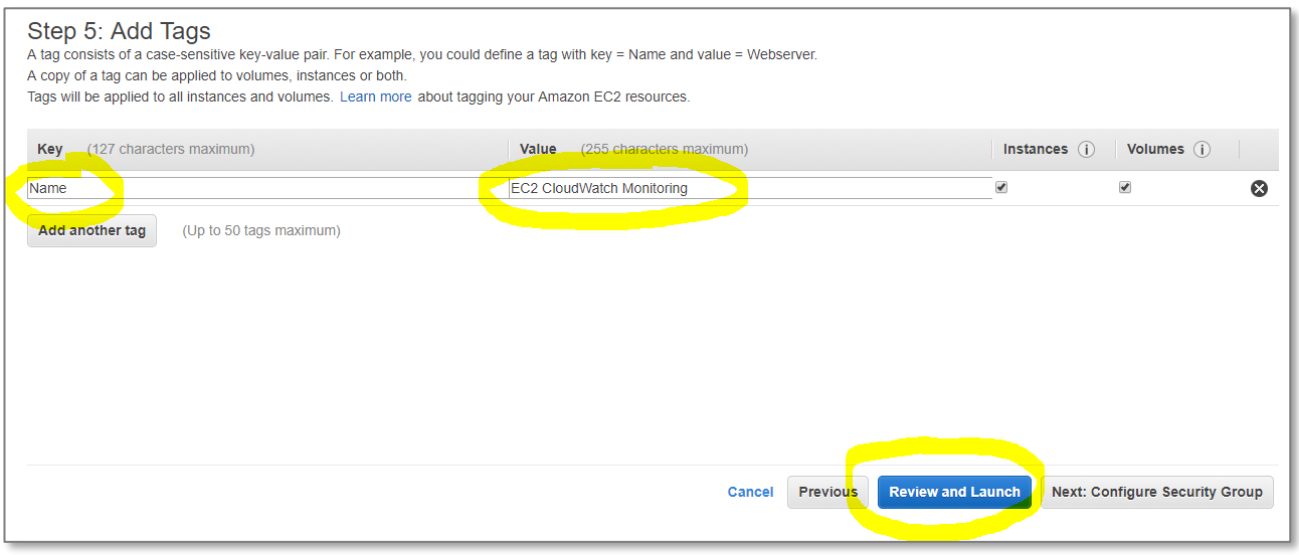
Click Review and Launch”



Scroll down and click “Edit tags”



Give the instance a “Name” tag



Click Review and Launch”

Click “Launch”

Select a key pair launch the instance

Running the EC2 Monitoring Scripts

Open your command line.

Navigate to the location of your key pair

Connect to your instance using the CLI

```
G:\OneDrive\Documents\KeyPairs\BackSpace
λ ssh -i "pcoady-us-east-1.pem" ec2-user@ec2-52-207-109-192.compute-1.amazonaws.com
The authenticity of host 'ec2-52-207-109-192.compute-1.amazonaws.com (52.207.109.192)' can't be established.
ECDSA key fingerprint is SHA256:3D+WibAgR9sff7+GIiFP5E9rwr7mR2VNaWHDgxV17Pk.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added 'ec2-52-207-109-192.compute-1.amazonaws.com,52.207.109.192' (ECDSA) to the list of known hosts.

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

https://aws.amazon.com/amazon-linux-ami/2017.09-release-notes/
[ec2-user@ip-172-31-81-206 ~]$
```

Change directory to where the scripts are located

```
cd ~/../../aws-scripts-mon
```

Check the scripts are working

```
./mon-put-instance-data.pl --mem-util --verify --verbose
```

```
[ec2-user@ip-172-31-81-206 aws-scripts-mon]$ ./mon-put-instance-data.pl --mem-util --verify --verbose
MemoryUtilization: 7.96874938564312 (Percent)
No credential methods are specified. Trying default IAM role.
Using IAM role <EC2CloudWatch>
Endpoint: https://monitoring.us-east-1.amazonaws.com
Payload: {"MetricData":[{"Timestamp":1508823931,"Dimensions":[{"Value":"i-065f24f3730d00542","Name":"InstanceId"}],"Value":7.96874938564312,
"Unit":"Percent","MetricName":"MemoryUtilization"}],"Namespace":"System/Linux","__type":"com.amazonaws.cloudwatch.v2010_08_01#PutMetricDataI
nput"}

Verification completed successfully. No actual metrics sent to CloudWatch.

[ec2-user@ip-172-31-81-206 aws-scripts-mon]$
```

Run the script to send metric data to CloudWatch

```
./mon-put-instance-data.pl --mem-util --mem-used-incl-cache-buff --mem-used --mem-avail
```

Creating a CRON task to Push Metrics to CloudWatch

Edit the Linux CRON Table using the command

```
crontab -e
```

Press insert key

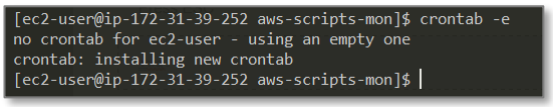
We will push metric data to Cloudwatch every minute

```
* * * * * ~/../../aws-scripts-mon/mon-put-instance-data.pl --mem-used-incl-cache-buff --mem-
util --disk-space-util --disk-path=/ --from-cron
```

After you have copied and pasted the CRON statement you can save and exit

Press ESC to exit insert mode

```
:wq!
```

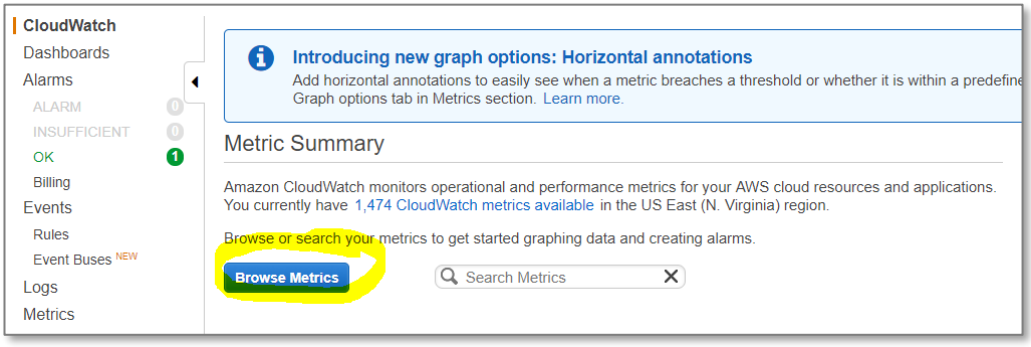


Viewing Metrics in CloudWatch

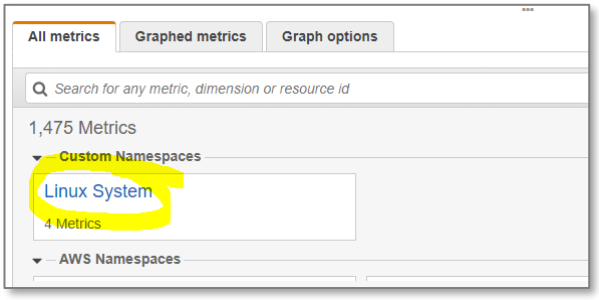
Wait a few minutes for data to be pushed to CloudWatch

Go to the CloudWatch Console

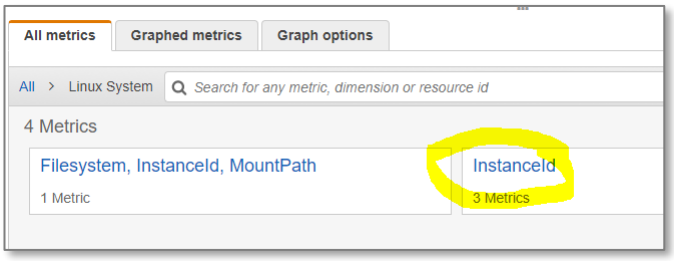
Click “Browse Metrics”



Select “Linux System”

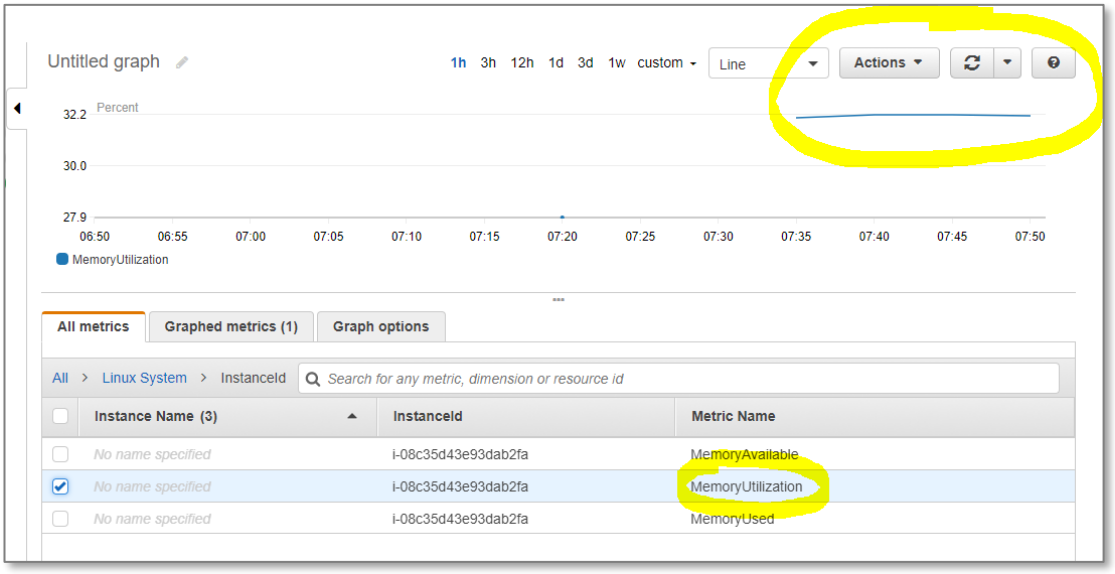


Select “Instance ID”



Select “Memory Utilization”

You will now see a graph of the metric pushed from the EC2 instance every minute



Clean Up

Now delete the EC2 Instance so that you don't get billed.