Lambda function to start using tags:

import json

import boto3

import datetime

ec2\_resource=boto3.resource('ec2')

def lambda\_handler(event, context):

#print("Received event: " + json.dumps(event, indent=2))

print('Starting instances tagged raghu')

linux\_instances = ec2\_resource.instances.filter(

Filters=[

{'Name': 'instance-state-name', 'Values': ['stopped']},

{'Name': 'tag-key', 'Values': ['']},

{'Name': 'tag-value', 'Values': ['']}

])

windows\_instances = ec2\_resource.instances.filter(

Filters=[

{'Name': 'instance-state-name', 'Values': ['stopped']},

{'Name': 'tag-key', 'Values': ['']},

{'Name': 'tag-value', 'Values': ['']}])

# linux\_start=linux\_instances.start()

# windows\_start=windows\_instances.start()

for i in linux\_instances:

print "starting this linux instance"

linux\_response=i.start()

print linux\_response

# print linux\_response.instance\_id

response=i.create\_tags(Tags=[{'Key':'LinuxPatchingstatedtime','Value':'linuxlatestpatch'}])

print response

for j in windows\_instances:

print "starting this windows instance"

windows\_response = j.start()

print windows\_response

# print windows\_response.instance\_id

winresponse=j.create\_tags(Tags=[{'Key': 'WindowsPatchingstatedtime', 'Value': 'windowslatestpatch'}])

print winresponse

Stop the instances automatically which are started by the above lambda function:

import json

import boto3

import datetime

ec2\_resource=boto3.resource('ec2')

ec2\_client=boto3.client('ec2')

def lambda\_handler(event, context):

#print("Received event: " + json.dumps(event, indent=2))

print('Starting instances tagged Patch Group')

linux\_instances = ec2\_resource.instances.filter(

Filters=[

{'Name': 'instance-state-name', 'Values': ['running']},

{'Name': 'tag-key', 'Values': ['LinuxPatchingstatedtime']},

{'Name': 'tag-value', 'Values': ['linuxlatestpatch']}

])

windows\_instances = ec2\_resource.instances.filter(

Filters=[

{'Name': 'instance-state-name', 'Values': ['running']},

{'Name': 'tag-key', 'Values': ['WindowsPatchingstatedtime']},

{'Name': 'tag-value', 'Values': ['windowslatestpatch']}])

# linux\_start=linux\_instances.start()

# windows\_start=windows\_instances.start()

for i in linux\_instances:

print "stopping this linux instance"

linux\_response=i.stop()

print linux\_response

# print linux\_response.instance\_id

response=i.delete\_tags(Tags=[{'Key':'LinuxPatchingstatedtime','Value':'linuxlatestpatch'}])

print response

for j in windows\_instances:

print "stopping this windows instance"

windows\_response = j.stop()

print windows\_response

# print windows\_response.instance\_id

winresponse=j.delete\_tags(Tags=[{'Key': 'WindowsPatchingstatedtime', 'Value': 'windowslatestpatch'}]) print winresponse