Deploying VM Series with Autoscale via VPN Attachments to a Transit Gateway

http://www.paloaltonetworks.com

Table of Contents

Version History

Version number	Comments
1.0	Initial GitHub check-in
1.1	Fixed issue with lambda files

1. About

This document will explain how to deploy an AWS Transit Gateway with the VM-Series on AWS. A Transit Gateway uses a hub and spoke architecture that allows security teams to centralize secure connectivity for VPC-to-VPC, VPC to corporate and VPC to the internet communications.

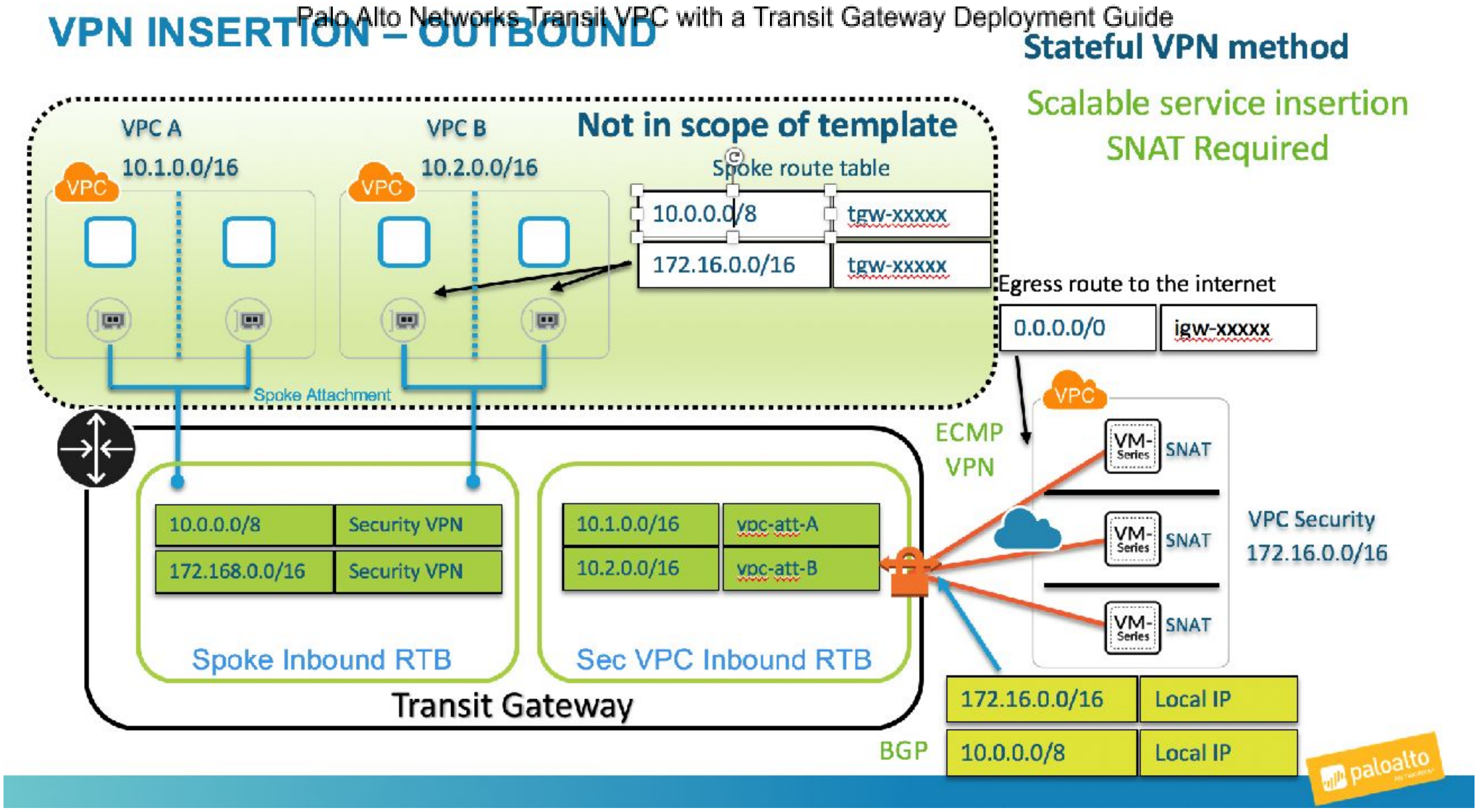
AWS Transit Gateway is a service that enables customers to connect their Amazon Virtual Private Clouds (VPCs) and their on-premises networks to a single gateway. As you grow the number of workloads running on AWS, you need to be able to scale your networks across multiple accounts and Amazon VPCs to keep up with the growth. Today, you can connect pairs of Amazon VPCs using peering. However, managing point-to-point connectivity across many Amazon VPCs, without the ability to centrally manage the connectivity policies, can be operationally costly and cumbersome. For on-premises connectivity, you need to attach your AWS VPN to each individual Amazon VPC. This solution can be time consuming to build and hard to manage when the number of VPCs grows into the hundreds.

With an AWS Transit Gateway, you only have to create and manage a single connection from the central gateway in to each VPC, on-premises data center, or remote office across your network. Transit Gateway acts as a hub that controls how traffic is routed among all the connected networks which act like spokes. This hub and spoke model simplifies management and reduces operational costs because each network only connects to the Transit Gateway and not to every other network. Any new VPC is simply connected to the Transit Gateway and is then automatically available to every other network that is connected to the Transit Gateway. This ease of connectivity makes it easy to scale your network as you grow.

More information on the AWS Transit Gateway can be found here:
https://aws.amazon.com/transit-gateway/

2. Topology

The template will deploy a security VPC with firewalls that will connect to an existing transit gateway via VPN attachments. The firewalls advertise a default route to the transit gateway via BGP over an IPSec connection:



VPN INSERTION – OUTBOUND

Stateful VPN method

Scalable service insertion
SNAT Required

Not in scope of template

Spoke route table

10.0.0.0/8	tgw-xxxxx
172.16.0.0/16	tgw-xxxxx

Egress route to the internet

0.0.0.0/0	igw-xxxxx

Spoke Inbound RTB

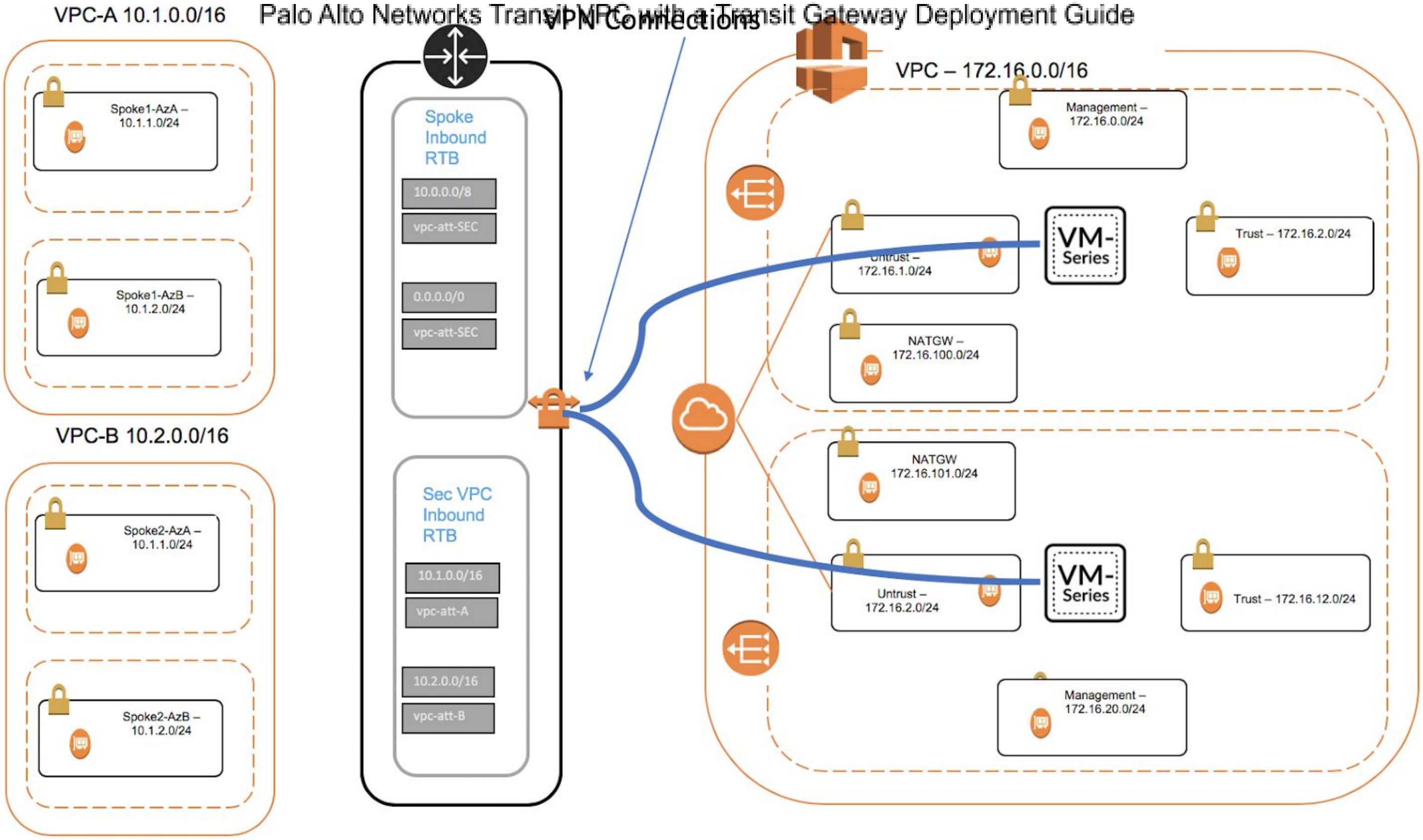
10.0.0.0/8	Security VPN
172.168.0.0/16	Security VPN

Sec VPC Inbound RTB

10.1.0.0/16	vpc-att-A
10.2.0.0/16	vpc-att-B

BGP

172.16.0.0/16	Local IP
10.0.0.0/8	Local IP


VPC-A 10.1.0.0/16
Spoke1-AzA – 10.1.1.0/24
Spoke1-AzB – 10.1.2.0/24
VPC-B 10.2.0.0/16
Spoke2-AzA – 10.1.1.0/24
Spoke2-AzB – 10.1.2.0/24
Spoke Inbound RTB
10.0.0.0/8
vpc-att-SEC
0.0.0.0/0
vpc-att-SEC
Sec VPC Inbound RTB
10.1.0.0/16
vpc-att-A
10.2.0.0/16
vpc-att-B
VPN Connections
VPC – 172.16.0.0/16
Management – 172.16.0.0/24
Untrust – 172.16.1.0/24
VM-Series
Trust – 172.16.2.0/24
NATGW – 172.16.100.0/24
NATGW 172.16.101.0/24
Untrust – 172.16.2.0/24
VM-Series
Trust – 172.16.12.0/24
Management – 172.16.20.0/24

3. Support Policy

This solution is released under an as-is, best effort, support policy. These scripts should be seen as community supported and Palo Alto Networks will contribute our expertise as and when possible. We do not provide technical support or help in using or troubleshooting the components of the project through our normal support options such as Palo Alto Networks support teams, or ASC (Authorized Support Centers) partners and backline support options. The underlying product used (the VM-Series firewall) by the scripts or templates are still supported, but the support is only for the product functionality and not for help in deploying or using the template or script itself.

4. Prerequisites

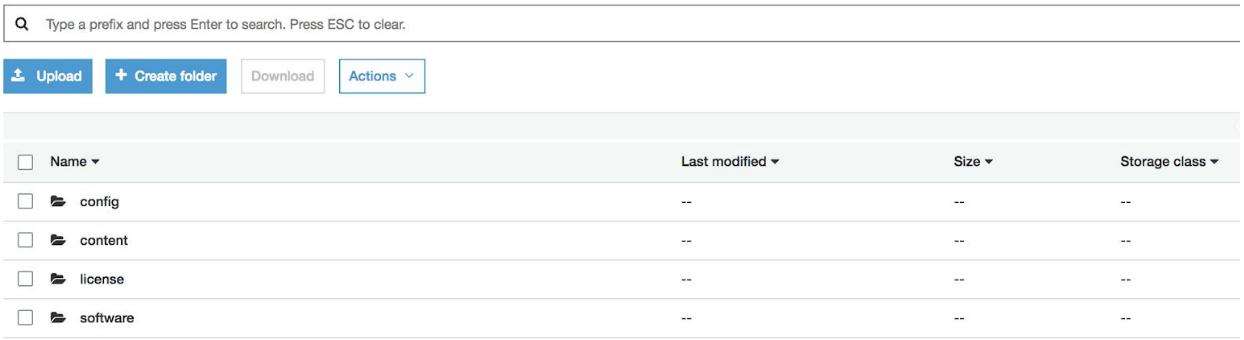
Here are the prerequisites required to successfully launch this template:

1. AWS account
2. Clone or download the files from the following GitHub repository on to your local machine:

   https://github.com/wwce/aws-cft/tree/master/transitgateway-demo-v2

5. Create S3 Buckets for Autoscale Security VPC

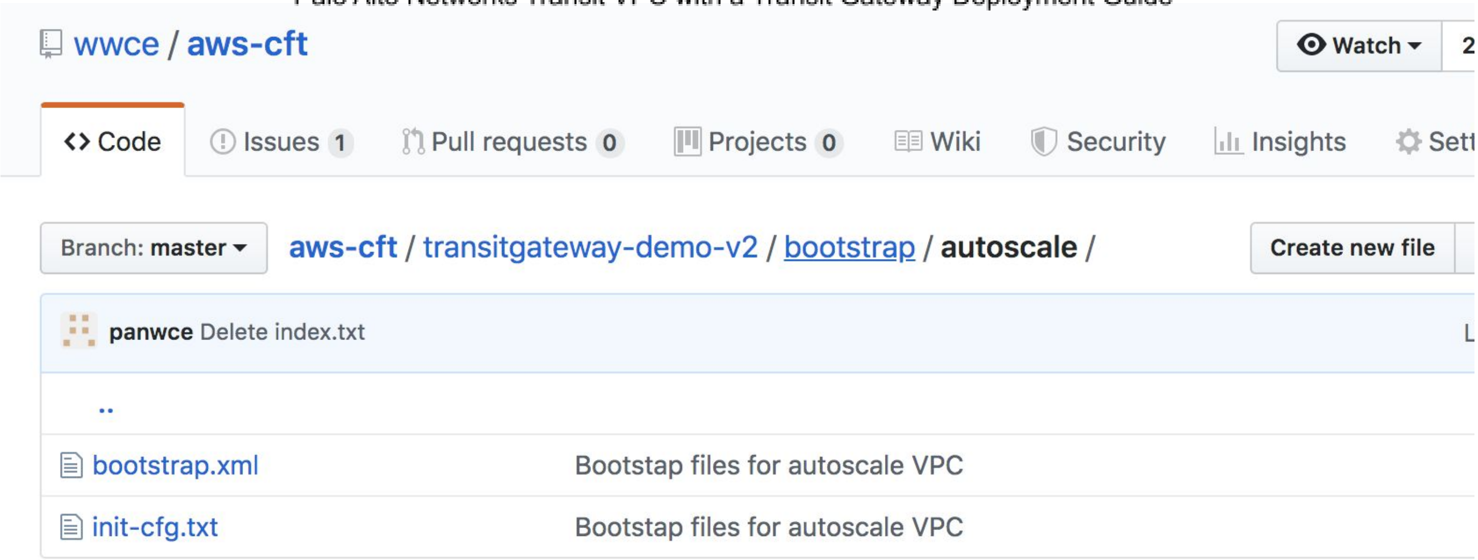
In the AWS S3 console, create an S3 bucket with config, content, license and software folders. All the folders must exist and in lowercase.


Type a prefix and press Enter to search. Press ESC to clear.
Upload
Create folder
Download
Actions

□	Name	Last modified	Size	Storage class
□	config	--	--	--
□	content	--	--	--
□	license	--	--	--
□	software	--	--	--

Copy the init-cfg.txt and bootstrap.xml files from the cloned ./bootstrap/autoscale folder to the config folder

NOTE: These are not the same bootstrap files that are used in the direct attached deployment. If you use those files the deployment will fail.



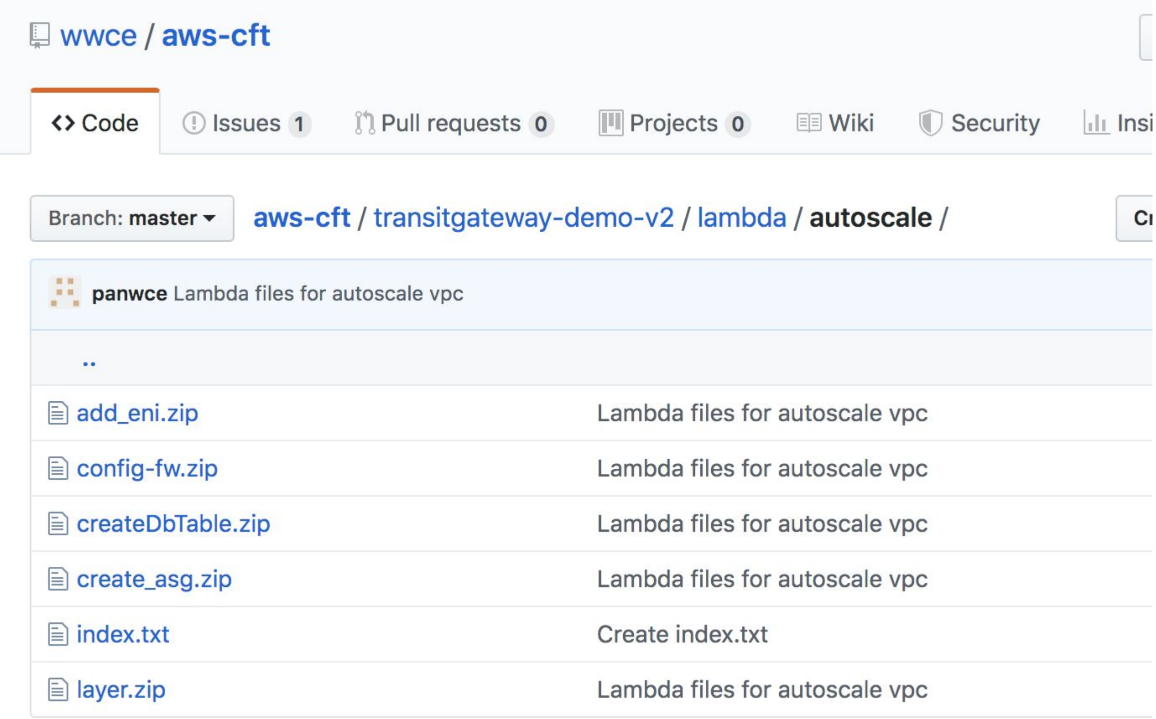
aws-cft / transitgateway-demo-v2 / bootstrap / autoscale /

panwce Delete index.txt

..

bootstrap.xml	Bootstap files for autoscale VPC
init-cfg.txt	Bootstap files for autoscale VPC

Either create another S3 bucket or use the existing bucket and add the lambda zip files to the folder. All the files shown below must be present and in zip format.



aws-cft / transitgateway-demo-v2 / lambda / autoscale /

panwce Lambda files for autoscale vpc

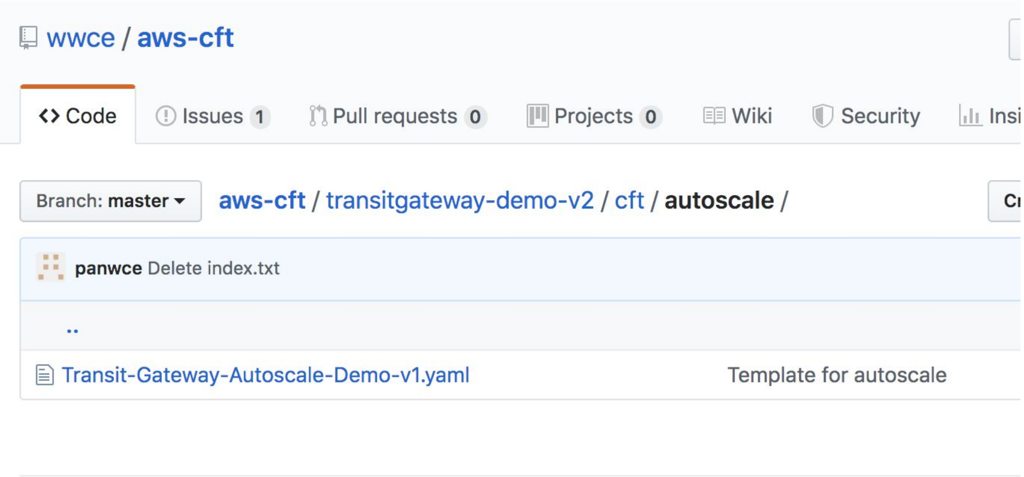
..

add_eni.zip	Lambda files for autoscale vpc
config-fw.zip	Lambda files for autoscale vpc
createDbTable.zip	Lambda files for autoscale vpc
create_asg.zip	Lambda files for autoscale vpc
index.txt	Create index.txt
layer.zip	Lambda files for autoscale vpc

Note: The buckets need to be in the same region in which you will deploy the Transit Gateway template.

6. Deploy the Transit Gateway Direct Attach Stack

In the AWS CloudFormation console create a new stack and select the
Transit-Gateway-Autoscale-Demo-v1.yaml template
and fill in the parameters



aws-cft / transitgateway-demo-v2 / cft / autoscale /

panwce Delete index.txt

..

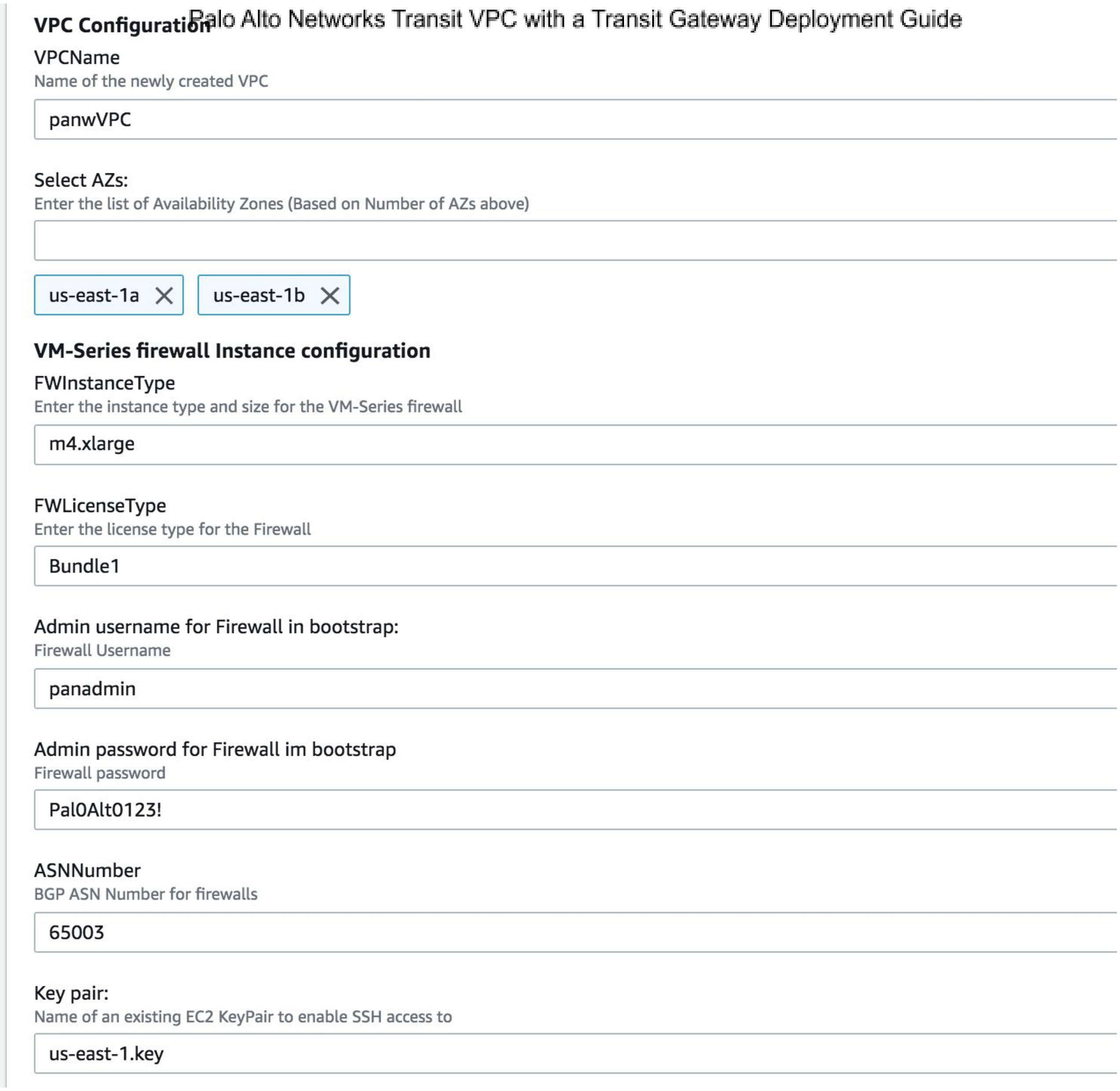
Transit-Gateway-Autoscale-Demo-v1.yaml Template for autoscale

a. VPC Configuration
   Select the number of VPCs required. Two is recommended for most deployments

   The firewall admin name and password should be used unless you have modified the bootstrap.xml file. Once the firewall has been configured it is recommended that the firewalls are connected to Panorama and the account credentials modified or deleted. In a production environment it is recommended that all credentials are stored in a secure vault.

   Any private ASN number can be used apart from the AWS default of 64512.

   In most cases the same lambda bucket can be used for firewall bootstrap and for the lambda code.



VPC Configuration

VPCName
Name of the newly created VPC
panwVPC

Select AZs:
Enter the list of Availability Zones (Based on Number of AZs above)

us-east-1a ✕ us-east-1b ✕

VM-Series firewall Instance configuration

FWInstanceType
Enter the instance type and size for the VM-Series firewall
m4.xlarge

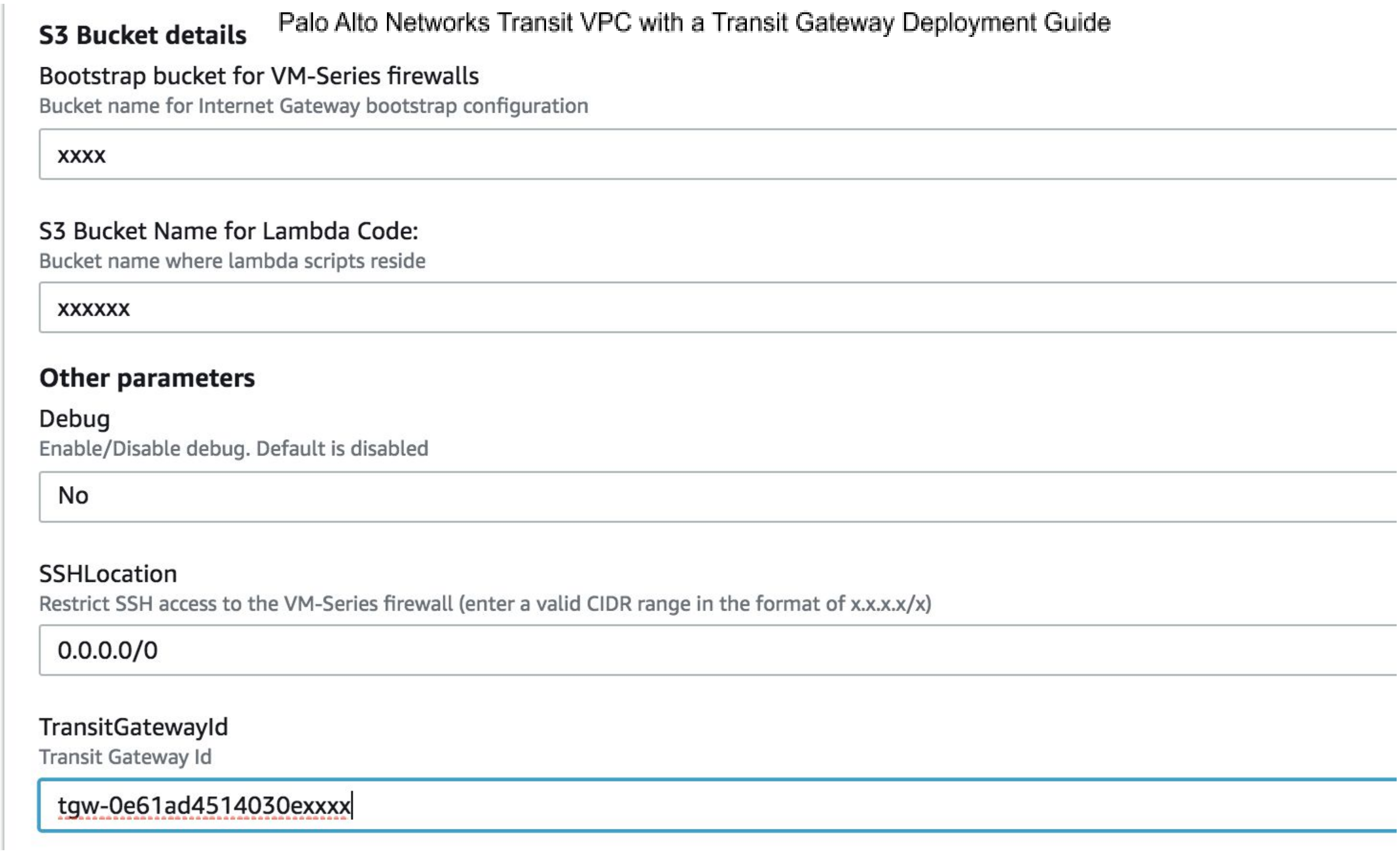
FWLicenseType
Enter the license type for the Firewall
Bundle1

Admin username for Firewall in bootstrap:
Firewall Username
panadmin

Admin password for Firewall im bootstrap
Firewall password
Pal0Alt0123!

ASNNumber
BGP ASN Number for firewalls
65003

Key pair:
Name of an existing EC2 KeyPair to enable SSH access to
us-east-1.key



S3 Bucket details

Bootstrap bucket for VM-Series firewalls
Bucket name for Internet Gateway bootstrap configuration
xxxx

S3 Bucket Name for Lambda Code:
Bucket name where lambda scripts reside
xxxxxx

Other parameters

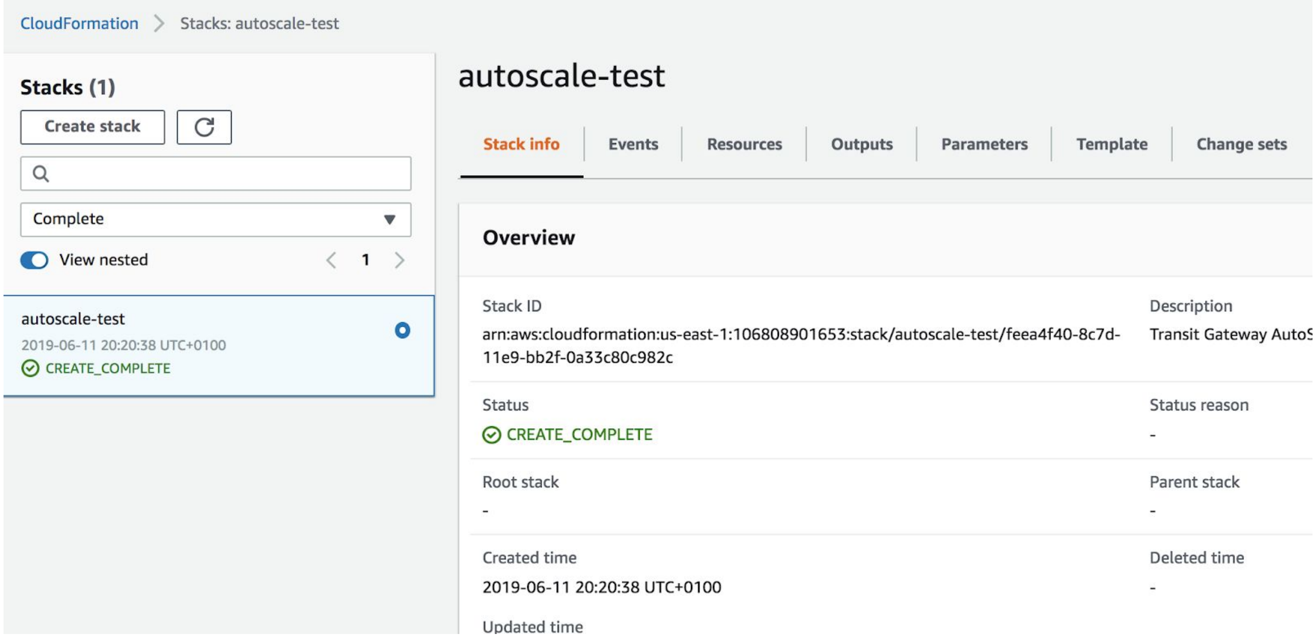
Debug
Enable/Disable debug. Default is disabled
No

SSHLocation
Restrict SSH access to the VM-Series firewall (enter a valid CIDR range in the format of x.x.x.x/x)
0.0.0.0/0

TransitGatewayId
Transit Gateway Id
tgw-0e61ad4514030exxxx

Click through to kick of stack creation.

You should see a stack create complete when the autoscale template has been successfully initialized.



Stacks (1)

Complete

autoscale-test
2019-06-11 20:20:38 UTC+0100
CREATE_COMPLETE

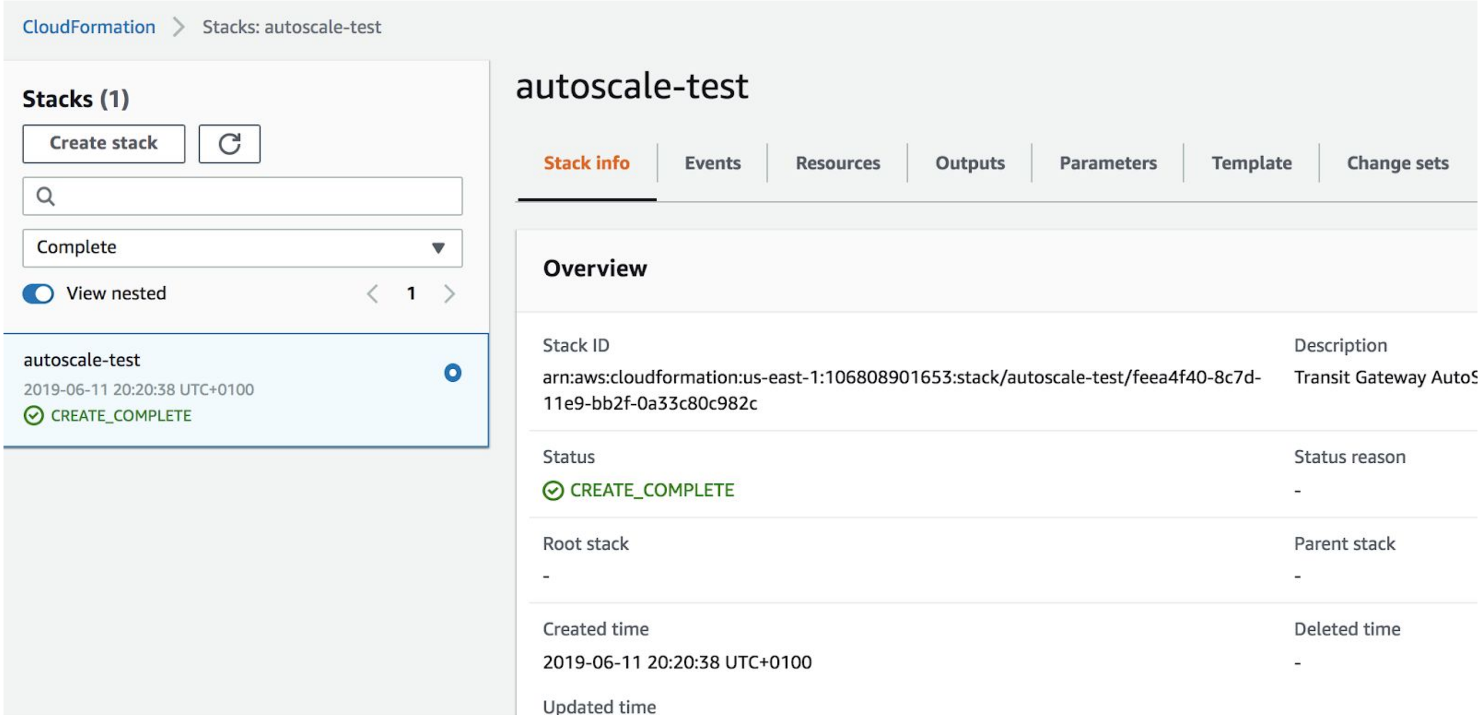
autoscale-test

Overview

Stack ID arn:aws:cloudformation:us-east-1:106808901653:stack/autoscale-test/feea4f40-8c7d-11e9-bb2f-0a33c80c982c	Description Transit Gateway AutoS
Status CREATE_COMPLETE	Status reason -
Root stack -	Parent stack -
Created time 2019-06-11 20:20:38 UTC+0100	Deleted time -
Updated time	

9. When everything works

.You should see a “create_complete” status if the template had deployed correctly



autoscale-test
2019-06-11 20:20:38 UTC+0100
CREATE_COMPLETE

autoscale-test

Overview

Stack ID
arn:aws:cloudformation:us-east-1:106808901653:stack/autoscale-test/feea4f40-8c7d-11e9-bb2f-0a33c80c982c

Description
Transit Gateway AutoS

Status
CREATE_COMPLETE

Status reason
-

Root stack
-

Parent stack
-

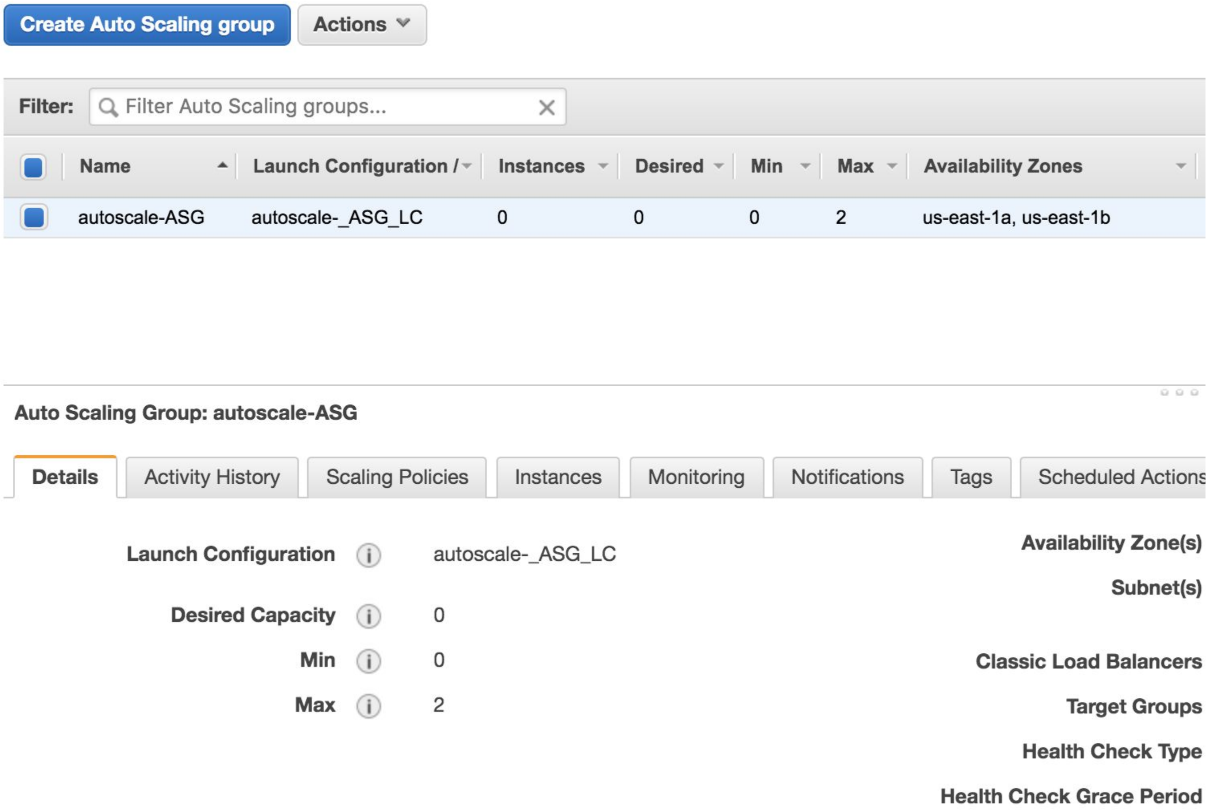
Created time
2019-06-11 20:20:38 UTC+0100

Deleted time
-

Updated time

The template will deploy and autoscale group and launch configuration with the “Desired” count set to zero. You can then add a firewall when you are ready

The template creates an autoscale group with a Desired Capacity and minimum of 0. When you wish to deploy the firewalls edit the autoscale configuration and modify these values match your desired number of firewalls. If you are deploying more that two firewalls you may be required to increase the resource limits on your account via AWS support.



Name	Launch Configuration /	Instances	Desired	Min	Max	Availability Zones
autoscale-ASG	autoscale-_ASG_LC	0	0	0	2	us-east-1a, us-east-1b

Auto Scaling Group: autoscale-ASG

Launch Configuration autoscale-_ASG_LC

Desired Capacity 0

Min 0

Max 2

Availability Zone(s)

Subnet(s)

Classic Load Balancers

Target Groups

Health Check Type

Health Check Grace Period