Public Facing Workload Configuration Sample

This page describes the steps needed to configure a public facing web application that is deployed within a workload AWS Account in the Secure Environment Accelerator (SEA).

The high-level steps are the following:

1) Create a SSL public certificate in AWS Certificate Manager.
2) Create a DNS entry for the web application.
3) Create Application Load Balancer Target Groups for the web application
4) Create an Application Load Balancer Rule to forward traffic to the Firewalls.
5) Configure the Firewalls.

The screenshots and steps in this page are specific to the Fortigate Firewalls.

Perimeter SEA AWS Account

SSL Certificate Configuration

1) Within the Perimeter SEA AWS Account, navigate to the Certificate Manager service.
2) Follow the steps to request a new public certificate. This will be used to support https for the web application. Note that the SEA deploys 'example' certificates, but these should not be used at the perimeter. Here's an example showing a wildcard cert.

Certificates

3) Navigate to the ALBs and select the Load Balancer that will support the incoming requests for the web application. In this example, it will be the 'Public-DevTest-perimeter-alb'.

ALBDevTest

4) Select the 'Public-DevTest-perimeter-alb' ALB and click the **View/edit certificates** link button.

ALBSSLConfigure

5) Click the + button and select the new SSL Certificate. Click **Add**.
6) Return back to the ALBs and select the 'Public-DevTest-perimeter-alb' ALB. Select the default HTTPS listener and click **Edit**

ALBDevTest

7) Change the Default SSL certificate to the newly created public cert and update the settings.

ALBDefaultSSL

ALB Target Group Configuration

1) Navigate to the EC2 Load Balancers and view the default Application Load Balancers (ALB).

ALBs

2) List the ALB Target Groups

ALB Target Groups

These are pairs of targets (one for each firewall) that direct traffic from the perimeter ALB to the firewall. The two pairs were created as part of the default configuration and provide health checks to the shared VPCs. For support a new web application, a new pair will be created. One for each firewall (i.e. one per AZ).

1) Click the **Create target group** button. (Note: This will be repeated for each Firewall).

2) Enter the following parameter values:

   - Target group name: Public-DevTest-SampleApp-azA
   - Protocol: HTTPS
   - Port: (pick an unused port on the Firewall). Example 7006
   - VPC: Perimeter_VPC

ALB New Target Group

5) When Registering a target, pick the instance that aligns with the Availability Zone (AZ) that is being configured. Example: Firewall_az[A|B]. If creating 'Public-DevTest-SampleApp-**azA**', then choose Firewall instance 'Firewall_**azA**'.

ALB New Target Group Register

6) Ensure that the port value is using the previous entered port value. Click the **Include as pending below**.

ALB New Target Group Register Instance

7) Click the **Create target group** button when ready.

8) Repeat for the additional firewalls.

ALB Listener Rule Configuration

1) Create a DNS entry for the web application that resolves to the perimeter ALB being configured. For example: webapplication.mydomain.ca resolves to 'Public-DevTest-perimeter-alb-1616856287.ca-central-1.elb.amazonaws.com'

2) Navigate to the ALBs and select the 'Public-DevTest-perimeter-alb' ALB. Click the **View/edit rules** link button.

ALBDevTest

3) Click the + button to create a new rule. Then click the + **Insert Rule** button.

ALBNeRule

4) Configure a match condition on **Host header...**. enter the value of the DNS entry for the web application.

ALBNeRuleHostHeader

ALBNeRuleHostHeader

5) Click the checkmark to update it.

6) Click the + **Add action** and select **Forward to...**

ALBNeRuleForwardTo

7) Configure both Targets using the ones previously created (one per firewall). Adjust for 50% load balanced traffic.

ALBNeRuleForwardToTargetGroup

ALBNeRuleForwardToTargetGroupLB

8) Click the checkmark to update and then click the **Save** button.

Fortigate Firewall Configuration

The following configuration will be executed per Firewall instance (twice with the default SEA configuration).

1) Log in to the firewall instance.
2) Switch the Virtual Domain (vdom) to **FG-traffic**.

FGVdeom

3. Navigate to **Policy & Objects** and select **Addresses**

FGVdeom

4. Create a new entry using the following parameter values:

- Name: Dev1-SampleWebApplication-ALB-FQDN
- Type: FQDN
- FQDN: (use the DNS value of the internal load balancer in front of the web application)
- Interface: tgw-vpn1

FGNewAddress

5. After saving the entry, refresh the Address grid and verify that the row colour is white.

FGAddress2

6. Navigate to **Policy & Objects** and select **Virtual IPs**

FGVIPs

7. Make note of the used ip address in the Details column. In the example above "100.96.250.22".

8. Click the CLI command icon in the top right corner. Note that the following must be done using the CLI.

FGVIPsCLI

FGVIPsCLI1

9) Update the following script template replacing values for the following: name, extip, mapped-addr, extport

```
config firewall vip
edit "Dev1-SampleWebApplication-ALB"
        set type fqdn
        set extip 100.96.250.22
        set extintf "port1"
        set portforward enable
        set mapped-addr "Dev1-SampleWebApplication-ALB-FQDN"
        set extport 7006
        set mappedport 443
    next
end
```

FGVIPsCLI2

10) Returning back to the UI interface shows the new entry.

FGVIPsCLI3

11) Navigate to **Policy & Objects** and select **IPv4 Policy** and expand **public (port1)**

FGIPv4Policy1

12) Locate the desired policy (ex: Dev-Test #8 in the example below). Right-click and click **Edit**.

FGIPv4Policy2

13) Locate the **Destination** field entry and click the + button.

FGIPv4Policy3

14) Locate the newly created VirtualIP entry (ex: Dev1-SampleWbApplication-ALB) and save the changes. NOTE: The entry is NOT the Address/FQDN entry.

FGIPv4Policy4

15) After refreshing the page, the row background should be white, and the new destination is visible.

FGIPv4Policy5