Ignite18-US HOW-11-GoogleCloud

Hands-On Workshop

Cloud Security for

Google Cloud Platform

http://www.paloaltonetworks.com

Last Update: 20180517

Table of Contents

Activity 0 –Login

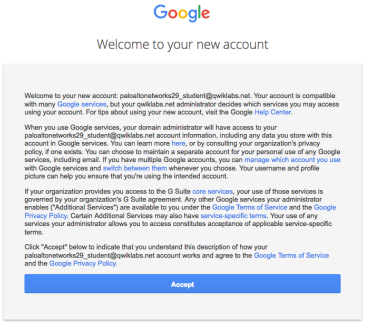
In this activity, you will:

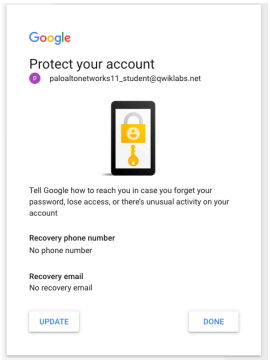
Log in to the GCP Console using the account provided.

Task 1 – Log into GCP Console

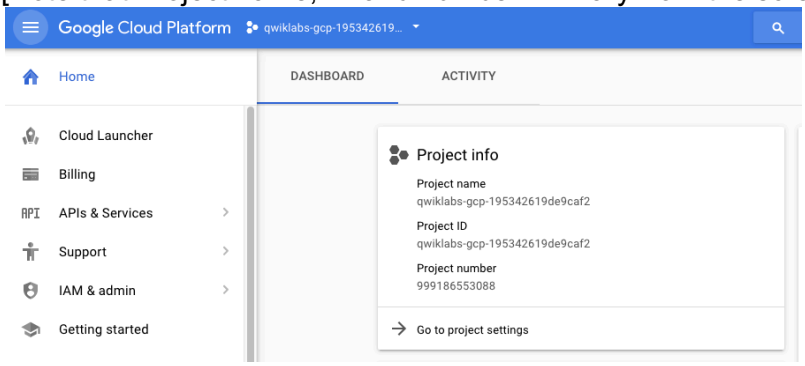
Step 1: Browse to http://console.cloud.google.com and login. If you do not have an account for GCP browse here https://console.cloud.google.com/freetrial

Step 2: Accept the Service agreement and click **Done** to login to the account. You can agree to the updated Term and Services as well.





Step 3: Once you have successfully logged in, you should have full access to the Google Cloud Platform console. [Note that Project name, ID and number will vary from the screenshots.]



End of Activity 0

Activity 1 – Launch Deployment Manager Template

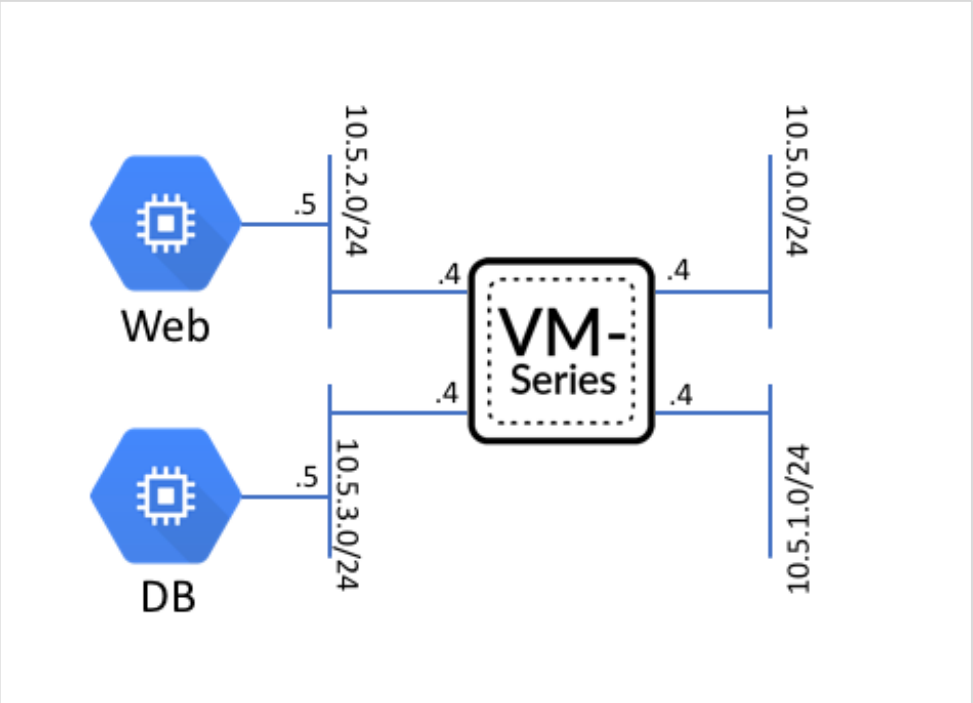
GCP Deployment Manager Templates, are Python (or Jinja) files that can launch nearly all GCP resources including VPCs, subnets, security groups, route tables, plus many more. Templates are used for ease of deployment.

This lab will demonstrate how to deploy a sample template that launches everything that is shown below. This includes, a WordPress server, a MySQL server, a VM-Series firewall and the subnets. In addition, the firewall uses a native bootstrapping feature that allows for additional configuration of the firewall (such as routes, security policies, etc.) Once the sample template has been deployed, the network topology should align with the following:

For more information on Templates refer to Google's documentation

https://cloud.google.com/deployment-manager/docs/how-to#adding-templates

There are also many sample templates available here
https://github.com/GoogleCloudPlatform/deploymentmanager-samples/tree/master/templates



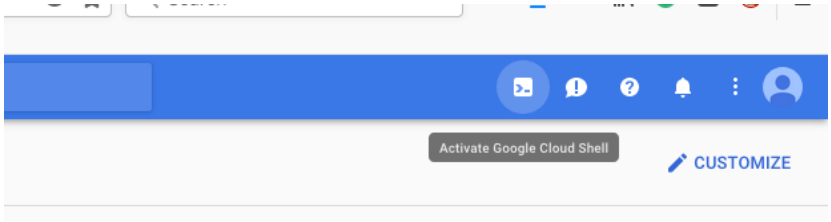
Template Instances used
When using the sample template in this lab, the following machine types are used:

Instance name	Machine Type
WordPress Web Server	f1-micro
WordPress DB Server	f1-micro
VM Series Firewall	n1-standard-4

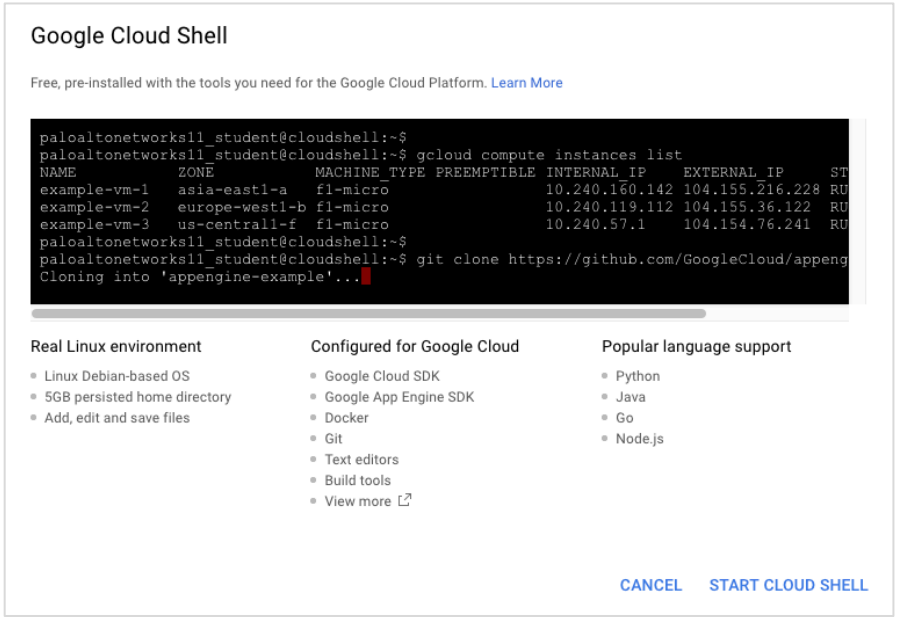
Task 1- Launch Template with Google Cloud Shell

Google Cloud Shell is a shell environment for managing resources hosted on Google Cloud Platform. You can perform some basic operations in Google Cloud Shell. You can learn more about it here.

Step 1: Open Google Cloud Shell using the button on the upper right-hand corner.

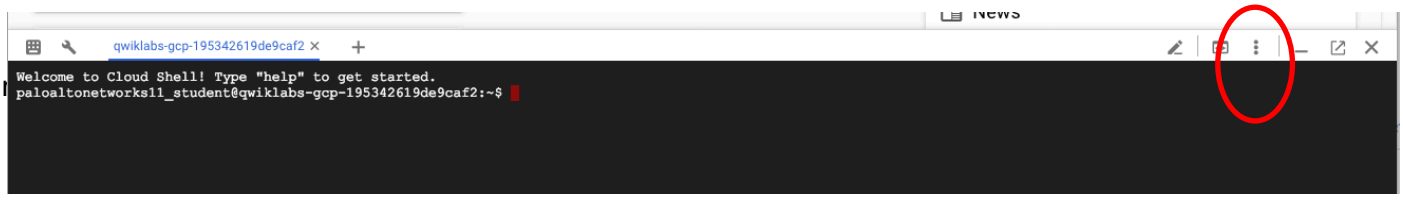


Step 2: Click **Start Cloud Shell** to open the Cloud Shell window below. Note, this could take a few minutes for Google Cloud Shell to be ready.

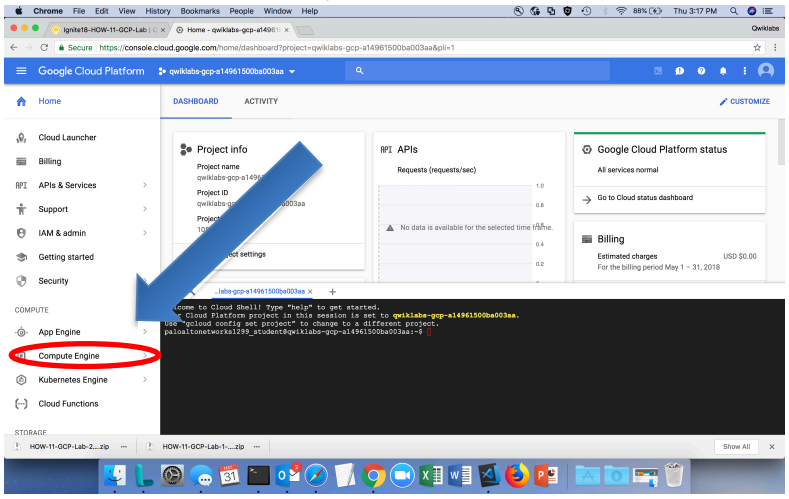


Once cloud shell is started and open click on the 3-dot ellipse to upload the deployment files

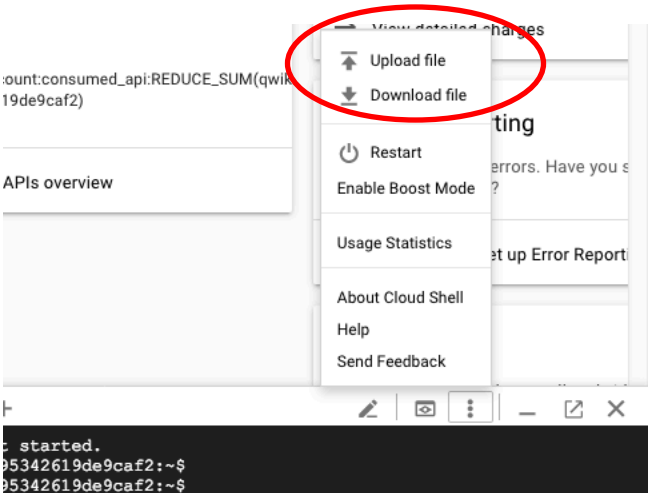
Ign



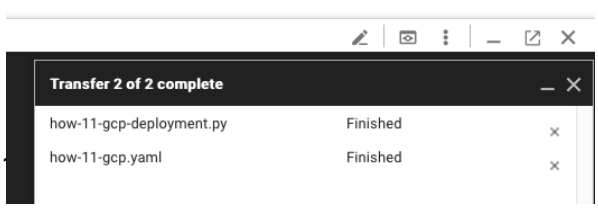
If you **DO NOT SEE** the 3-dot ellipse and your screen looks like this. Click on Compute Engine and the screen will reformat and you will see the 3-dots.



Step 3: Click the **Upload File** in upper right-hand corner of the Google Cloud Shell console.



Step 4: Upload the **how-11-gcp-lab1.yaml** and **how-11-gcp-lab1.py** file. Then close the Transfer window.



Step 5: In Cloud Shell prompt, you can use the **ls** command to confirm files have been uploaded.

```
paloaltonetworks29_student@qwiklabs-gcp-10607a6a0e3e9206:~$ ls
how-11-gcp-deployment.py  how-11-gcp.yaml  README-cloudshell.txt
paloaltonetworks29_student@qwiklabs-gcp-10607a6a0e3e9206:~$
```

Step 6: Create a deployment using the following gcloud command:

gcloud deployment-manager deployments create <your-last-name> --config how-11-gcp-lab1.yaml

We recommend using your last name as the deployment name just for easy identification. You can use any name. This will run the deployment for about 3minutes. However, the firewall is still building the overall process take about 10-13 minutes.

```
paloaltonetworks30_student@qwiklabs-gcp-da864b98801115f1:~$ gcloud deployment-manager deployments create your-last-name --config how-11-gcp.yaml
The fingerprint of the deployment is _v-6oMTxFQlbaeRD9B9Gsg==
Waiting for create [operation-1524181038286-56a3c0d5c44b2-92b286a5-73bf7f29]...\
```

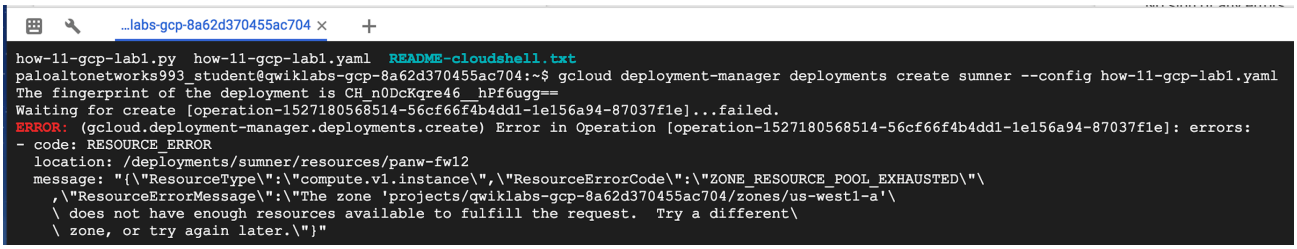
Step 7: When the deployment is completed, you will see the creation completed message for each operation.

```
paloaltonetworks31_student@qwiklabs-gcp-a04fda96beb99bb7:~$ gcloud deployment-manager deployments
The fingerprint of the deployment is SAqvkNSEu0viVd34sAAXgA==
Waiting for create [operation-1524186866688-56a3d68c2a002-54964a6b-a0f86b63]...done.
Create operation operation-1524186866688-56a3d68c2a002-54964a6b-a0f86b63 completed successfully.
NAME                TYPE                    STATE      ERRORS  INTENT
db-firewall90       compute.v1.firewall     COMPLETED  []
db-network90        compute.v1.network      COMPLETED  []
db-route90          compute.v1.route        COMPLETED  []
db-subnet90         compute.v1.subnetwork   COMPLETED  []
dbserver90          compute.v1.instance     COMPLETED  []
mgmt-firewall90     compute.v1.firewall     COMPLETED  []
mgmt-network90      compute.v1.network      COMPLETED  []
mgmt-subnet90       compute.v1.subnetwork   COMPLETED  []
panw-fw90           compute.v1.instance     COMPLETED  []
public-firewall90   compute.v1.firewall     COMPLETED  []
public-network90    compute.v1.network      COMPLETED  []
public-subnet90     compute.v1.subnetwork   COMPLETED  []
web-firewall90      compute.v1.firewall     COMPLETED  []
web-network90       compute.v1.network      COMPLETED  []
web-route90         compute.v1.route        COMPLETED  []
web-subnet90        compute.v1.subnetwork   COMPLETED  []
webserver90         compute.v1.instance     COMPLETED  []
paloaltonetworks31_student@qwiklabs-gcp-a04fda96beb99bb7:~$
```

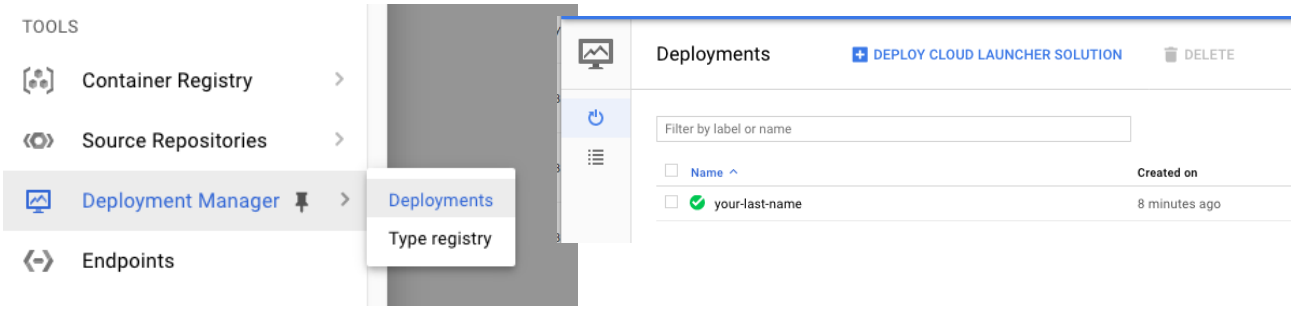
NOTE: If you receive a resource failure stating “ZONE_RESOURCE_POOL_EXHAUSTED”. You must delete your Deployment Manager Installation before trying again. To delete the deployment, use the command below in the cloud shell.

gcloud deployment-manager deployments delete <your-last-name>

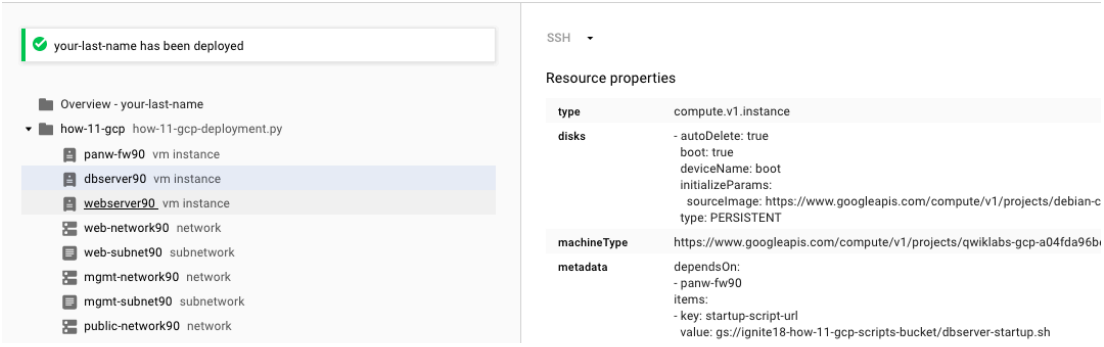
Screen Shot of Resource Error Below



Step 8: You can also go to the Deployment Manager > Manager service to view the deployment.



Step 9: Click the deployment that you have created and review the items that are created by the deployment manager.



Custom Scripts/Linux Extensions

The template deploys Linux extensions to configure the firewall, web server (with Apache and WordPress) and database server (MySQL). Linux extensions are resources that can be used to configure Linux VMs. Each custom script downloads and runs a specific script (found in the Github repo) that configures a specific VM. The web-vm-custom script configures the firewall and the web server. The db-vm-custom script configures the database server

Note: The entire template should take about 10 minutes to deploy and be fully functional

End of Activity 1

Activity 2 – Review VPC and Resource Configurations

In this activity, you will:

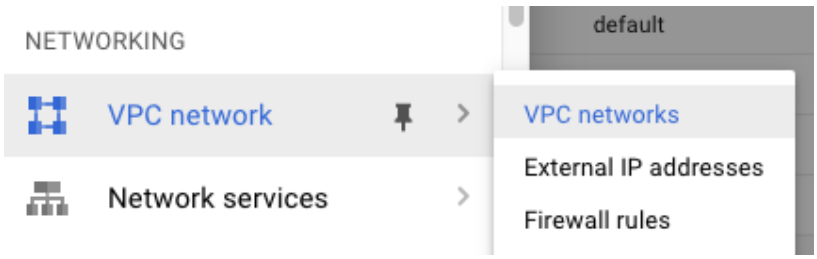
Review the VPC created with Deployment Manager Template

Review the GCE instances and the configuration created

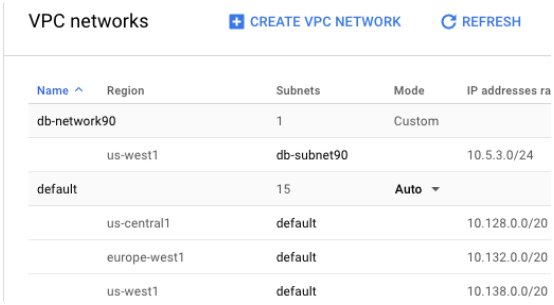
Task 1 - Review VPC Networks

Let's review all the components we have launched in the VPC based on the deployment manager file.

Step 1: The VPC networks can be accessed via ***VPC Networks > VPC Networks***. There you should see all networks created in your project:



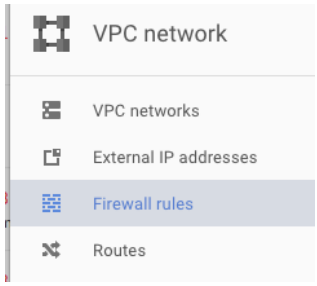
Step 2: The VPC networks created by the deployment will have a random two-digit number to it. The ***default*** network is created by default.

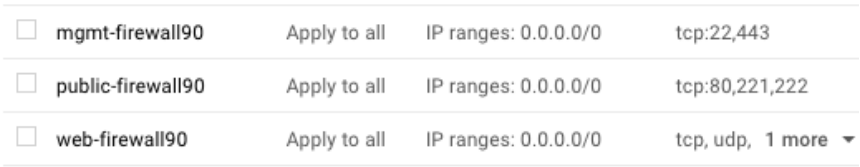


Step 3: Three networks are created by the deployment. They are **db-network##**, **mgmt.-network##**, **public-network## and web-network##**.

Note: Each network corresponds to an interface on the firewall. Although, you can create multiple subnets in a single network, a VM instance cannot have more than one interface on the same network.

Step 4: Click on the ***Firewall rules*** to review the rules that are created by the deployment.

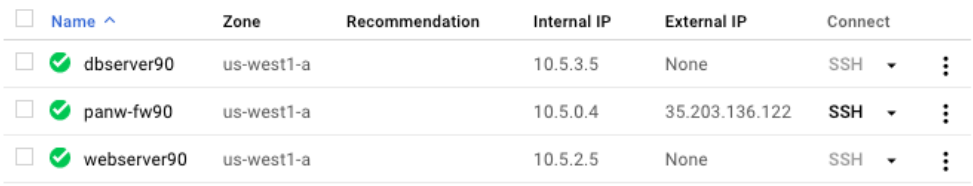




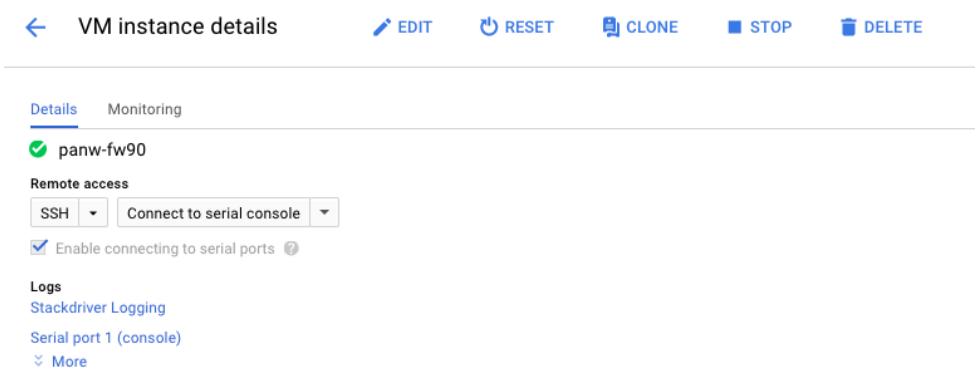
Task 2 - Review Compute Engine

We will review the VM instances created by the deployment in this task.

Step 1: Go to the ***Compute Engine > VM instances***. There are at least 3 instances created by in this deployment, Web server, a DB server and a VM-series firewall.



Step 2: Click the ***panw-fw##*** instance to view the details.

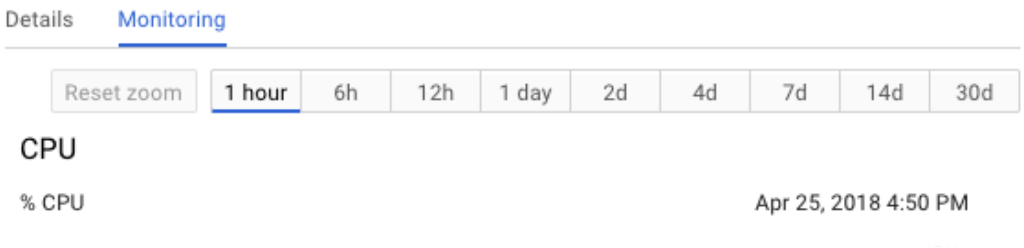


Step 3: Note that there are four network interfaces created for the firewall. The two external IP addresses will be used in the next task and we will come back here to get these two IPs.

Network interfaces

Network	Subnetwork	Primary internal IP	Alias IP ranges	External IP	IP forwarding
mgmt-network82	mgmt-subnet82	10.5.0.4	—	35.203.170.133 (ephemeral)	On
public-network82	public-subnet82	10.5.1.4	—	35.185.255.13 (ephemeral)	
web-network82	web-subnet82	10.5.2.4	—	None	
db-network82	db-subnet82	10.5.3.4	—	None	

Step 4: Click the ***Monitoring*** tab to review some of the basic metrics monitored by GCP. Note that this VM is quite new, so you may not see a lot of data at this point.



Note: The VM-Series firewall on GCP can publish native PAN-OS metrics to Stackdriver, which you can use to monitor the firewalls. These metrics allow you to assess performance and usage patterns that you can use to take action for launching or terminating instances of the VM-Series firewalls. We will not go into the details at this point but you can discuss with instructor if you want to know more about how to enable Stackdriver monitoring on the VM-Series firewall.

Step 5: Go to **VM *Instances*** and select the ***dbserver##*** instance to review the details for that instance. Under the network interfaces, note that there is no public IP for the dbserver.

Network interfaces

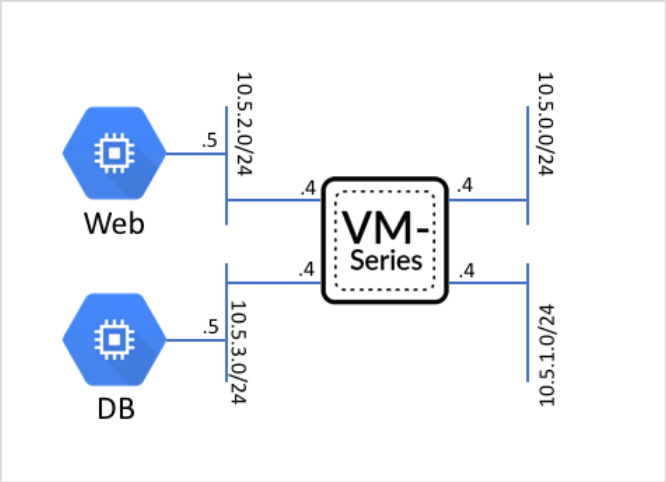
Network	Subnetwork	Primary internal IP	Alias IP ranges	External IP	IP forwarding
db-network82	db-subnet82	10.5.3.5	—	None	Off

Step 6: Go to **VM *Instances*** and select the web***server##*** instance to review the details for that instance. Under the network interfaces, note that there is no public IP for the webserver as well.

Network interfaces

Network	Subnetwork	Primary internal IP	Alias IP ranges	External IP	IP forwarding
web-network82	web-subnet82	10.5.2.5	—	None	Off

All of the VPC configuration should match the topology shown previously:



End of Activity 2

Activity 3 - Review PAN-OS WebUI

In this activity, you will:

- Login to the VM-Series firewall
- Review key portions of the firewall configurations

Task 1 - Access the firewall

Step 1: Once the template deployment is complete, the firewall will show a green checkmark indicating it is running. The firewall may be still executing the bootstrap function.

NOTE: Bootstrapping a VM-Series firewall takes approximately 15 minutes. So once the template has been deployed successfully, it may be a while before the firewall is up and you are able to log into the firewall.

Step 2: Identify the External IP address in the mgmt-network##, that will be the public IP where you can access the firewall on.

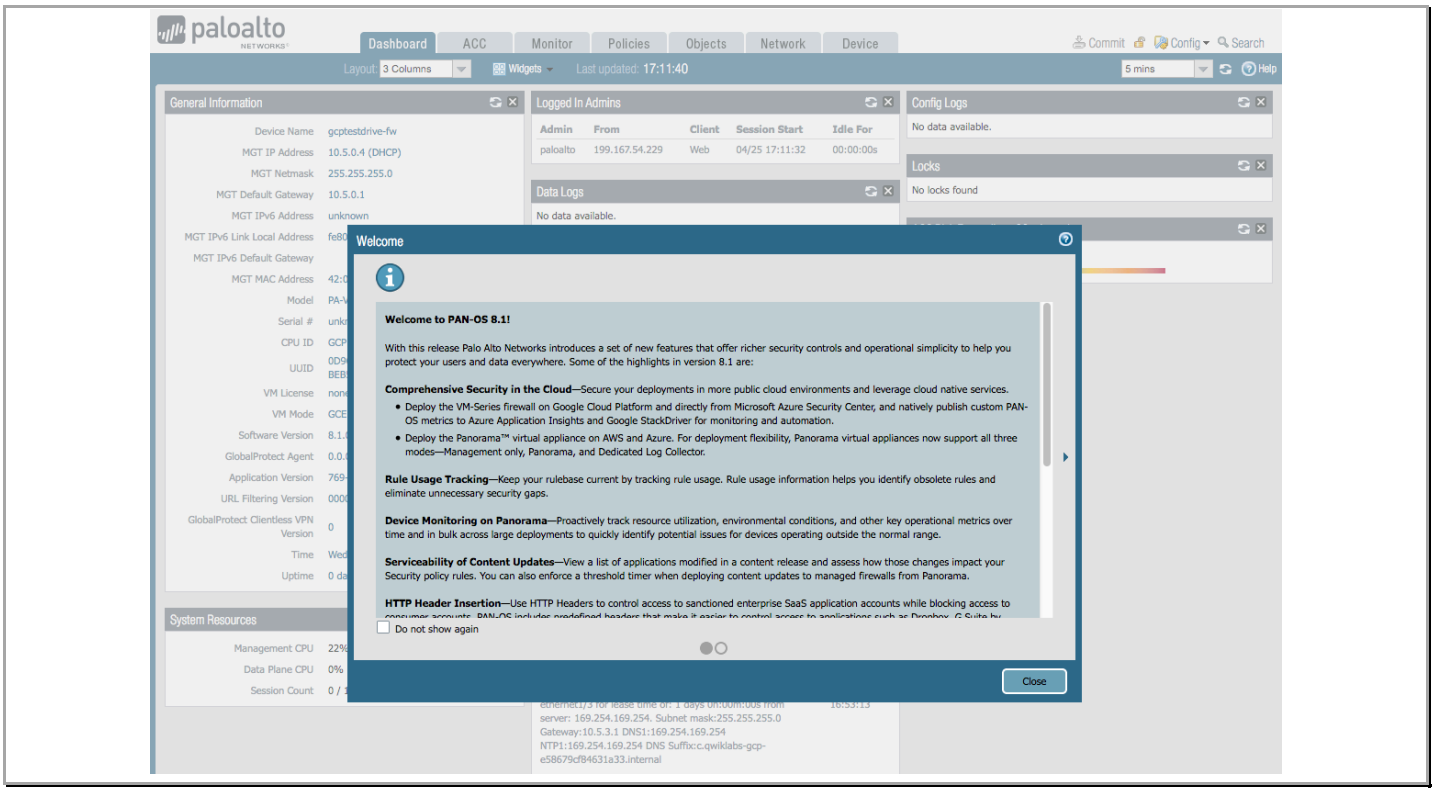
Network interfaces

Network	Subnetwork	Primary internal IP	Alias IP ranges	External IP	IP forwarding
mgmt-network82	mgmt-subnet82	10.5.0.4	—	35.203.170.133 (ephemeral)	On
public-network82	public-subnet82	10.5.1.4	—	35.185.255.13 (ephemeral)	

In a new browser tab, open a https:// using the mgmt external IP address.

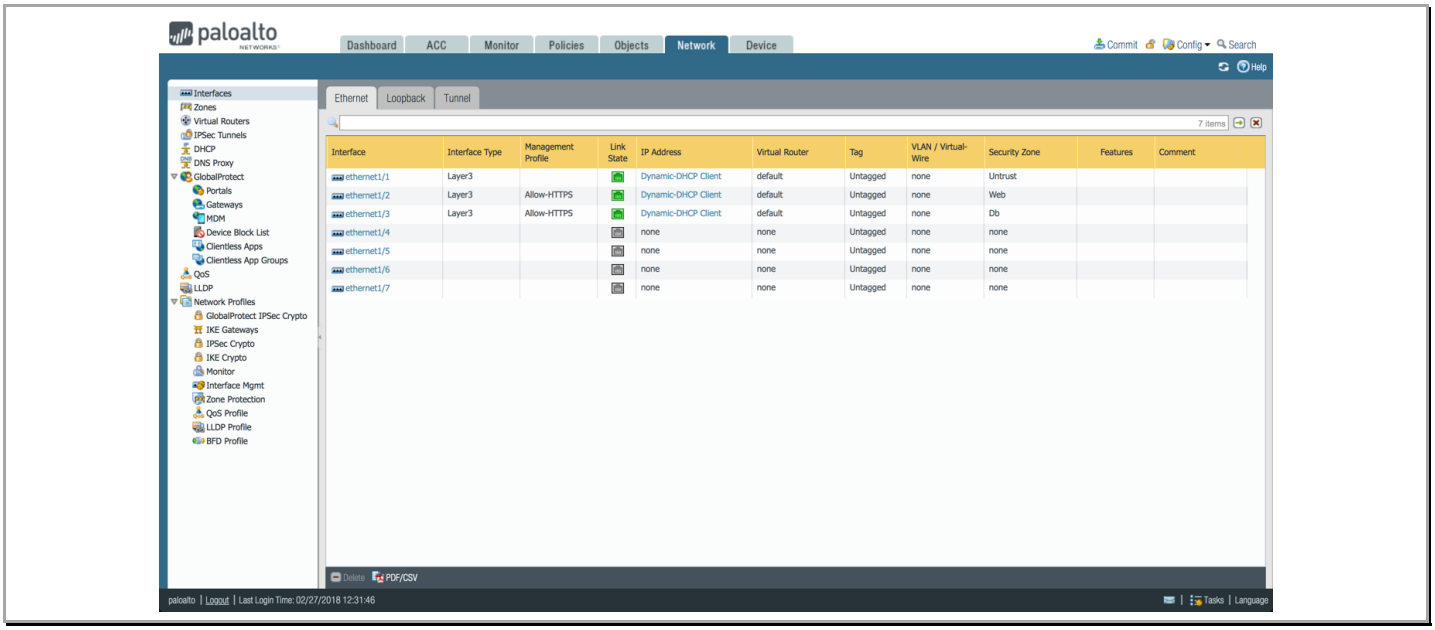
Step 3: You should now be able to login to the firewall using:

username: **paloalto**
password: **Pal0Alt0@123**

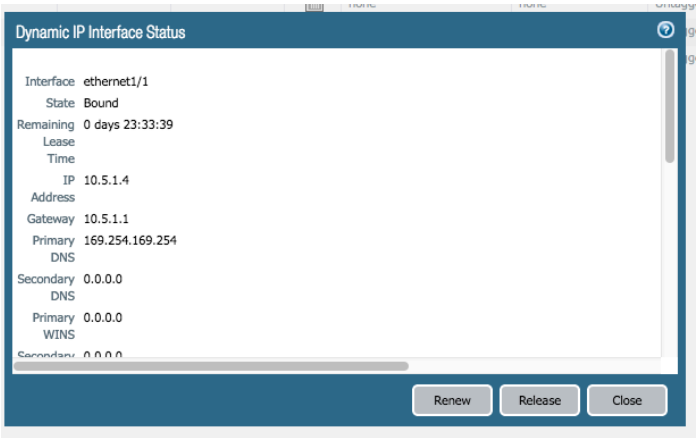


Task 2 – Review the Networks and Polices tab

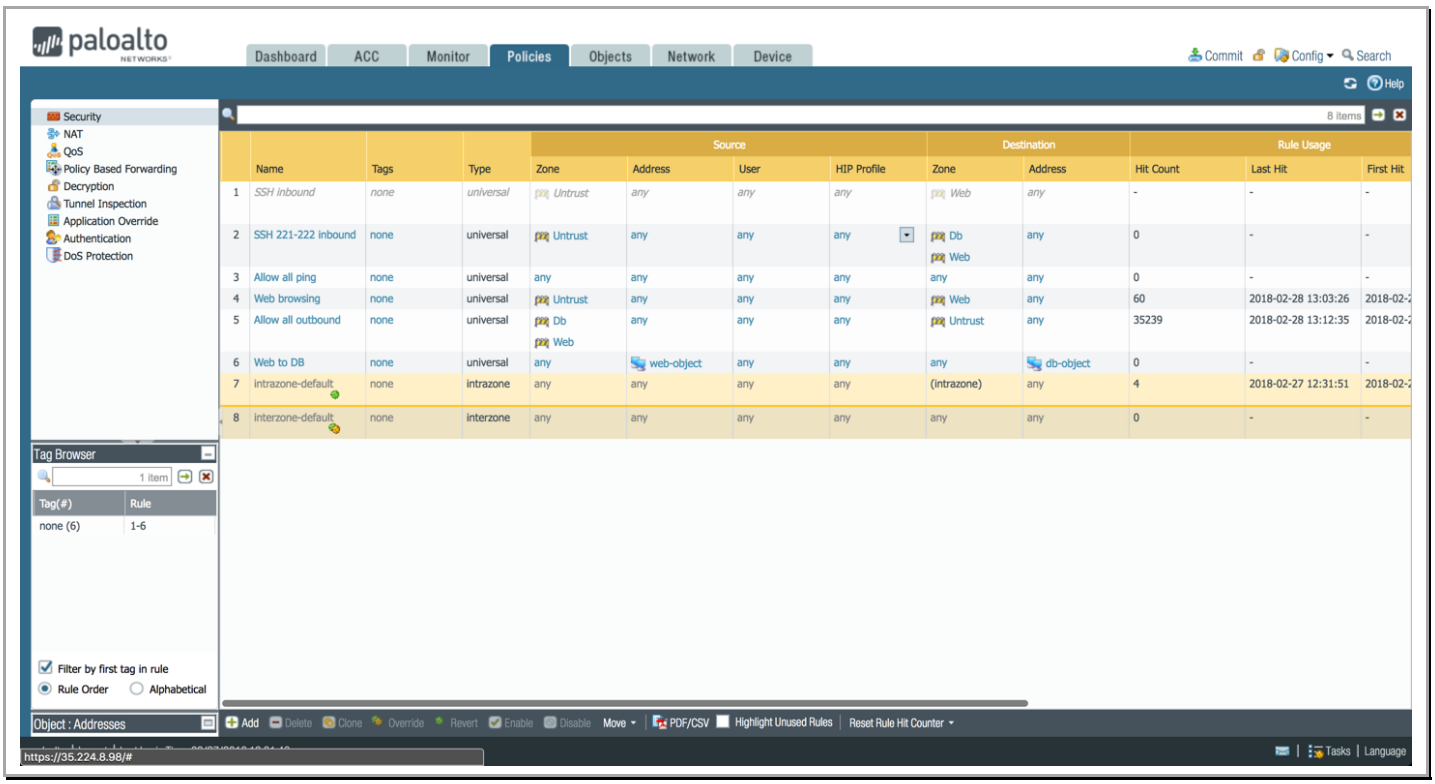
Step 1: Click the ***Networks*** tab to see the interfaces to zone mappings:



Step 2: Click the ***Dynamic-DHCP-Client*** under the IP Address to see the status of each interface.

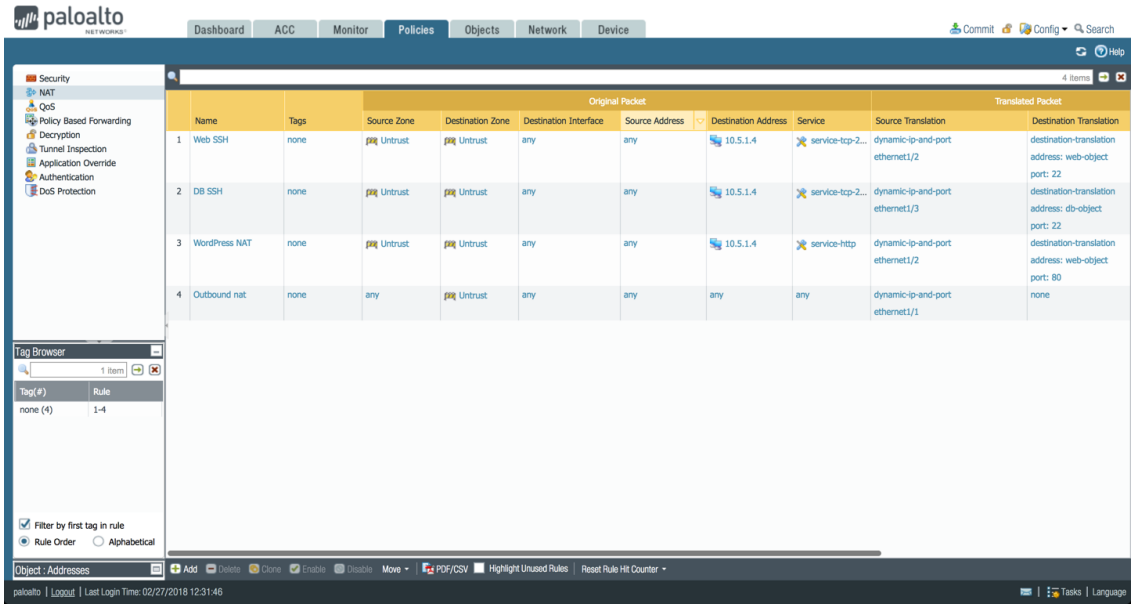


Step 3: In the ***Policies*** tab you can review the security policies:

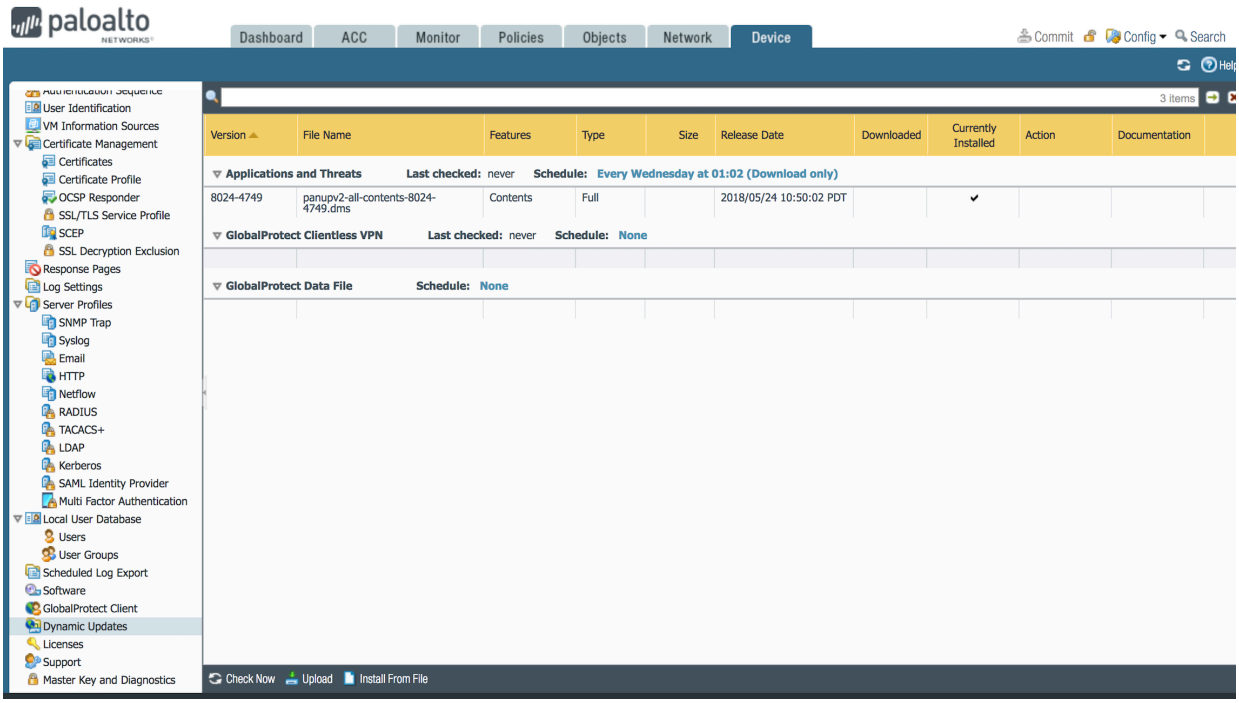


These policies are defined to allow ssh access on ports 221 and 222 to the web and db server respectively (for troubleshooting purposes), secures N/S traffic and E/W traffic between zones.

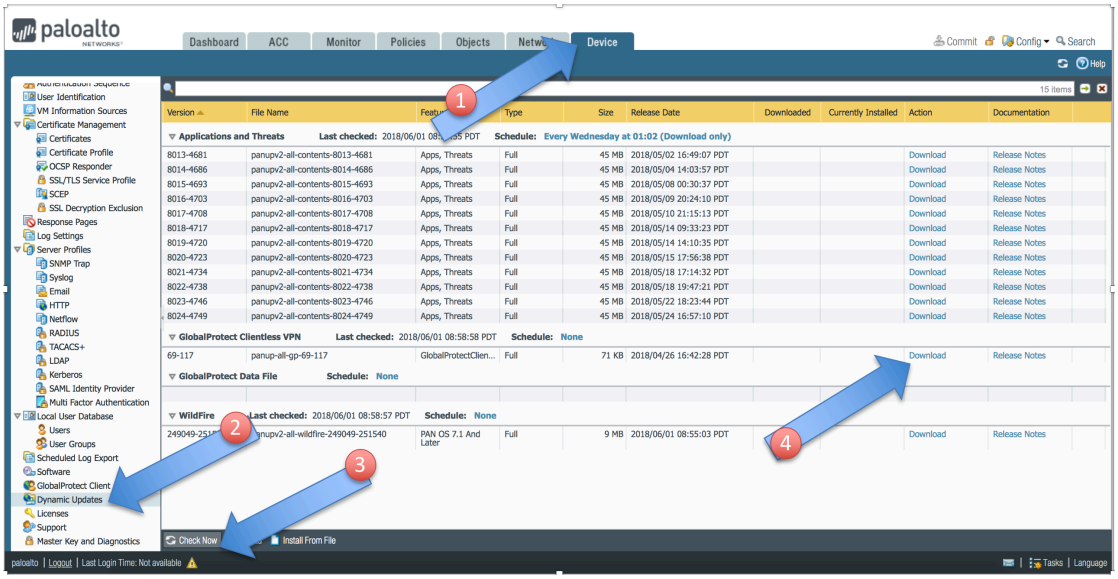
Step 4: Select NAT on the left menu, review the NAT policies allow for ssh access to the web and db servers as well as directing web traffic to the web server only. There is also a rule for source NAT from web and db servers to the outside world.

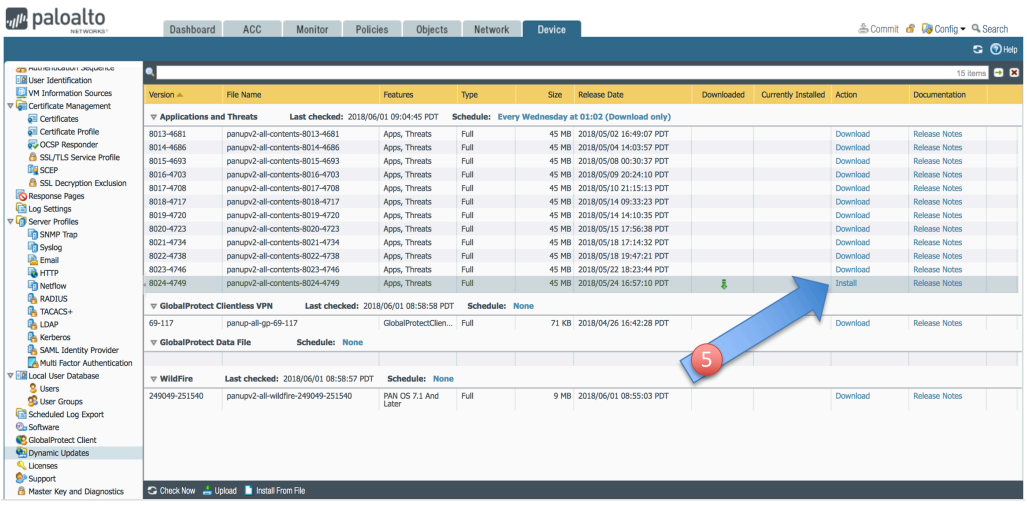


Step 5: Check to make sure that the Dynamic Updates installed during the bootstrap process. You should see similar content in your Dynamic Updates area.



Step 6: If you do not have any dynamic updates or would like to make sure you have the latest Dynamic Updates complete the following. Select the 1.Device Tab in the Firewall. On the left menu screen select 2.Dynamic Updates. Note there are no threat updates. Select 3."Check Now" in the lower left corner of the Primary Screen. 4.Download the latest threat content update. Once downloaded you must select 5.install for the content to be active on your VM-Series FW.





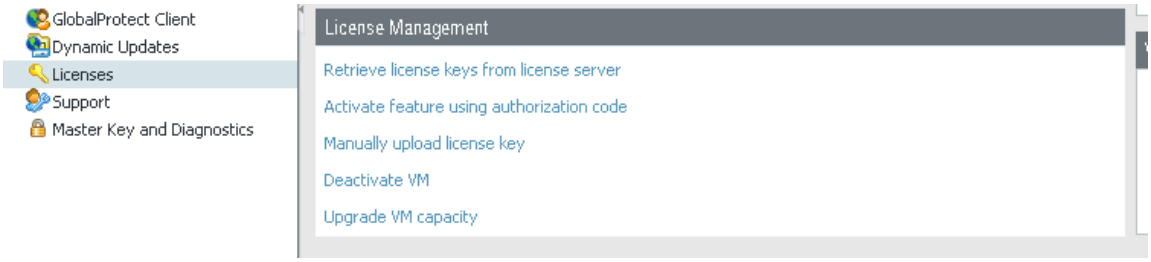
Task 3 – (Optional Not Needed for Qwiklab UTD) License the firewall [BYOL]

Skip to Activity 4 unless otherwise instructed.

If BYOL – Bring Your Own License instance is used, you will need to activate the firewall to fully utilize the all the features. The steps below demonstration on to license a BYOL firewall instance through a registered Authorization code (provided by the instructor).

Note: PAYG – Pay As You Go licensing model is also available in Google Cloud Platform, where license and subscription fees can be included as part of the charges.

Step 1: Go to the ***Device*** tab and the ***License*** node, under the ***License Management*** widget.



Step 2: Click Activate feature using authorization code.

Step 3: Enter the authorization code if applicable. The firewall will reboot to install the new license. [The authorization code used in the lab will be disabled shortly after this lab.]

End of Activity 3

Activity 4 – Safely Enable Applications

In this activity, you will:

- Generate traffic on the firewall and review the traffic log
- Edit the security policy to allow inter-tier application traffic

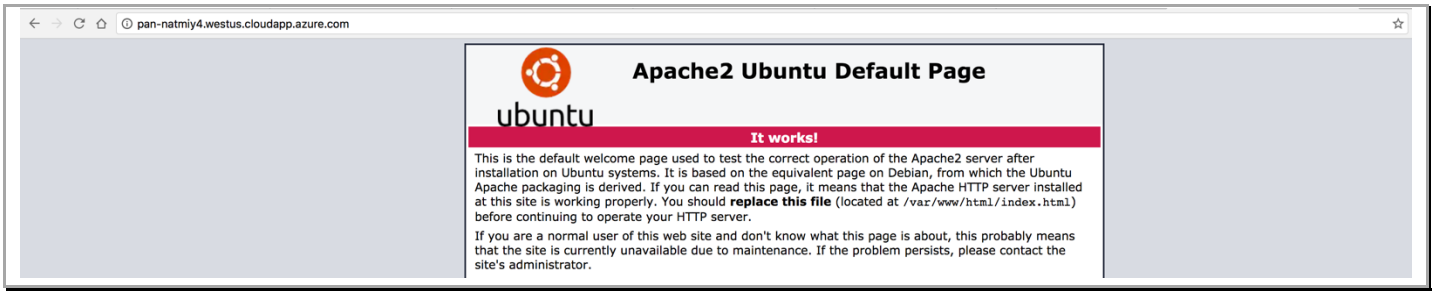
Task 1 - Verify Static Content on Web Server

Step 1: Identify the External IP address of the public-network## for Firewall, that will be the public IP of the web server.

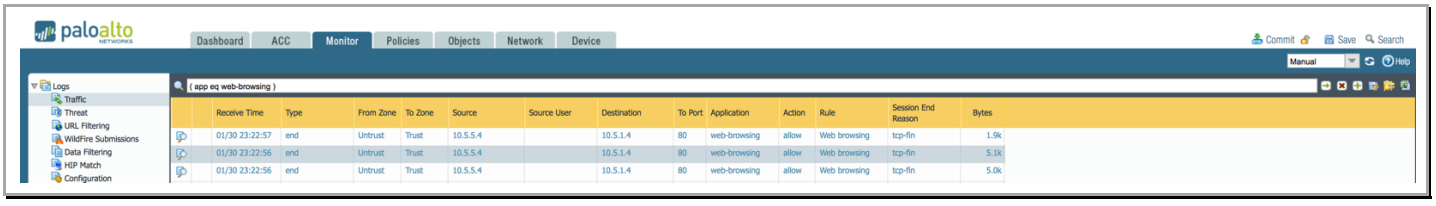
Network interfaces

Network	Subnetwork	Primary internal IP	Alias IP ranges	External IP	IP forwarding
mgmt-network82	mgmt-subnet82	10.5.0.4	—	35.203.170.133 (ephemeral)	On
public-network82	public-subnet82	10.5.1.4	—	35.185.255.13 (ephemeral)	

Step 2: Using the public-network## external IP on the firewall to access the static content of the webserver and you should see the screenshot below:



Step 3: Check firewall logs under the Monitor tab to verify that the traffic is passing through the firewall:



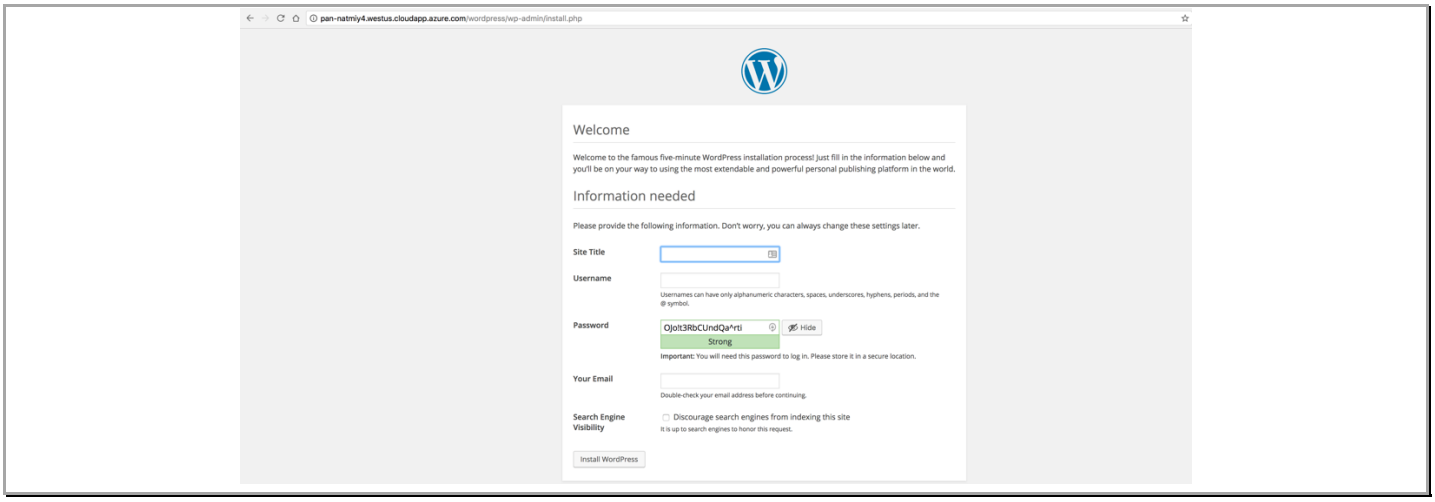
Task 2 - Verify Dynamic Content on Web Server

In this task, you will generate a WordPress content request from your web browser that will trigger a database query to the MySQL server. Like many web-based applications, WordPress uses a backend database to create, store, and retrieve dynamic content. You will use the WordPress application to show exactly this type of behavior and demonstrate how the VM-Series firewall will secure this traffic.

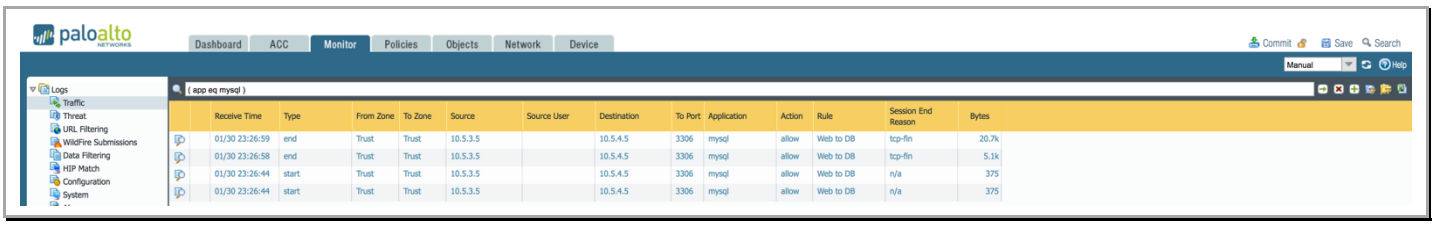
In the browser, go to the WordPress server (http://public-network-external-IP/wordpress)

And you should see the WordPress welcome page.

Note: You don't need to actually configure the new WordPress server. In its initial, un-configured state, it will generate the traffic we need to test the VM-Series firewall.



Now, head back to the firewall and verify that the traffic did indeed go through the firewall from web to db:



End of Activity 4

Activity 5 - Safe Application Enablement

In this activity, you will:

- Generate two simulated east/west (web tier to database tier) attacks
- Monitor the firewall logs to see the results of the attacks

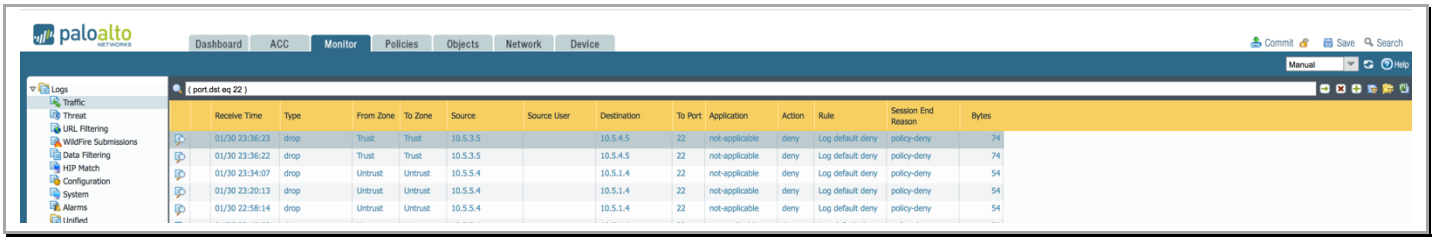
Task 1 - Attempt to SSH from the web server to the DB server

This task will simulate a compromised web server that is being used to attack the database. This is a common attack strategy of getting a foothold on the web front-end server and then expanding to the other application tiers with the ultimate goal of accessing all data in the database.

Go to (http:// public-network-external-IP /sql-attack.html) and simulate a web to db ssh attempt by clicking on the **LAUNCH WEB TO DB SSH ATTEMPT.**

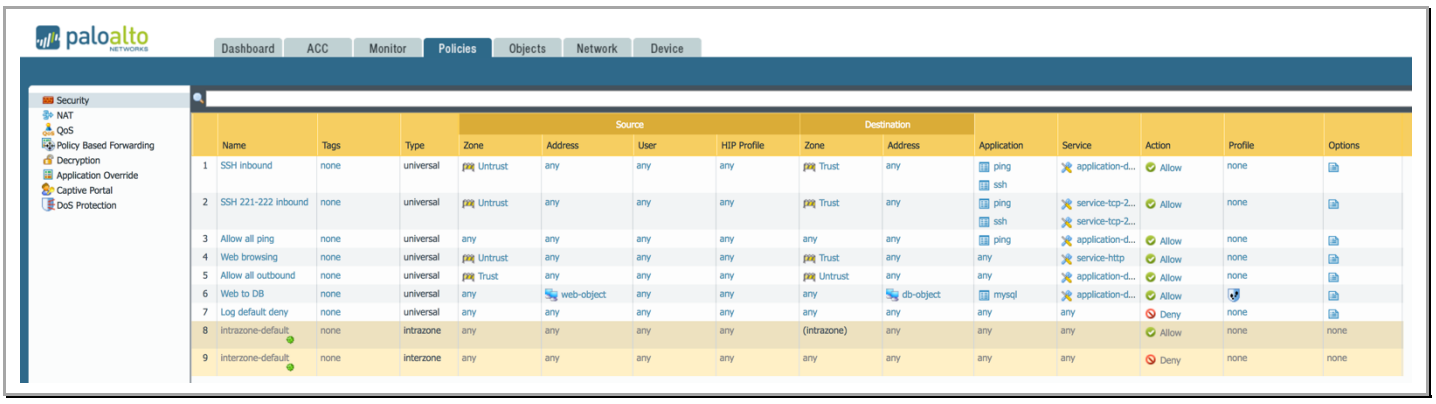
LAUNCH WEB TO DB SSH ATTEMPT

This launches a CGI script that attempts to ssh as root to the db server from the web server. Now return to the firewall's monitor tab to note the failed traffic:

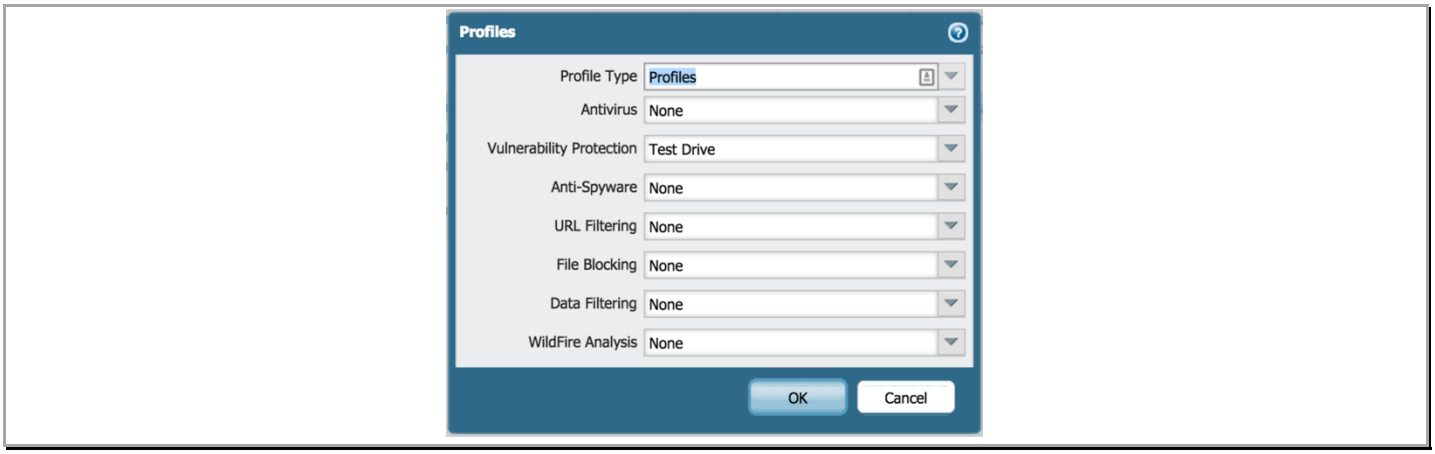


Task 2 - Trigger the SQL brute force attack and review logs

On the firewall's security policies tab, under Security, Rule 6, you will notice that the web to db traffic is protected further by a vulnerability profile:



Now click on the icon in the Profile column and you will see all the threat protection profiles

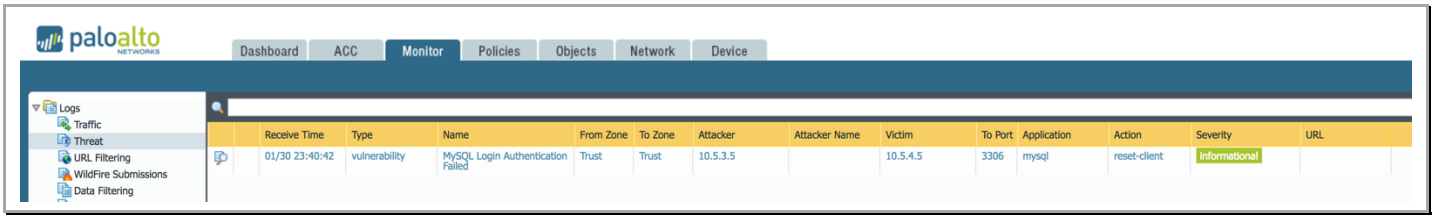


Note the Vulnerability Protection profile. This is a custom profile created just for this lab. It is part of the default vulnerability protection profile but is called out separately for the purpose of this demo environment.

Let's finally trigger the attack. Head back to the sql-attack.html page at (http://WebserverURL/sql-attack.html)

Click on Launch Brute Force Attack to start a script that will generate multiple failed MySQL authentication attempts.

This will launch some scripted attacks on the SQL server and use the pre-configured threat protection to show and block those attacks on the VM-Series firewall. Now return to the firewall and click the Monitor tab and then click on Threats in the left-hand pane under Logs and notice the new vulnerability log message regarding the failed MySQL events:



The CGI script you launched above attempted to login to the MySQL database multiple times with an incorrect password. The VM-Series firewall saw this activity and using the vulnerability profile, reset the connection and logged the activity.

Task 3 - Lab Cleanup

If you have completed the above lab activities, we recommend deleting the deployment before moving on to the next activity.

Step 1: (optional) If you elected to work through Lab1, delete it prior to proceeding with Lab 2 by issuing the following command:

gcloud deployment-manager deployments delete <your-last-name>

End of Activity 5

Activity 6 – LAB 2-- Deploy Firewalls with Load Balancer

GCP Deployment Manager Templates, are Python (or Jinja) files that can launch nearly all GCP resources including VPC networks, subnets, security groups, route tables, etc. Templates are used for ease of deployment and are key to any auto-scaling environment.

This lab will demonstrate how to deploy a sample template that launches everything that is shown below. This includes:

1 – External Load Balancer (HTTP)

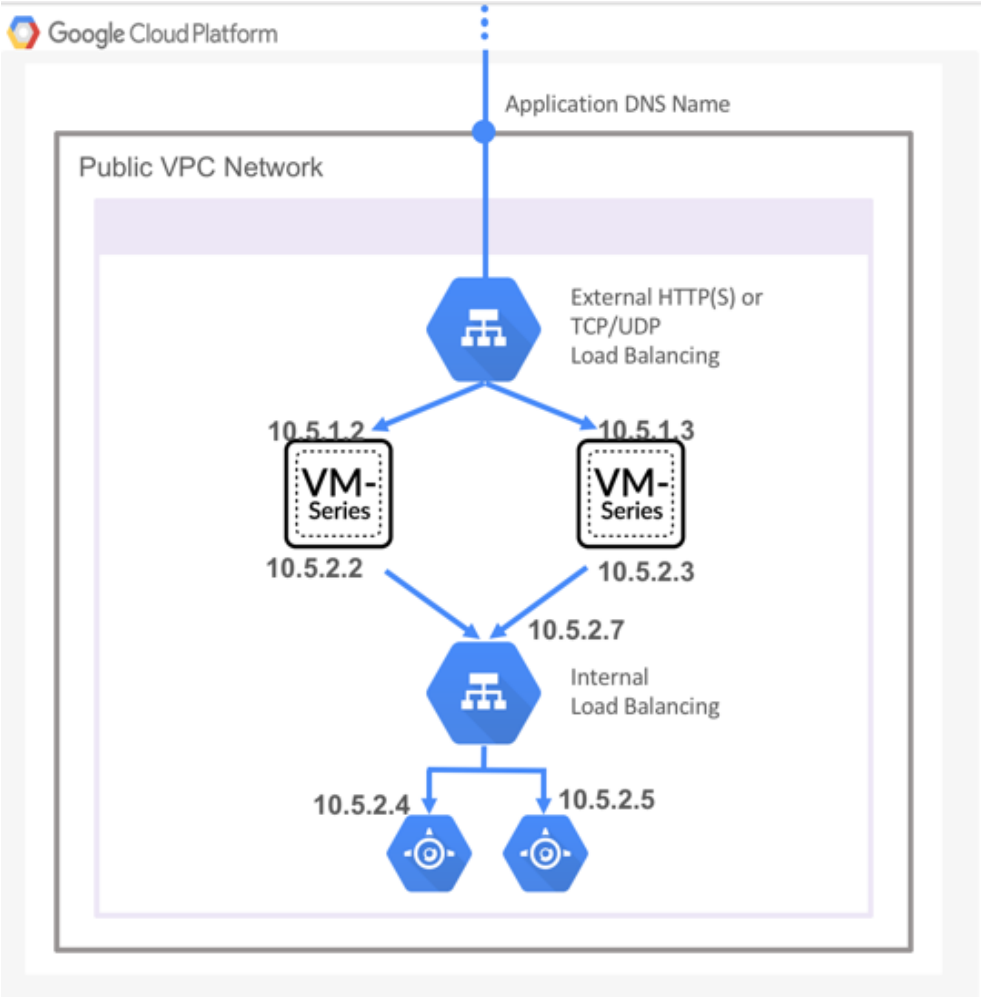
2 - VM-Series firewalls

1 – Internal Load Balancer (TCP/80)

2 – Webservers

Supporting infrastructure (VPC Network, Subnets, GCP Firewall Rules, etc.)

The firewall uses bootstrapping for a basic configuration (interface settings, administrative account, etc.) Once the sample template has been deployed, the network topology should align with the following:



For more information on Templates refer to Google's documentation

https://cloud.google.com/deployment-manager/docs/how-to#adding-templates

There are also many sample templates available here
https://github.com/GoogleCloudPlatform/deploymentmanager-samples/tree/master/templates

Template Instances used

When using the sample template in this lab, the following machine types are used:

Instance name	Machine Type
Web Servers	n1-standard-1
VM Series Firewalls	n1-standard-4

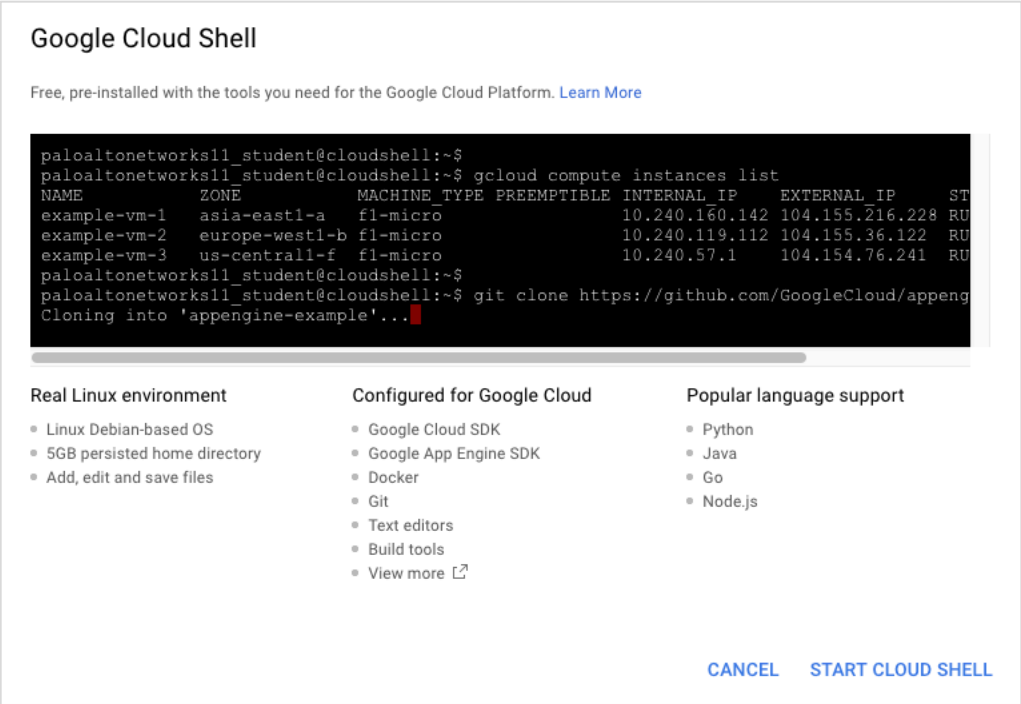
Task 1 –Launch Template with Google Cloud Shell

Google Cloud Shell is a shell environment for managing resources hosted on Google Cloud Platform. You can perform some basic operations in Google Cloud Shell. You can learn more about it here.

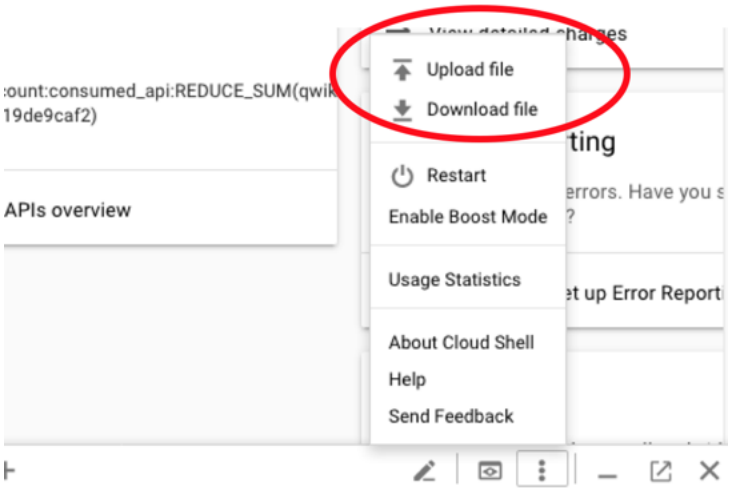
Step 2: (optional) If you elected to work through Lab1, delete it prior to proceeding with Lab2 by issuing the following command:

gcloud deployment-manager deployments delete <your-last-name>

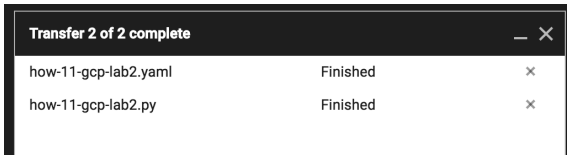
Step 3: Click the start cloud shell icon in the upper right corner of the browser window ( ) to open the Cloud Shell window below. Note that it may take a few minutes for Google Cloud Shell to be ready.



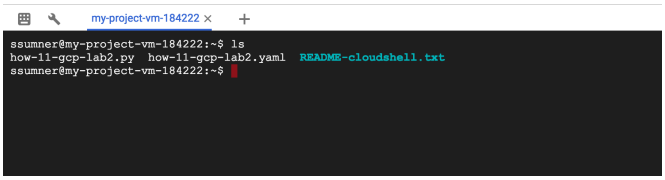
Step 4: Click the ellipses in upper right-hand corner of the Google Cloud Shell console and select **Upload File**. If you **Do Not See** the 3-dot ellipse please review Lab 1 >> Activity 1 >> Task 1 >> Step 2



Step 5: Upload the **how-11-gcp-lab2.yaml** and **how-11-gcp-lab2.py** file. Then close the Transfer window.

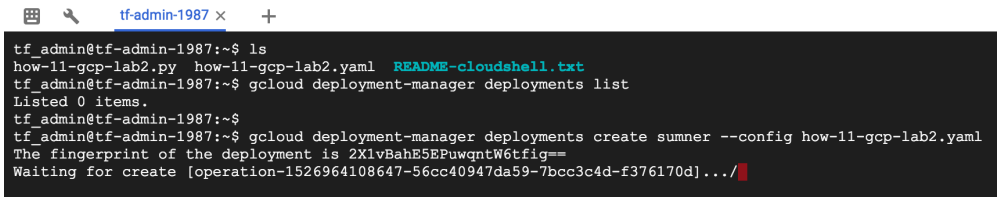


Step 6: In Cloud Shell prompt, you can use the **ls** command to confirm files have been uploaded.

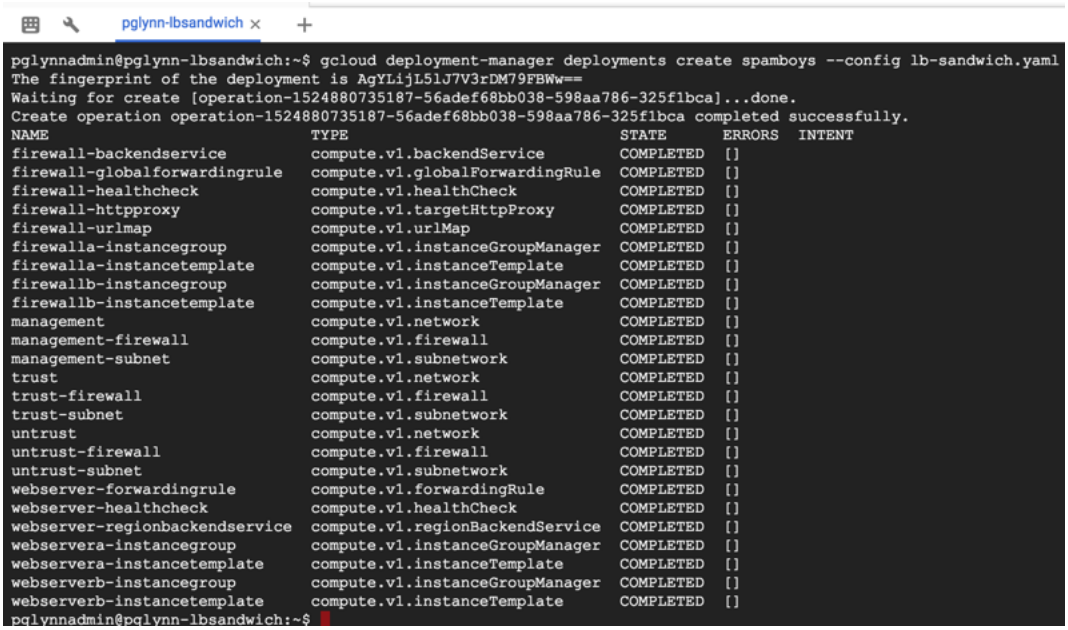


Step 7: Create a deployment using the command:
gcloud deployment-manager deployments create <your-last-name> --config how-11-gcp-lab2.yaml

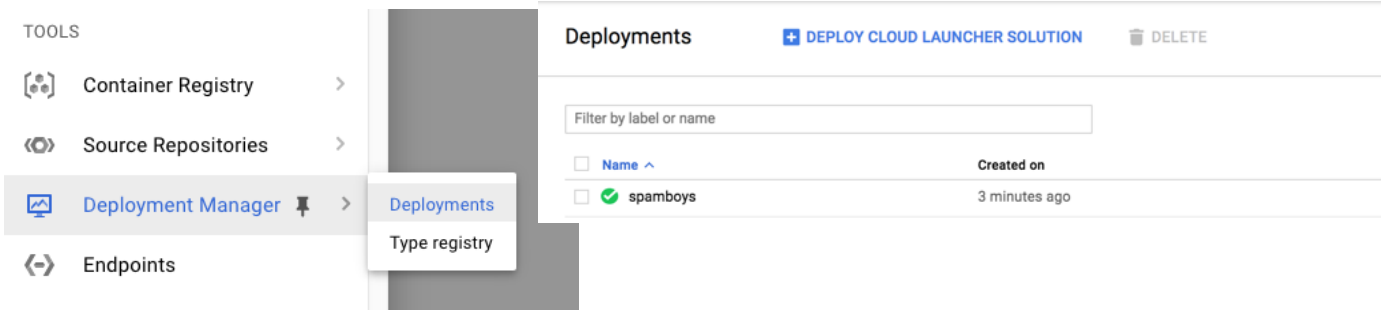
Although you can use any text to for the deployment name, using your last name provides for easy identification.



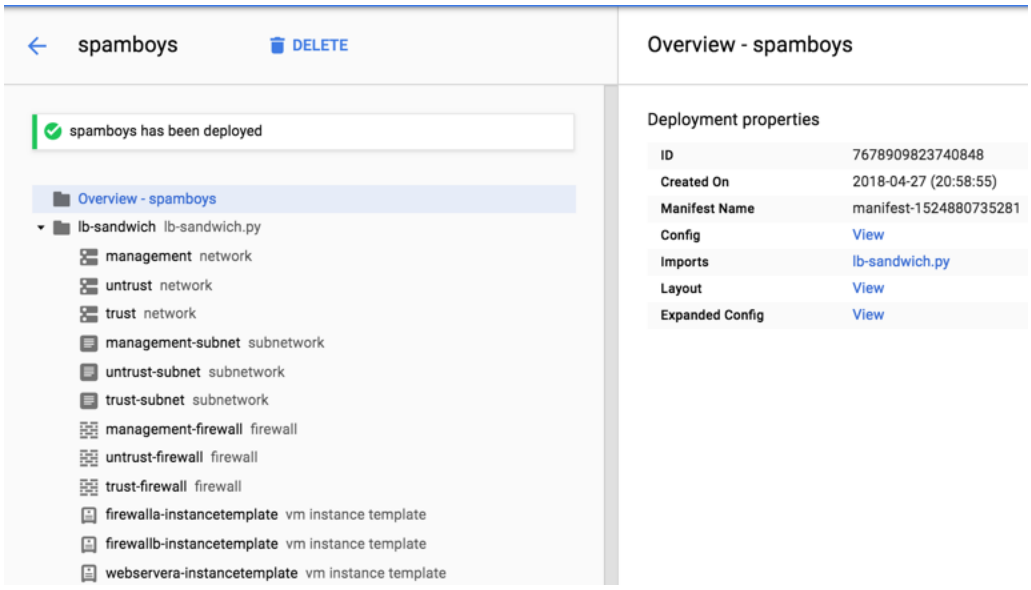
Step 8: When the deployment is completed, you will see the creation completed message for each operation.



You can also go to the **Deployment Manager > Deployments** service to view the deployment:



Step 9: Click the deployment that you have created and review the items that are created by the deployment manager.



Although the template will run to completion in a few minutes, it may take up to 10 minutes for the firewalls to complete the bootstrap process and respond to login requests.

The next sections will guide you through the completion of the configuration of the firewalls to support the load balancers.

End of Activity 6

Activity 7 – Review the Deployment

In this activity, you will:

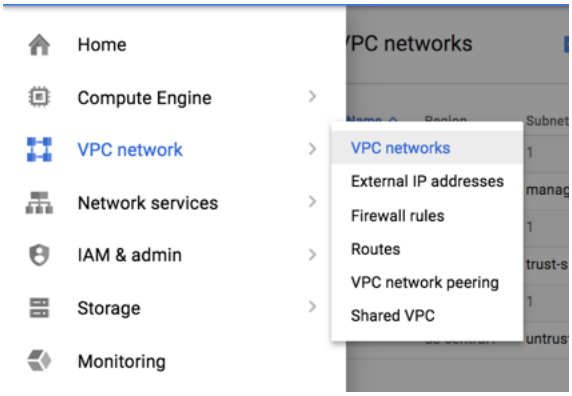
Review the GCP VPC Networks created via Deployment Manager Template

Review the GCE instances and the configuration created

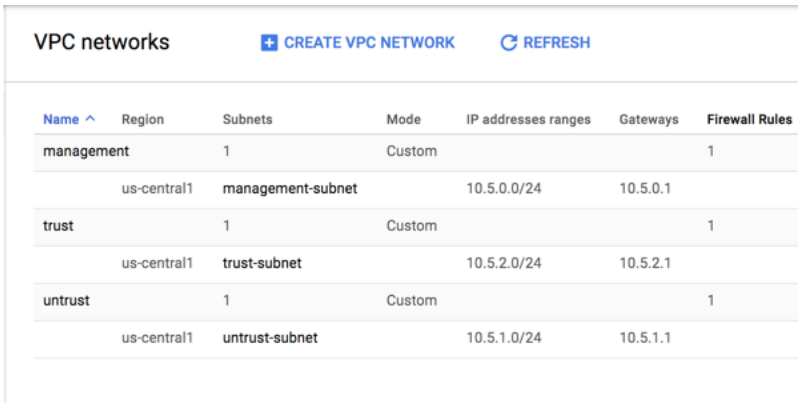
Task 1 - Review VPC Networks

Let's review all the components we have launched in the VPC based on the deployment manager file.

Step 1: The VPC networks can be accessed via ***VPC Networks > VPC Networks***. There you should see all networks created in your project:



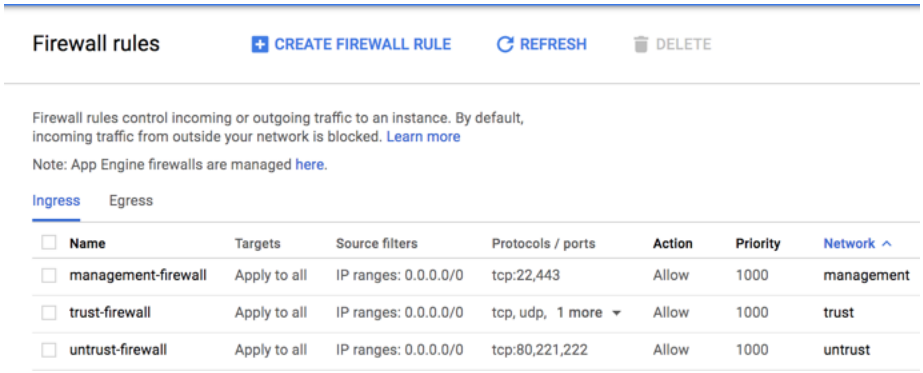
Step 1: A ***default*** network is created when the project is instantiated and can safely be ignored. The three networks created by the deployment are **management, untrust**, **trust**:



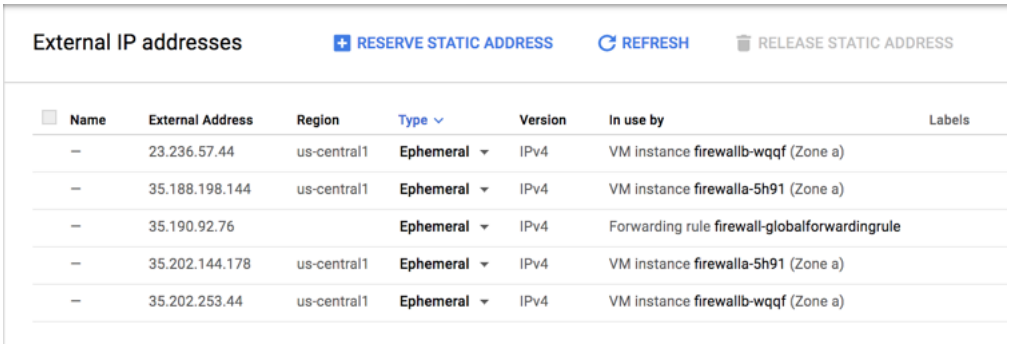
Name	Region	Subnets	Mode	IP addresses ranges	Gateways	Firewall Rules
management		1	Custom			1
	us-central1	management-subnet		10.5.0.0/24	10.5.0.1	
trust		1	Custom			1
	us-central1	trust-subnet		10.5.2.0/24	10.5.2.1	
untrust		1	Custom			1
	us-central1	untrust-subnet		10.5.1.0/24	10.5.1.1	

Note: Each network corresponds to an interface on the firewall. Although, you can create multiple subnets in a single network, a VM instance cannot have more than one interface attached to the same VPC network.

Step 2: Click on the ***Firewall rules*** to review the rules that are created by the deployment:



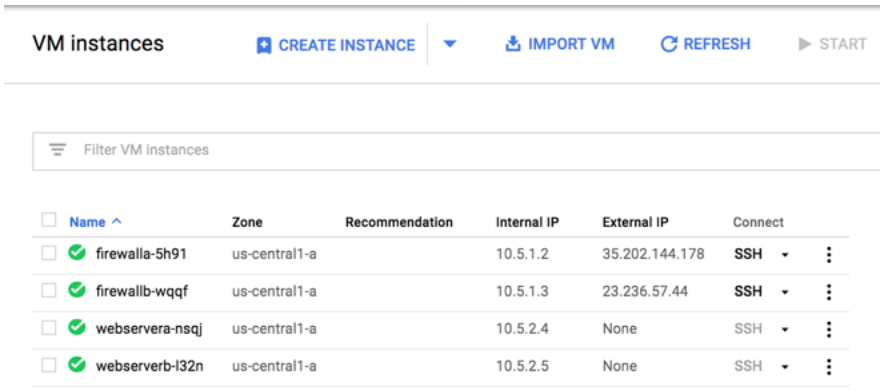
Step 3: Navigate to ***External IP addresses*** to view the public IP addresses that are deployed:



Task 2 - Review Compute Engine

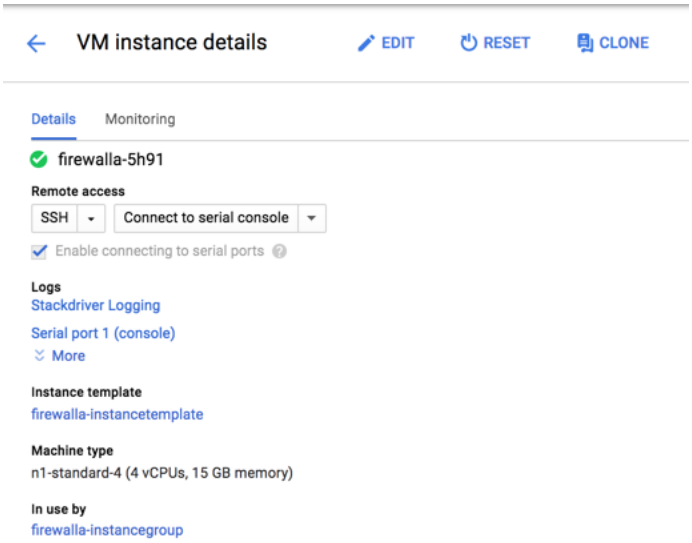
Review the VM instances created by the deployment in this task.

Step 1: Go to the ***Compute Engine > VM instances***. There are 4 instances in this deployment: 2 - Web servers, and 2 - VM-series firewalls (there may be additional instances depending upon what has been deployed previously):



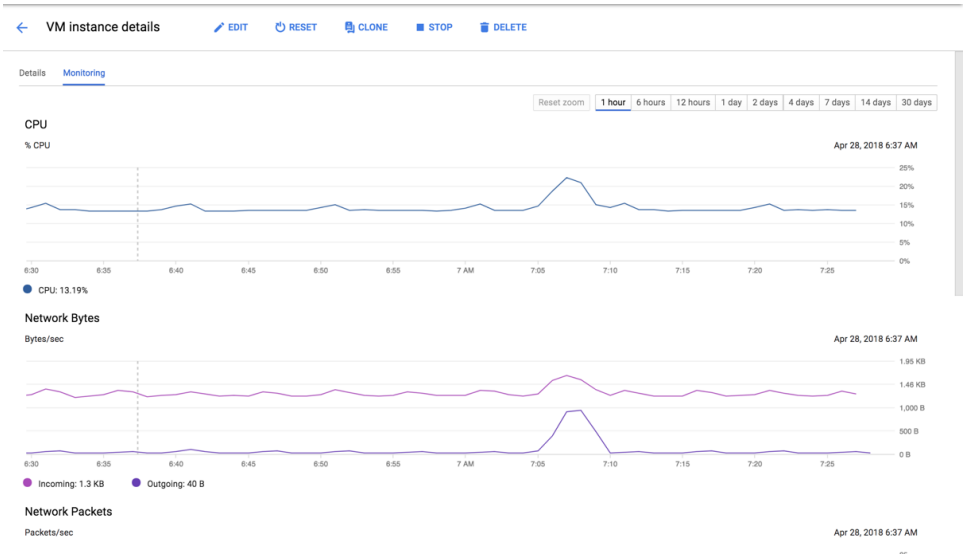
Note that firewalls will be named “firewalla” or “firewallb” followed by 4 random alphanumeric characters and webservers will be named “webservera” or “webserverb” followed by 4 random alphanumeric characters.

Step 2: Click one of the firewall instances to view the details:



From here, it is possible to connect to the virtualized serial console of the instance as well as see the mappings between public IP addresses and the firewall interfaces.

Step 3: Click the on **Monitoring** to review some of the basic metrics monitored by GCP:



Note: In addition to the standard metrics you see here, the VM-Series firewall can publish native PAN-OS metrics to GCP Stackdriver. The additional metrics allow you to assess performance and usage patterns that you can use to make the decision to launch additional VM-Series firewalls or terminate unneeded instances. We will not go into the details here but you can get more information from your instructor or your SE.

Step 4: Return to the instance details page and scroll down to ***Network Interfaces***. Notice that there are three network interfaces created for the VM-series firewall, management, eth1/1 for the public (untrust) network, and eth1/2 for the trust network.

End of Activity 7

Activity 8 - Review Firewall Configuration

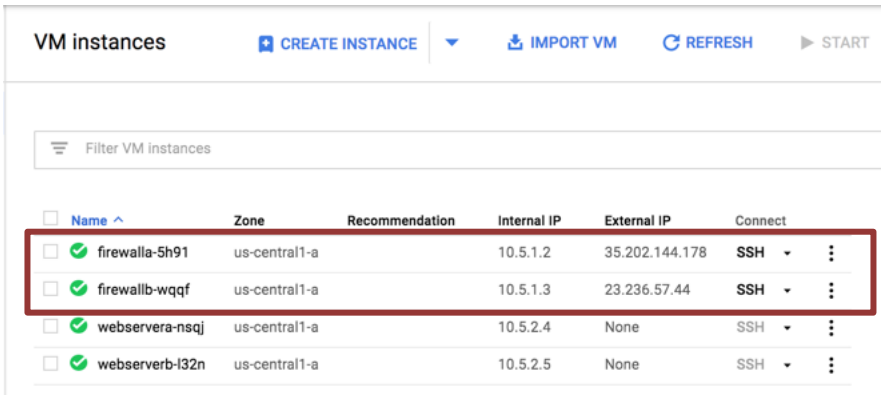
In this activity, you will:

- Login to the VM-Series firewall
- Review key portions of the firewall configurations

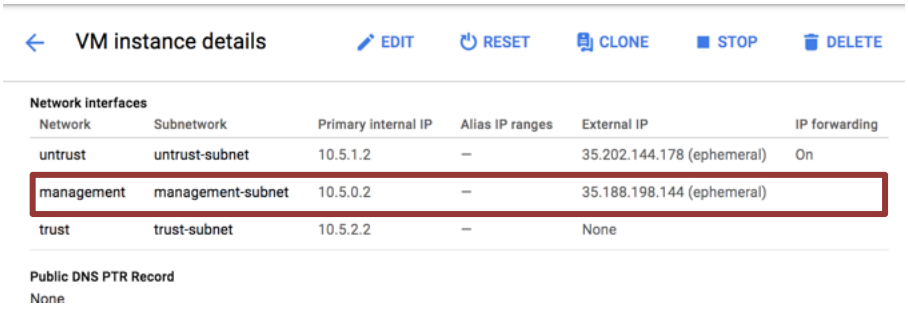
Task 1 - Access the firewall

NOTE: Bootstrapping a VM-Series firewall takes approximately 10 minutes. So once the template has been deployed successfully, it may be a while before the firewall is up and you are able to log into the firewall.

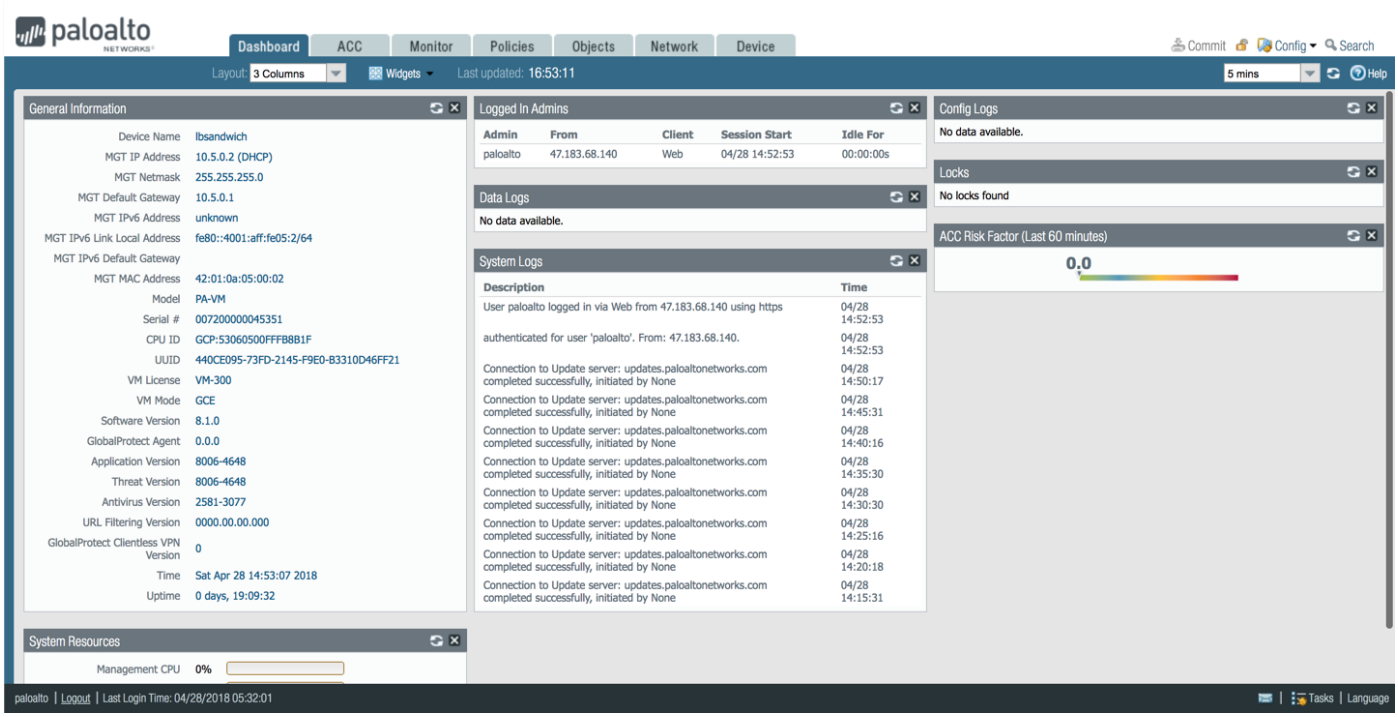
Step 1: Once the template deployment is complete, the firewall will show a green checkmark indicating it is running. Although you may see the checkmark, you may not be able to login immediately as the firewall may still be completing the bootstrap process:



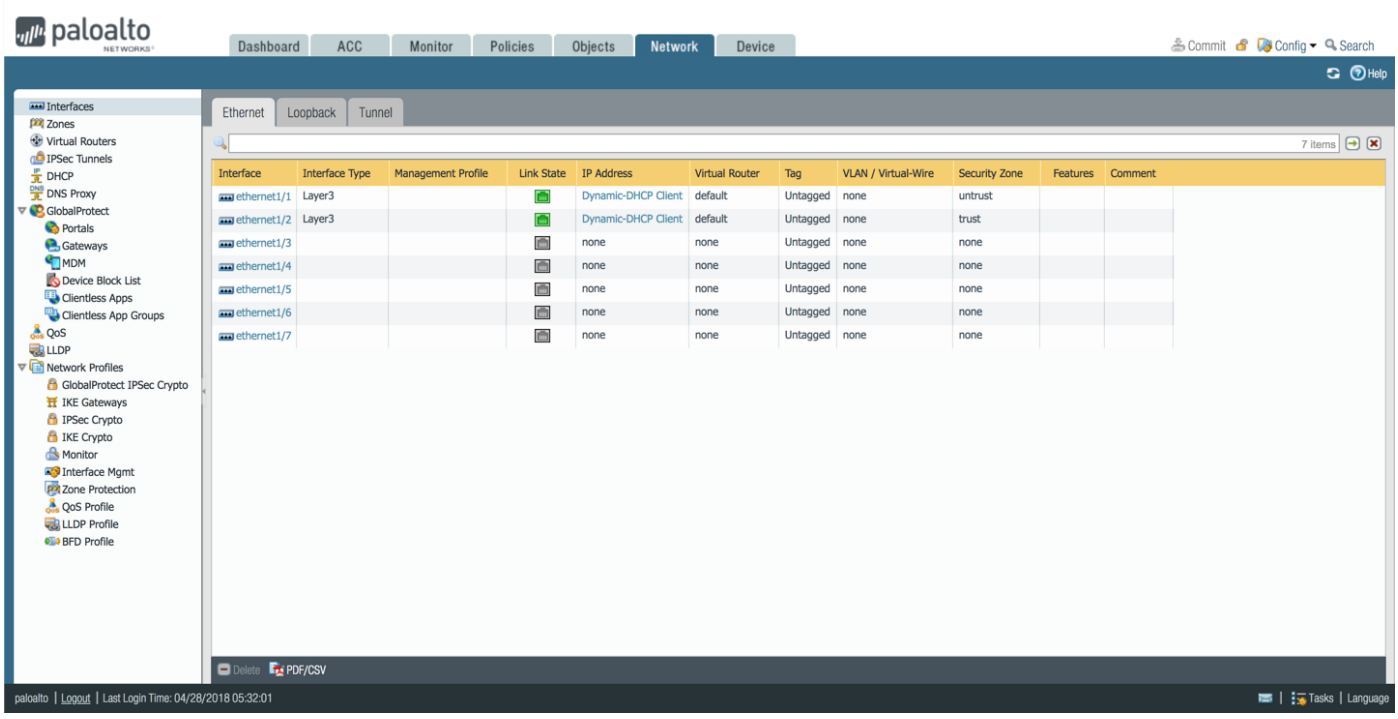
Step 2: The GCP load balancer will only send traffic to the primary ethernet interface of an instance. Consequently, we need to swap the management and ethernet interfaces (eth0 is management by default). Retrieve the management interface public IP address by clicking on the firewall name and scrolling down to **Network Interfaces:**



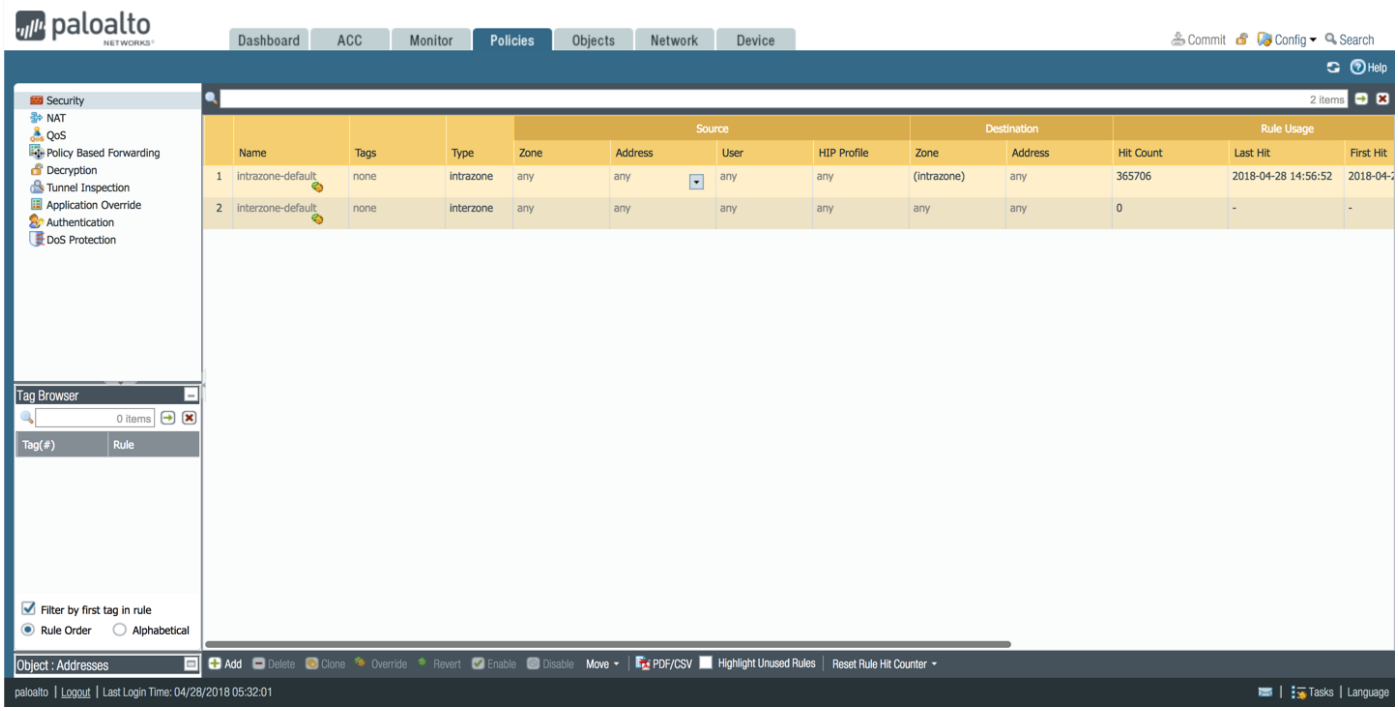
Step 3: Note the IP address and repeat the steps for the second firewall. You should be able to login to the firewall using the **username: paloalto** and password: **Pal0Alt0@123**



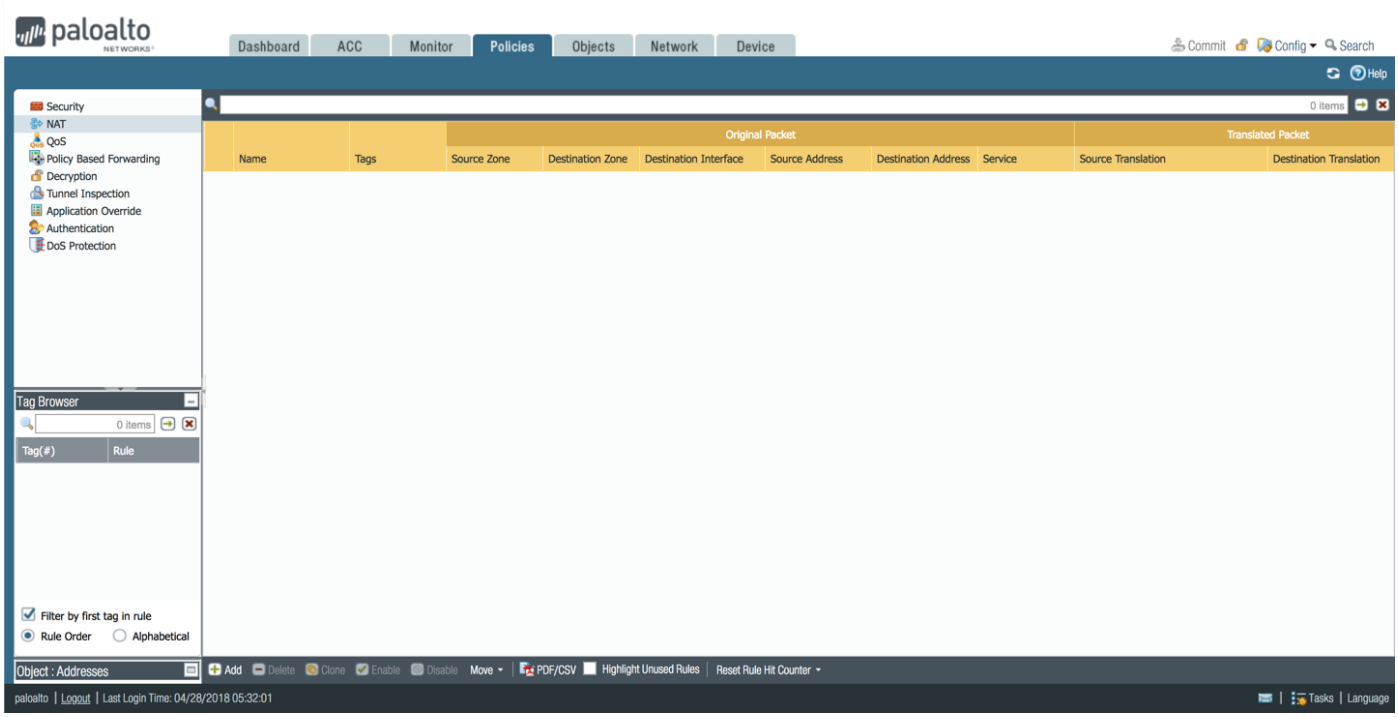
Step 4: The interface should be set up by the template deployment but you should review them and note the IP addresses on the interfaces:



Step 5: The default security policy has been modified to enable additional logging but is otherwise unaltered:

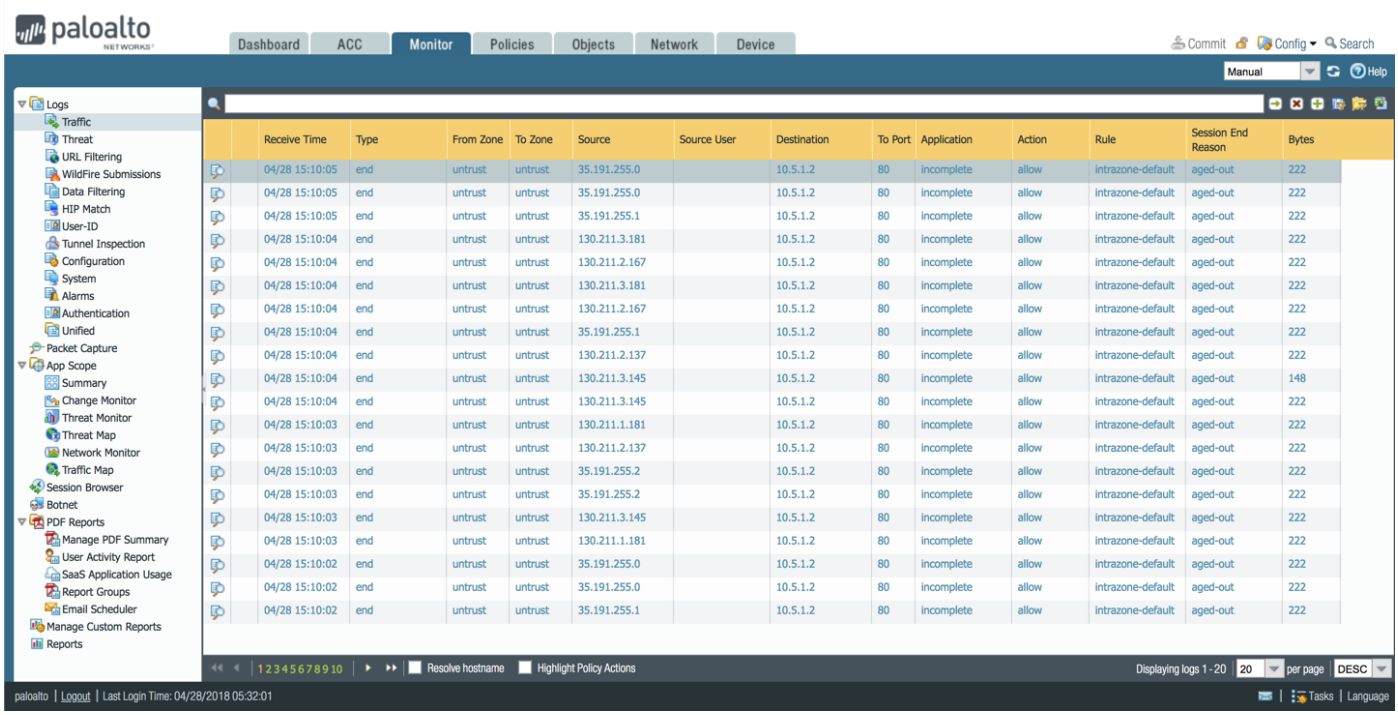


Step 6: No NAT policy has been defined:

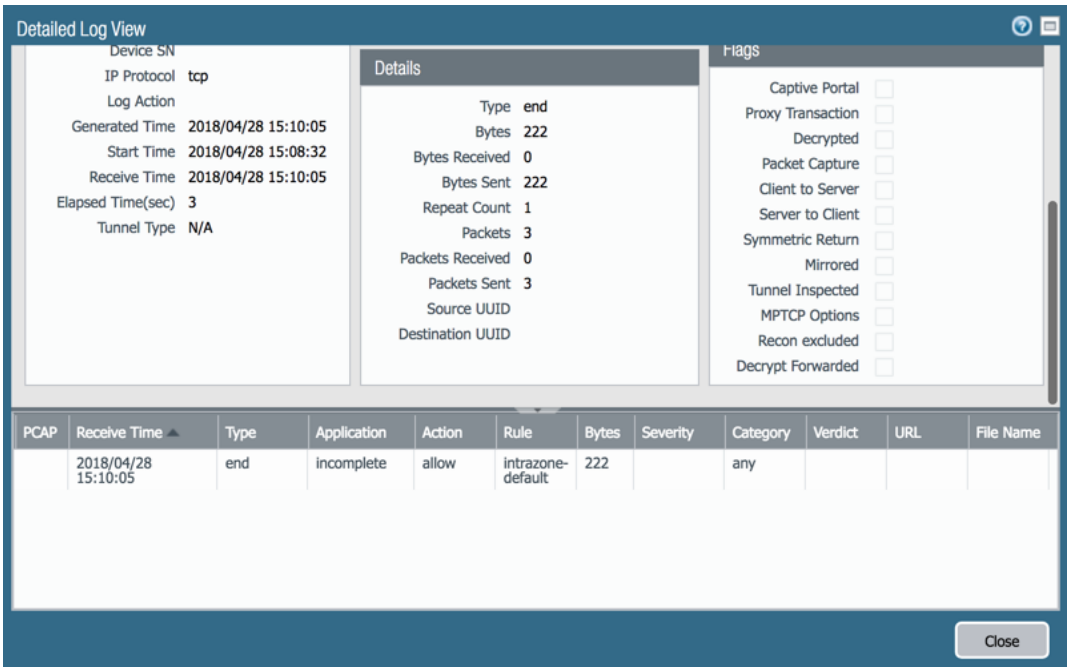


Task 2 – Review Monitor Tab

Step 1: Navigate to the **Monitor** tab. You should see traffic in the logs on port 80. This is the health check traffic that is coming from the external load balancer. If you look at the application column, you should see “incomplete”. This indicates that the firewall is unable to identify the application.



Step 2: Click on the icon () to show more detail about one of the traffic log entries. A new window will pop up:



Note that **Bytes Received** is zero. This indicates that the firewall is only seeing one side of the session (the request from the load balancer). Because the intrazone policy is set to permit by default, the firewall is accepting the health checks from the load balancer but because there is no NAT rule in place to translate the traffic, the firewall does not know what to do and it gets dropped. We will fix that in a subsequent step.

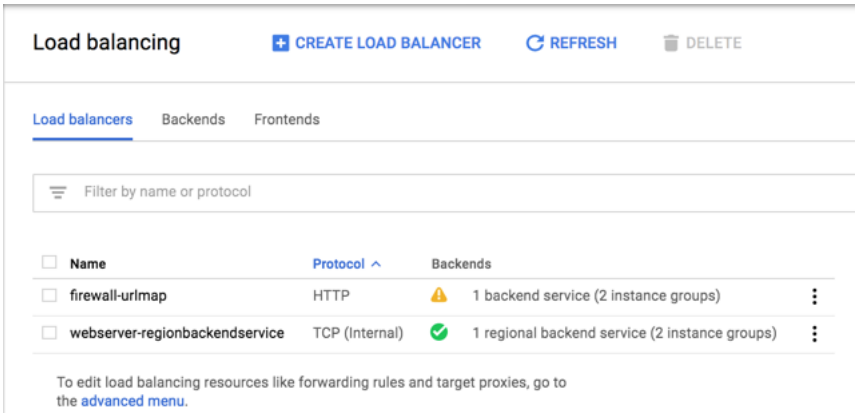
End of Activity 8

Activity 9 – Complete and Verify Firewall with Load Balancer Configuration

Task 1 – Complete the firewall configuration

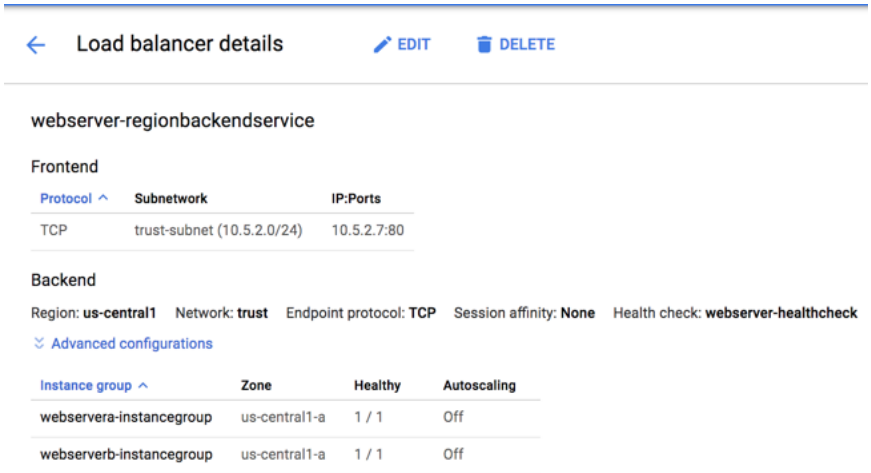
In this task, you will complete the firewall configuration and then verify that you can reach the web server through the firewall.

Step 1: Navigate to **Network Services > Load Balancing**:



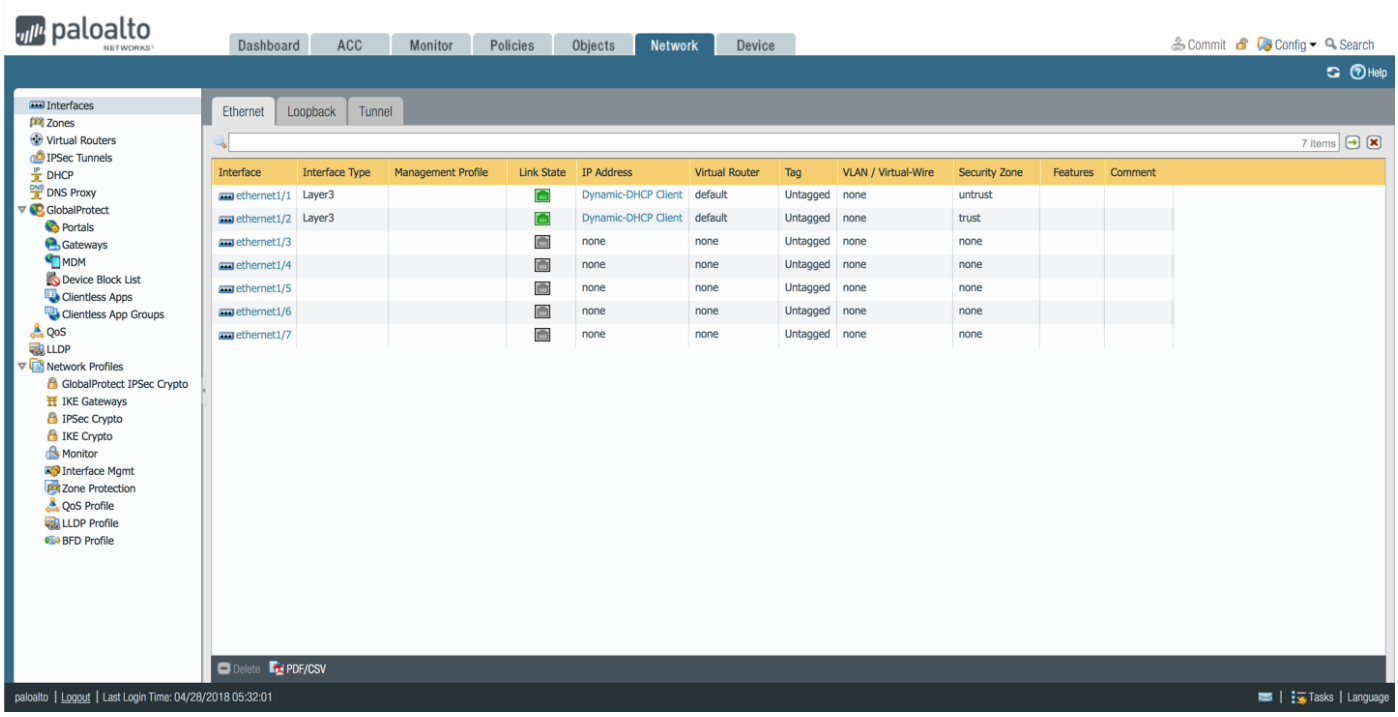
The **firewall-urlmap** load balancer is the internet-facing load balancer that is in front of the firewalls. It is showing unhealthy due to the lack of policies to process the traffic. The **webserver-regionbackendservice** is the internal load balancer that distributes traffic to the webservers on the backend.

Step 2: Click on **webserver-regionbackendservice:**

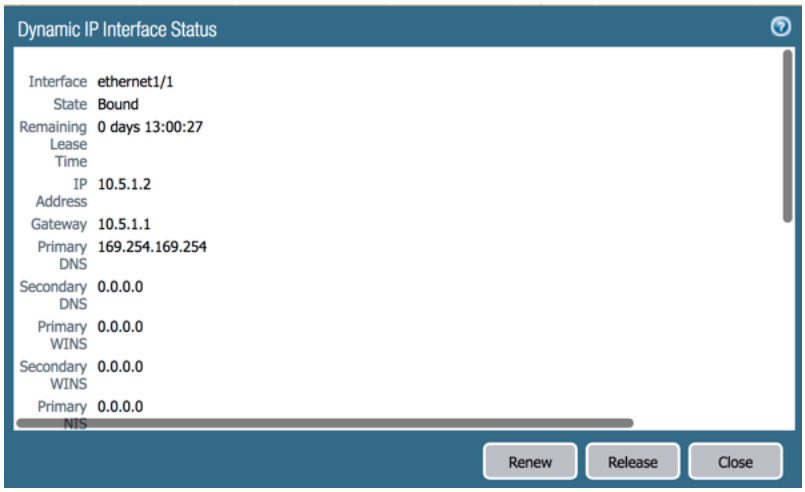


Note the IP address associated with the frontend (10.5.2.7). This is the IP address that the firewall will need to send traffic to. Make a note of this address.

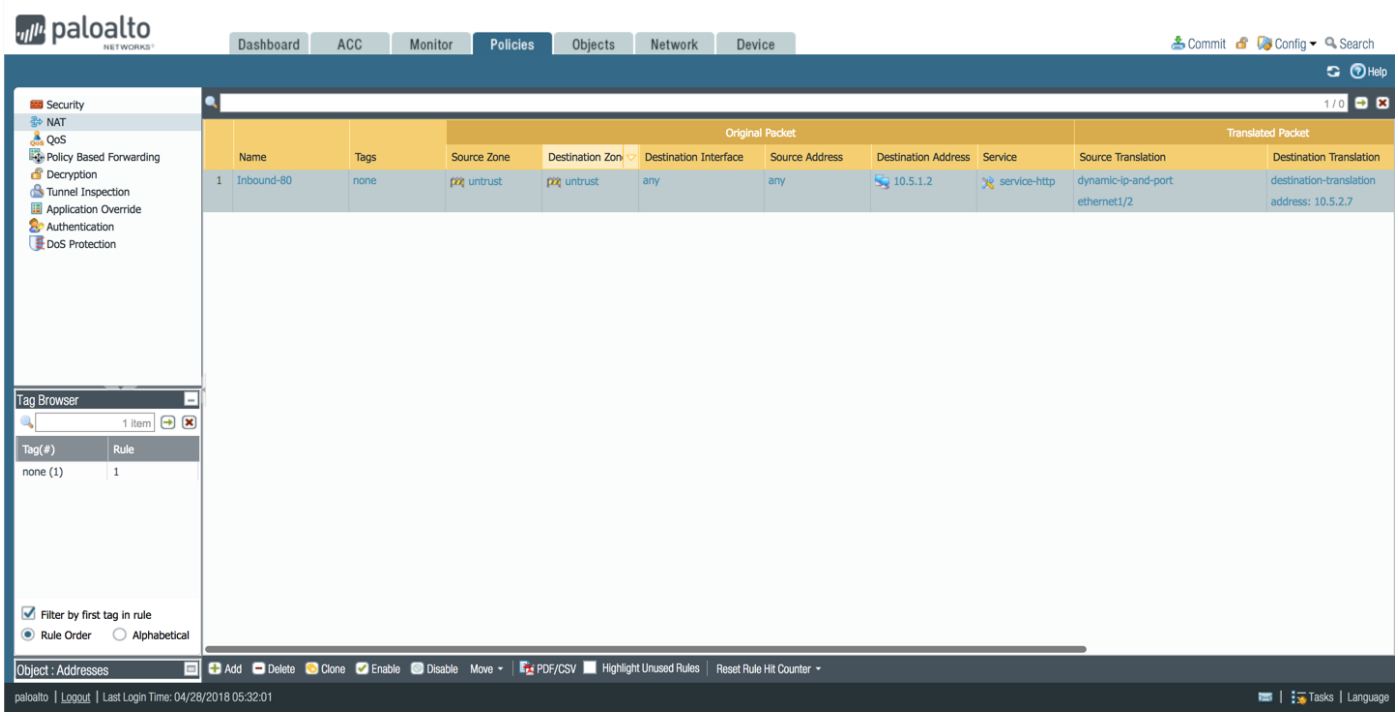
Step 3: Go back to the firewall and navigate to the **Network** tab:



Step 4: Click on ethernet1/1 (the external interface) and make a note of the IP address:



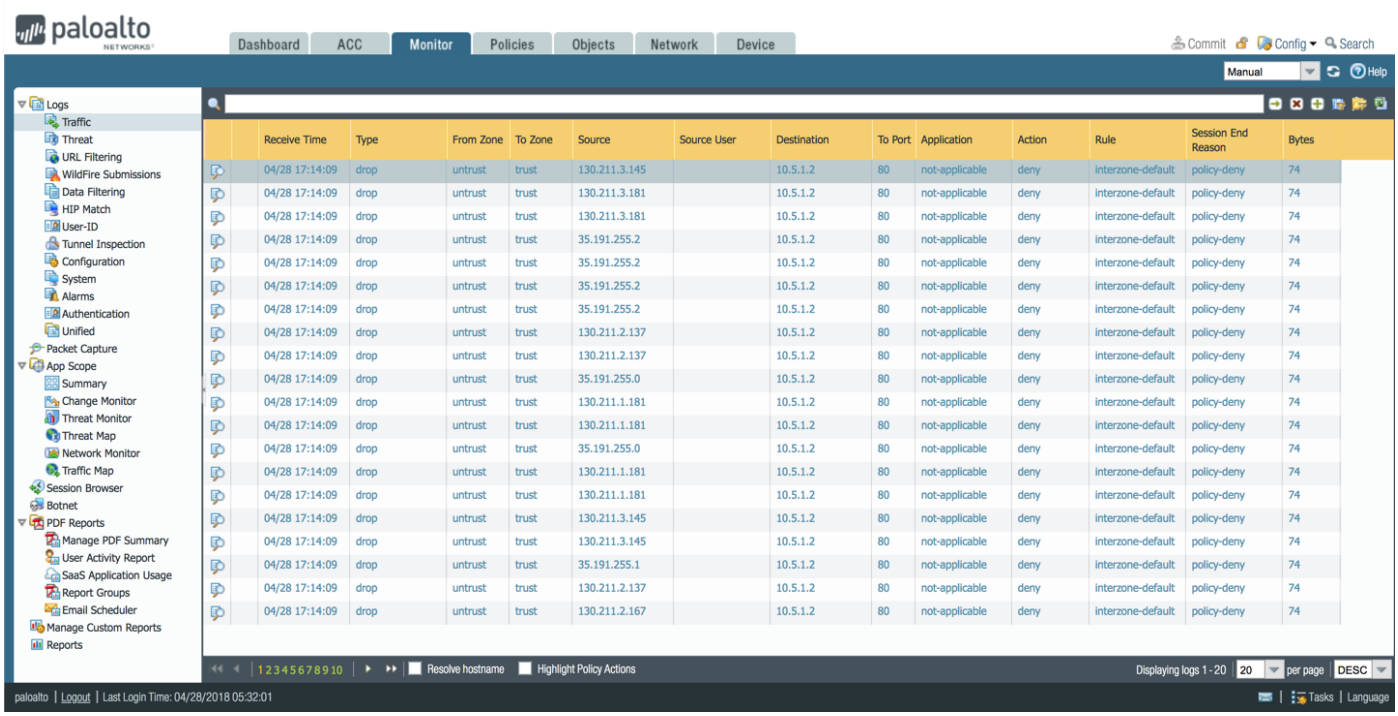
Step 5: Navigate to the **Policies** tab and add a rule translating inbound connections on port 80 arriving at the external IP of the firewall to the frontend IP address of the internal load balancer:



In addition to the destination translation, we also add a source NAT that hides the traffic behind the IP address of the internal interface of the firewall. What purpose does it serve?

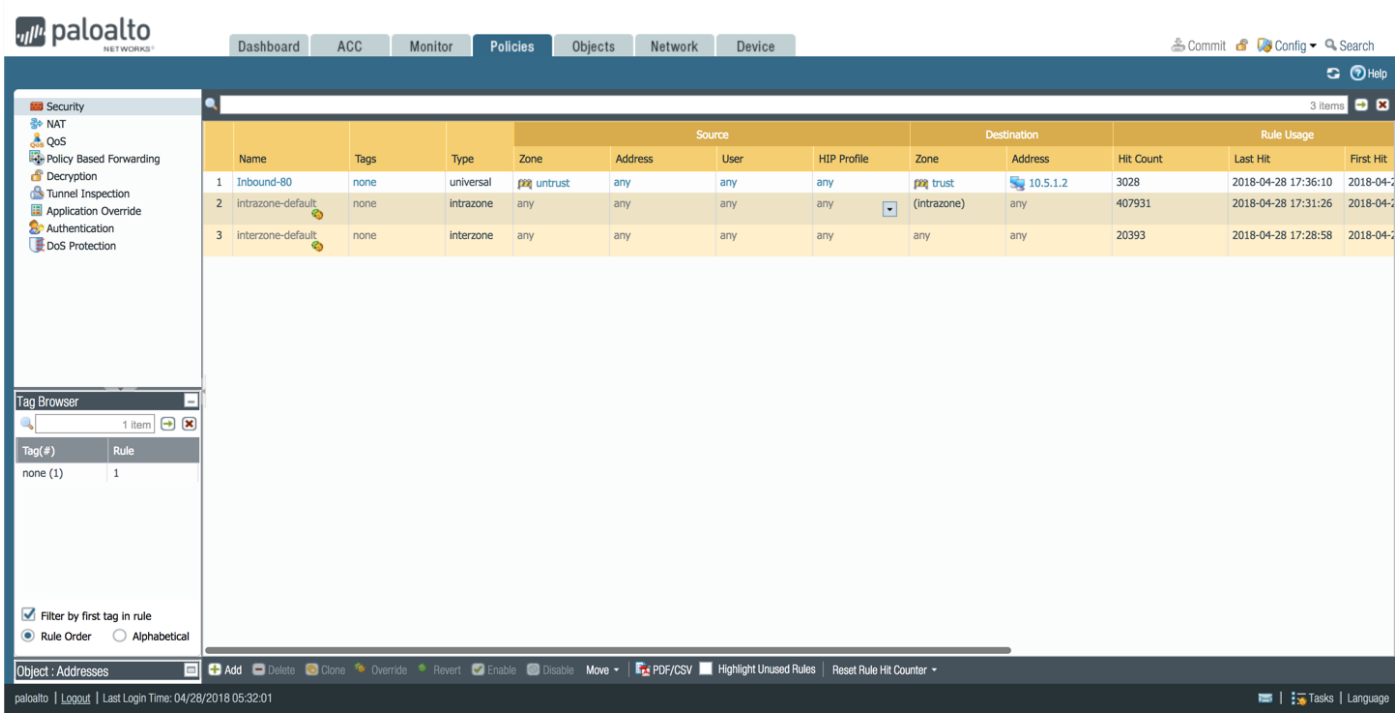
Step 6: Replicate the changes to the second firewall (beware the IP address on the external interface) and commit the changes on both.

Step 7: Navigate to the **Monitor** tab and note the change in the logs. Instead of source and destination zone of **untrust** and application of **incomplete**, we now see that the source zone is **untrust** and the destination zone is **trust** and the application is **not-applicable**.

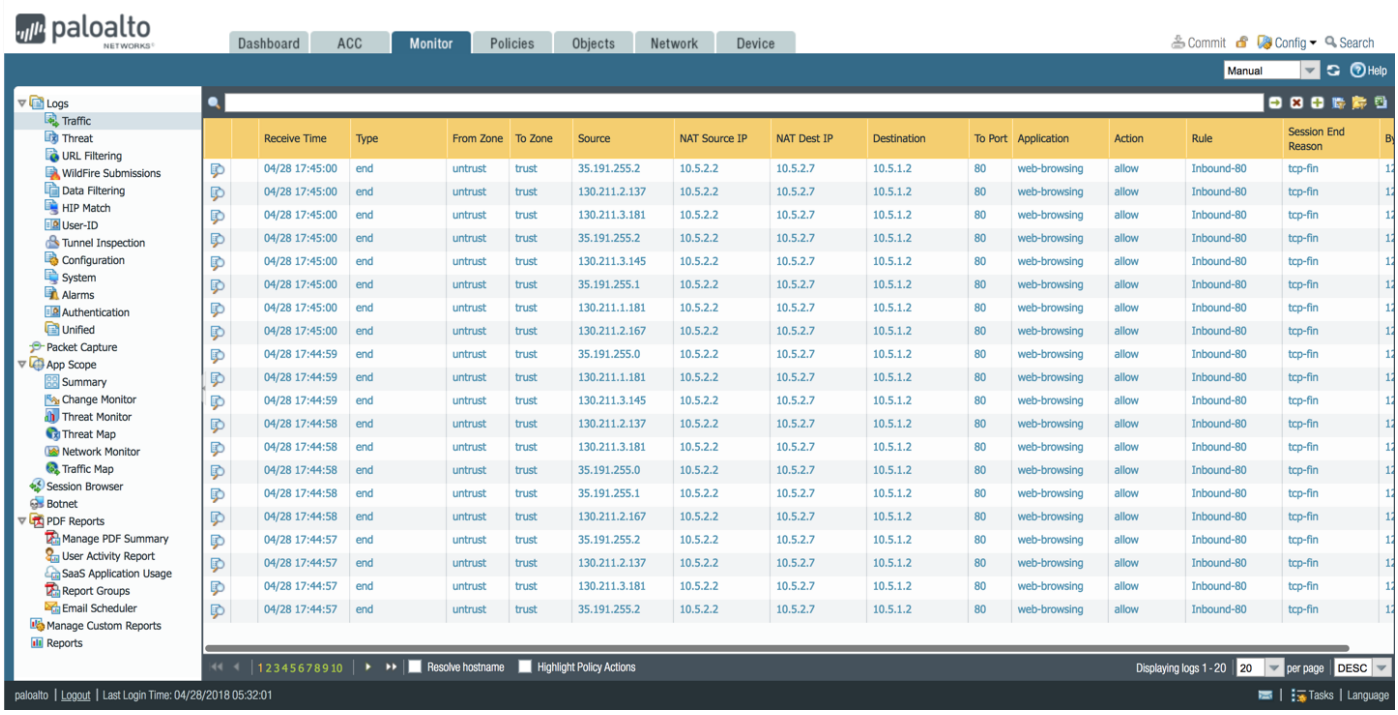


Progress but we are not there yet. What's up with that? Hint: the firewall policy permits **intrazone** traffic but denies **interzone** traffic.

Step 8: Jump back to the policy tab and add a **Security** rule that permits traffic from the **untrust** zone to the **trust** zone with a destination of the firewall external IP address, an application of **web-browsing** and a service of **application-default**:

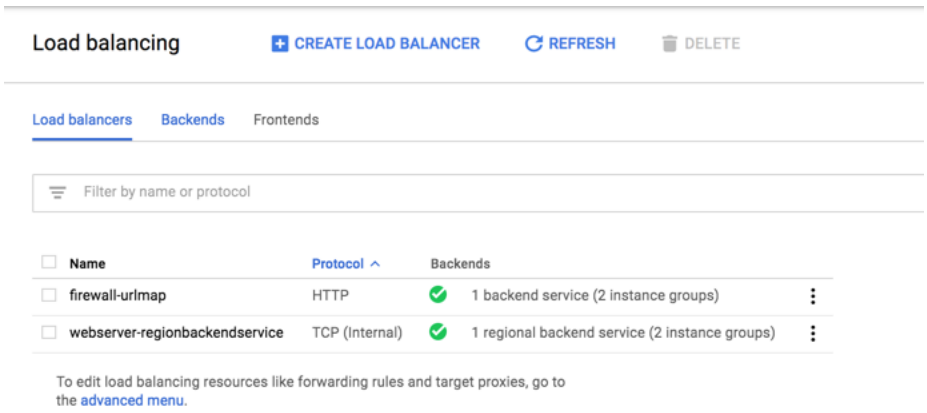


Step 9: Navigate back to the **Monitor** tab et voilà:



You can add the **NAT Source IP** and **NAT Dest IP** columns to see how the firewall is translating the addresses.

Step 10: Go back to the GCP console, **Network services > Load balancing** (Click **REFRESH** at the top if you are already there):

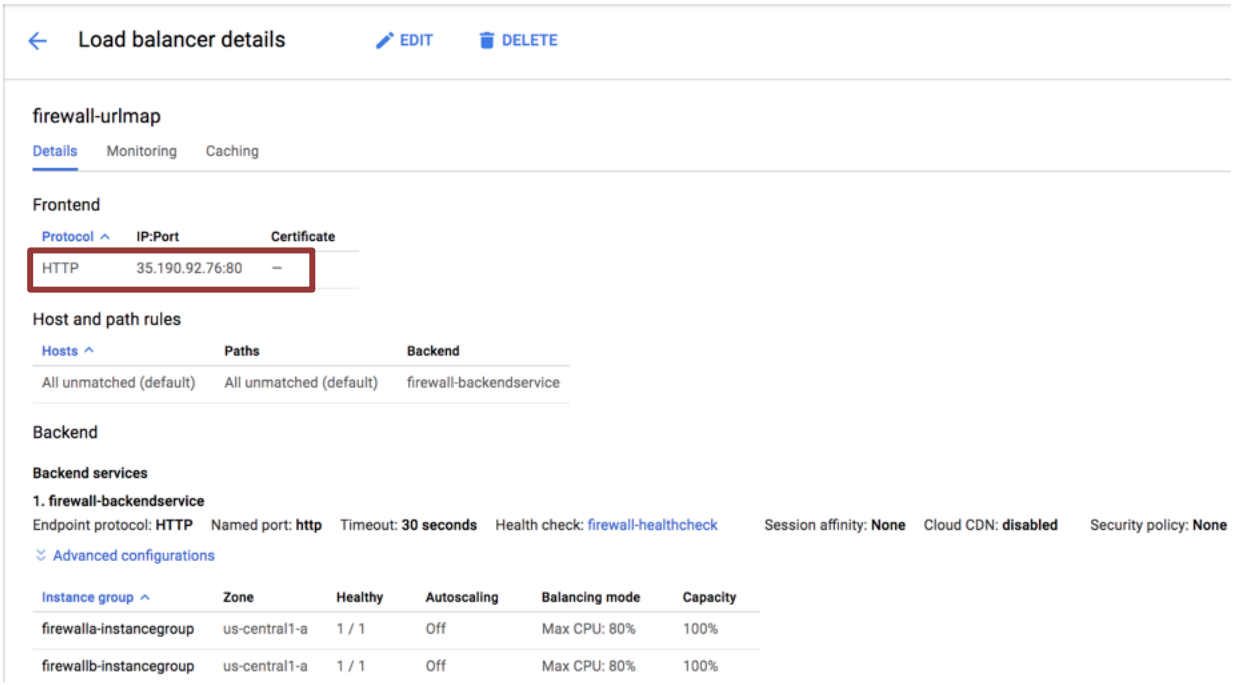


Now we can see that the backend service associated with the internet-facing load balancer is reported as healthy.

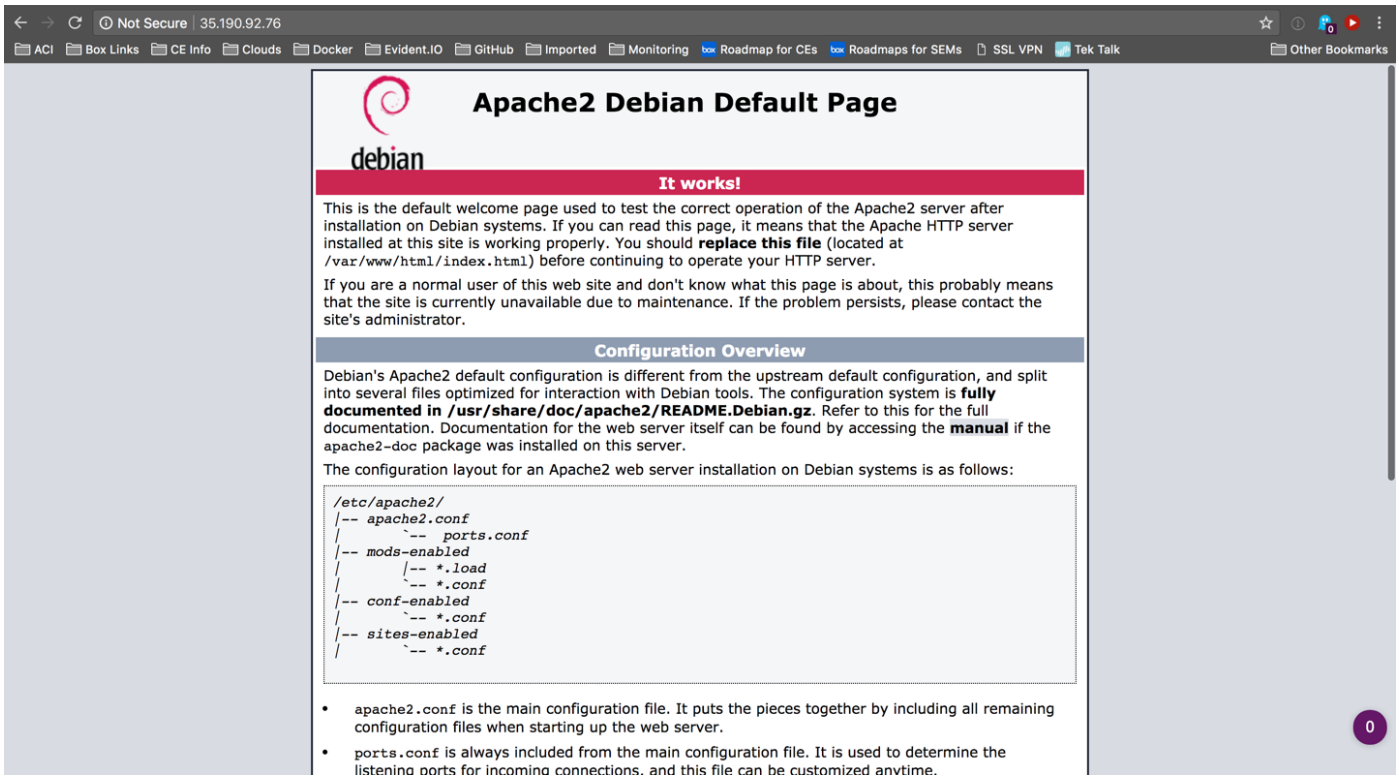
Task 2 – Verify the Configuration

In this task, you will verify that you can reach the web server through the firewall.

Step 1: From the **Network services > Load balancing** section of the GCP console, click on the internet load balancer (**firewall-urlmap**) and note the public IP address associated with the load balancer:



Step 2: Open a web browser, paste the address in and press enter:

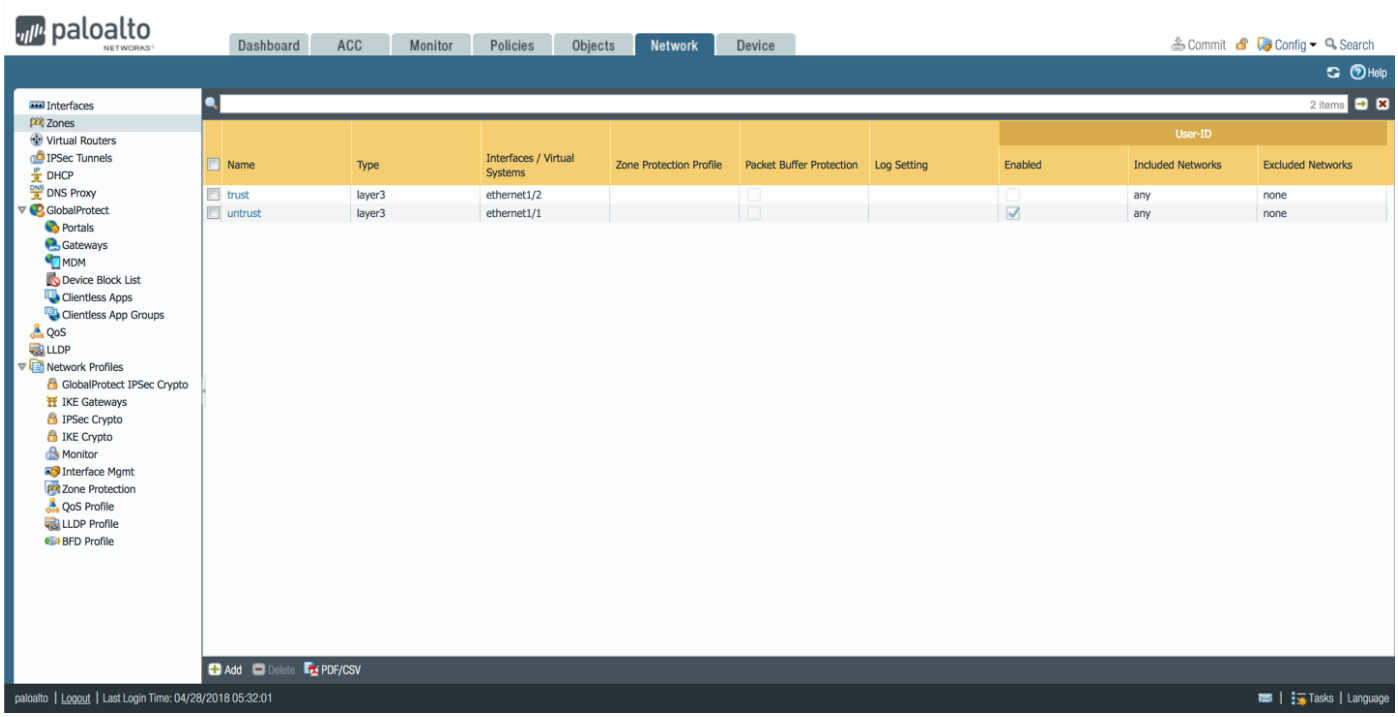


Success! You have validated that the traffic is passing through the firewall and reaching the webservers on the backend.

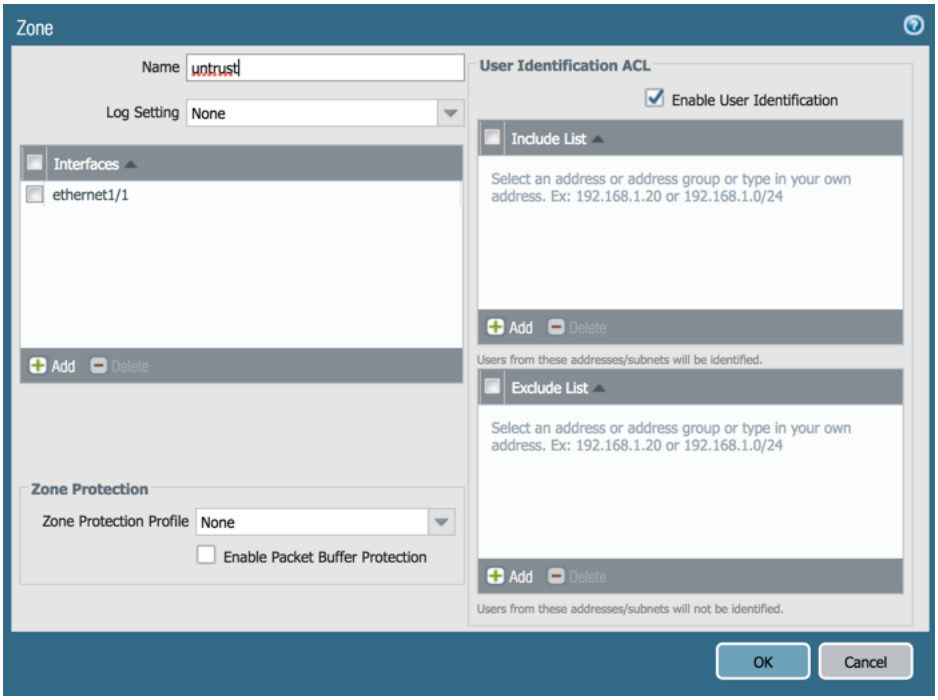
Task 3 - Log X-Forwarded-For (Bonus)

One side effect of using an application load balancer (ALB) is that the original source address is hidden from the firewall. Fortunately, most ALBs insert the original source address as an X-Forwarded-For (XFF) header that the VM-Series firewall can log. In this exercise, you will enable user-ID on the relevant interface, create a URL filtering profile to log XFF, and apply it to the firewall rule.

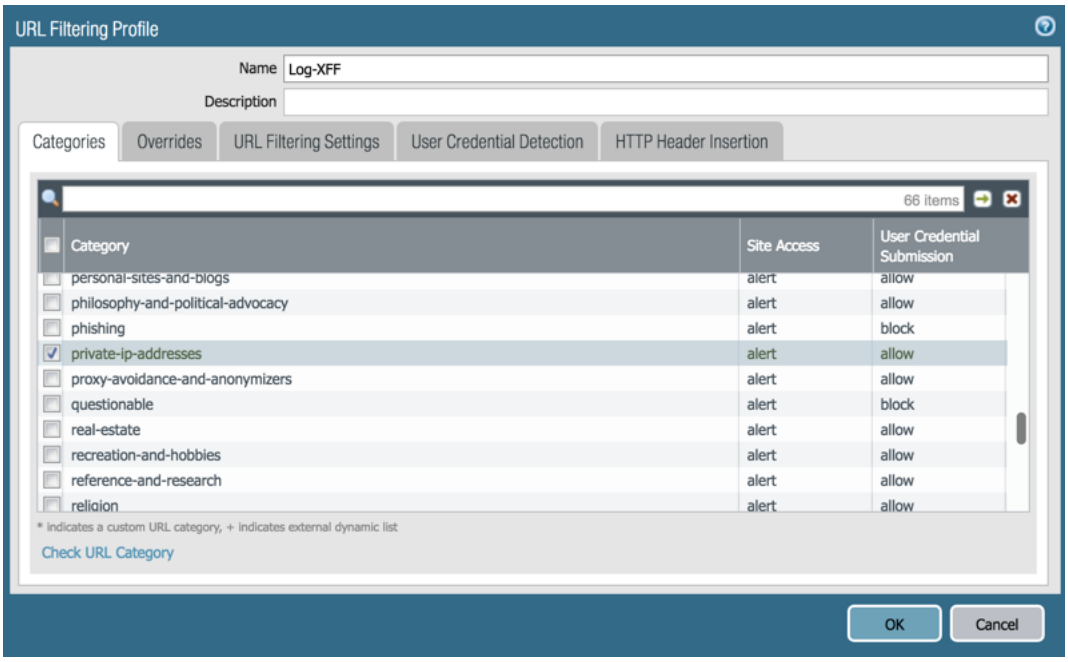
Step 1: On the firewall, navigate to the **Network** tab and then **Zones**:



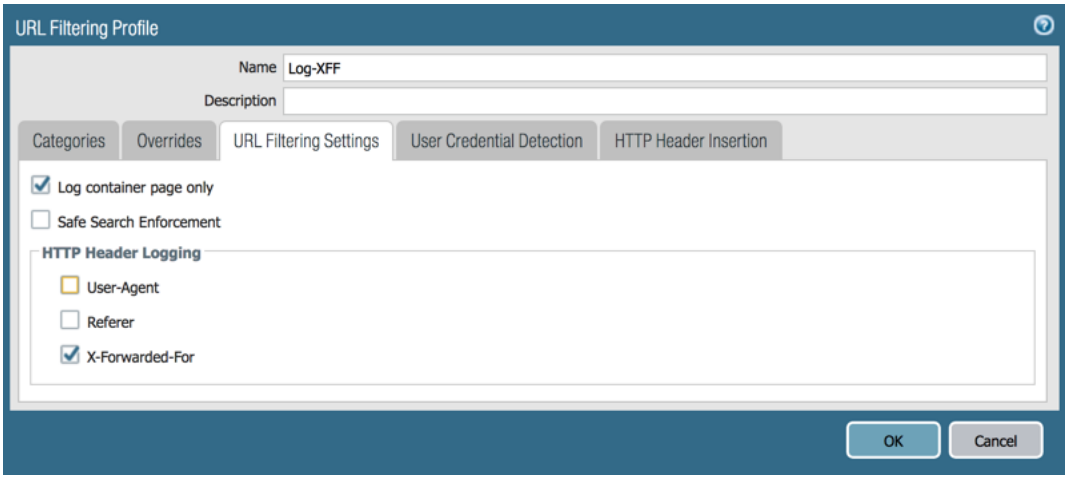
Step 2: Click on the **untrust** zone and tick the box **Enable User Identification**:



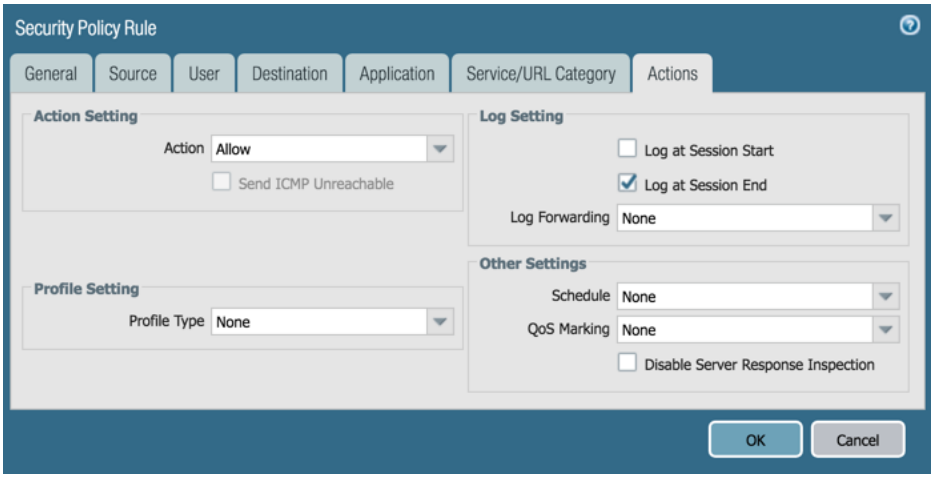
Step 3: Navigate to the **Objects tab > Security Profiles > URL Filtering** and create a new URL filtering profile setting **private-ip-addresses** to alert:



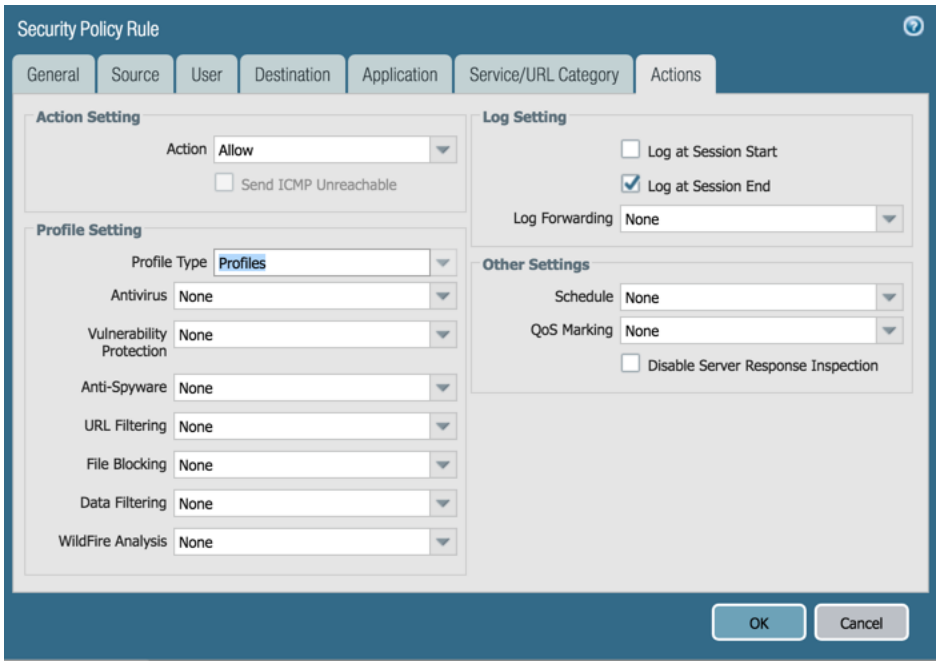
Step 4: Click on URL Filtering Settings and tick the **X-Forwarded-For** box under **HTTP Header Logging**:



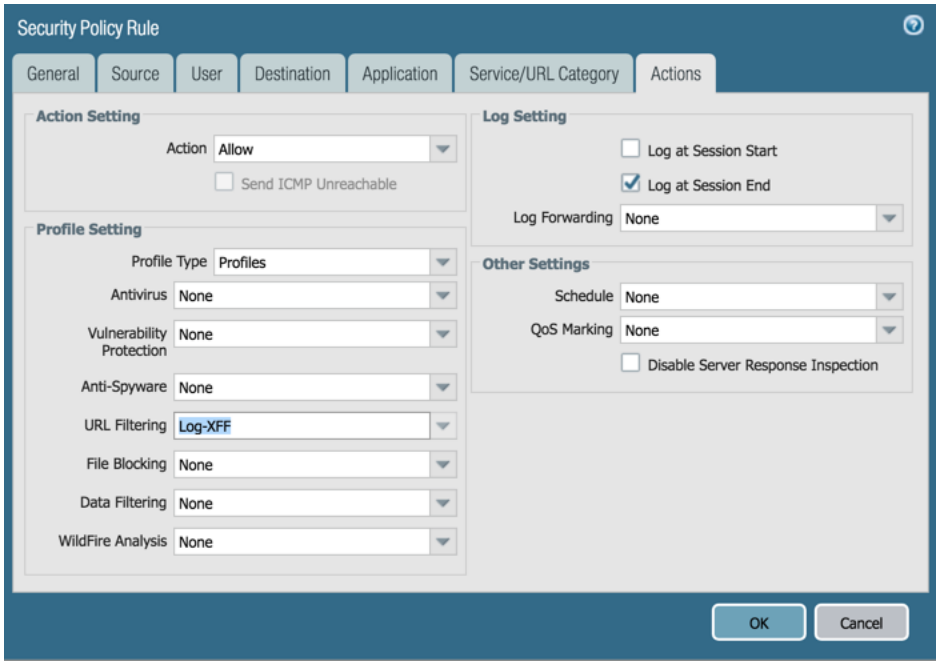
Step 5: Navigate to the **Policies** tab and edit the policy permitting the load balancer traffic. Click on the **Actions** tab:



Step 6: Click on the **Profile Type** dropdown and set it to **Profiles**:

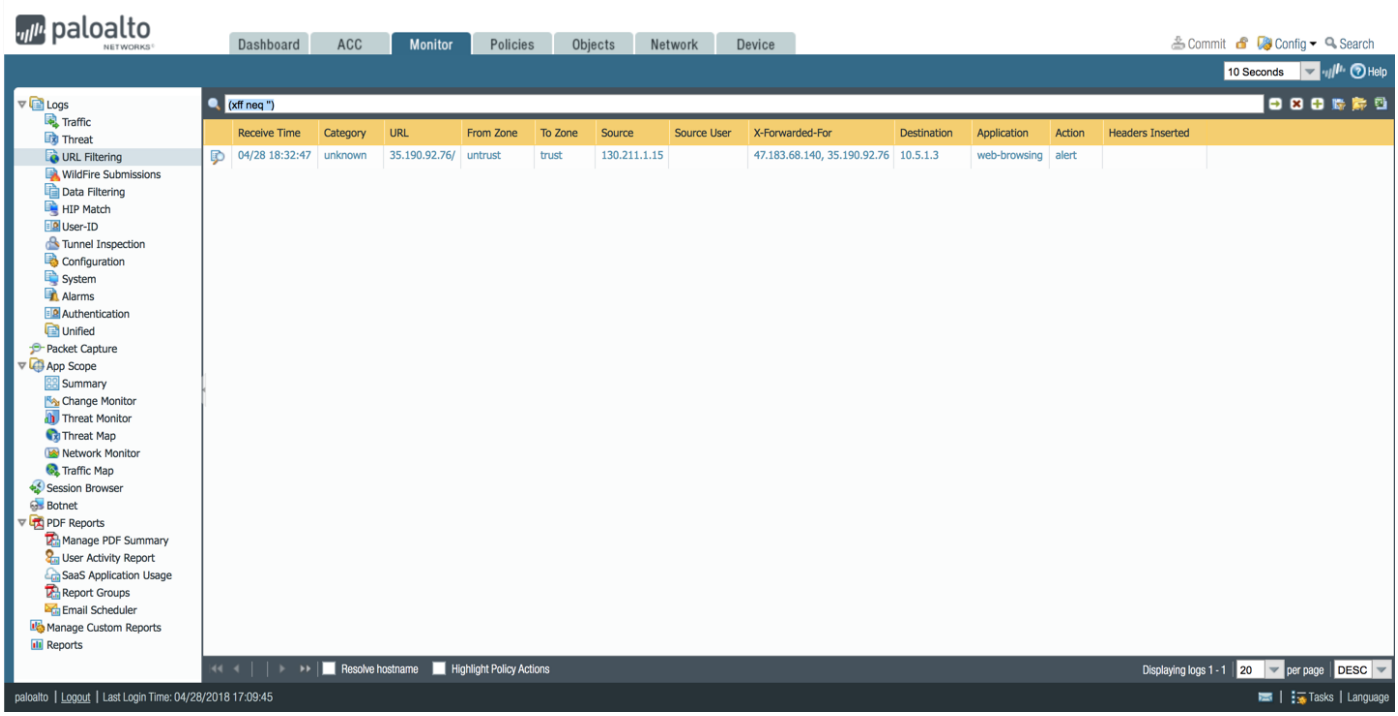


Step 7: Set the **URL Filtering** profile to the one created earlier:



Step 8: Commit the changes and replicate to the other firewall.

Step 9: Refresh the connection to the public load balancer a few times and then navigate to the **Monitor** tab **URL Filtering**. Add the **X-Forwarded-For** column to the firewall log and apply the filter (xff neq ‘’) (xff neq '') to the logs to filter out the load balancer health checks (it may take a few moments for the logs to show up). Note: in the filter the ‘’ are apostrophe’s not quotes.



If all goes well, you should see the request traffic logged with the original source IP and the IP address of the external load balancer.

End of Activity 9

Conclusion

Congratulations!! You have now successfully integrated the PANW VM series firewall into GCP to gain visibility into East/West and North/South traffic entering and leaving your Google Cloud.

You have also configured and leveraged the VM Series to log X Forward-For information.

No Cleanup required the lab will self-destruct when the Lab timer reaches 00:00.

Thank you for taking the Ignite18-HOW-11-GCP-Lab.