Configuring a Palo Alto Firewall in AWS

Version 1.0 10/19/2015

GRANT CARMICHAEL, MBA, CISSP, RHCA, ITIL

For contact information visit EVANCED.NET
Table of Contents

The Network Design................................................................................................................................................. 2

Step 1 – Building the AWS network .......................................................................................................................... 3
  Building the Internet Gateway..................................................................................................................................... 4
  Creating the Route Tables for the Public and Private VPC subnets....................................................................... 5

Step 2 – Building the Palo Alto Network (PAN) ........................................................................................................ 8
  Creating Security Groups ........................................................................................................................................ 8
  Add the Palo Alto Firewall Instance ......................................................................................................................... 10
  Configure the Palo Alto Firewall .............................................................................................................................. 16
  Adding the Security Policies .................................................................................................................................. 23

Step 3 – Creating the EC2 Linux servers behind the PAN .......................................................................................... 29
  Building the Bastion Host...................................................................................................................................... 31
  Creating the Web Server Instance ........................................................................................................................... 35
This tutorial assumes you already have an AWS account, security keys and know how to use a SSH client to access the AWS servers. For this exercise PuTTY, Pageant, and PuTTYgen will be used to access the AWS servers.

**NOTE:** Charges may apply when using AWS services. Before proceeding, be sure to read and understand Amazon’s user agreement and the respective charges. Secondly, this tutorial is intended to be a quick reference for setting up the Palo Alto in AWS, and in no way recommends, implies or suggests best practice for securing the environment.

### The Network Design

In this tutorial you will create a web server farm behind a Palo Alto firewall in AWS. Web servers will be built in a private DMZ network. An Internet Gateway will be created for Internet access, and Elastic IPs will be used to associate (or NAT) to the public network.

<table>
<thead>
<tr>
<th>PRIVATE IP LIST</th>
<th>Public IP/Private IP</th>
<th>VPC Network</th>
</tr>
</thead>
<tbody>
<tr>
<td>10.0.1.10 : PAN Management NIC</td>
<td>X.X.X.X/10.0.1.10 PAN MGT</td>
<td>10.0.0.0/16</td>
</tr>
<tr>
<td>10.0.1.20 : PAN Public NIC</td>
<td>X.X.X.X/10.0.1.20 PAN Public</td>
<td></td>
</tr>
<tr>
<td>10.0.1.30 : Bastion Public NIC</td>
<td>X.X.X.X/10.0.1.30 Bastion Public</td>
<td></td>
</tr>
<tr>
<td>10.0.2.20 : PAN Private NIC</td>
<td></td>
<td></td>
</tr>
<tr>
<td>10.0.2.30 : Bastion Private NIC</td>
<td></td>
<td></td>
</tr>
<tr>
<td>10.0.2.40 : Web Server NIC</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Step 1 – Building the AWS network
Let’s get started by creating our Virtual Private Cloud (VPC) network, which is our virtual network where virtual resources will be launched.

Browse to VPC > Your VPCs and select Create VPC.

Next, create subnets within your VDC – one for the public side of your VDC and the other for the private.

Browse to VPC > Subnets and select Create Subnet

Create the VPC – Public Subnet

Create the VPC – Private Subnet
Here is the final result

**Building the Internet Gateway**

Now we need to create the Internet Gateway, which allows instances, i.e. servers, to communicate with the Internet. The Internet Gateway is the Internet router for your VPC.

Browse to **VPC > Internet Gateways** and select **Create Internet Gateway**

Select **Attach to VPC**
Creating the Route Tables for the Public and Private VPC subnets

Create two new routes for the VPC subnets and then associate the respective subnet to the route.

Browse to VPC > Route Tables and select Create Route Table

Create the Public VPC RT

Create the Private VPC RT
Now associate subnets to the new routes.

Select Private VPC RT, select the Subnet Associations tab and click Edit. Select the Private Subnet and click Save.

Repeat the same steps to associate the Public VPC RT with the Public Subnet.

Next create a default route on the Public VPC RT to the Internet Gateway.
Here is a look at the new routes

At this point, the network is almost complete. As we create the Palo Alto instance and the Linux servers, we will come back and add a few Elastic IPs.
Step 2 – Building the Palo Alto Network (PAN)

Creating Security Groups
When creating a new instance, you can add Security Groups on the fly, but we are going to create them now.

Create the Public Security Group that allows ICMP and HTTP. ICMP will be allowed so that server instances in the Security Group can be pinged for troubleshooting.

Create the Private Security Group, but don’t add any inbound rules just yet.

Now right-click on the VPC Private SG and select Edit inbound rules.
Add a rule that allows all traffic (or any traffic you want) to the VPC Private SG that originates from the VPC Private SG. This allows hosts inside the VPC Private SG to communicate. There could be instances where you need more granular ACLs, but for this tutorial let’s keep it simple.

Now create a VPC Management Security Group that will allow access to the PAN management interface, which is where you will login and manage the firewall.

*NOTE: Again, I’m using ACLs that are quite liberal. I wouldn’t recommend this for a production environment. I would restrict the ACLs to allow SSH and HTTPS only from your corporate network block or IP.*

Here is the final result for the Security Groups
Add the Palo Alto Firewall Instance

Browse over to the AWS MarketPlace, which is located at https://aws.amazon.com/marketplace. Search for Palo Alto and select a firewall. I’ve selected one that is a free trial, but select the firewall that fits your requirements.

I’m going to place the PAN in the US West (Oregon) region which is where I’ve been doing my work. When logged into the AWS console, you can view your region at the top left corner of your screen.

Next, select **Launch with EC2 Console** in the proper region. In this example, I use Oregon.

For the Instance Type, select the option that meets your needs and is a *Palo Alto supported EC2 Instance Type*. If you don’t select a supported instance type, the launch will fail.

I used a **c3.xlarge** for the Palo Alto firewall.

Configure the details of the instance. Select the proper Network, Subnet, and Enable Auto-Assign Public IP. Be sure to add the management IP (10.0.1.10) to eth0.

Click **Next: Add Storage**. I checked **Delete on Termination** because I want the EBS storage to be terminated along with the instance.
Click **Next: Tag Instance**. Click **Next: Configure Security Group**.

Click **Select an existing security group**, and then select **VPC Management SG**. Click the **Review and Launch** button.

Click **Launch**. Select your key pair, select the checkbox and then click **Launch Instances**.
Back at the EC2 Dashboard, the new Palo Alto instance will be launching. When complete, select the new instance and view the details.

It’s time to add two additional interfaces to the firewall. Browse to EC2 > Network Interfaces and select Create Network Interface. Do this two times, once for the Public interface and one for the Private interface. Assign the proper Private IP and Security Group for each.
For the Public and Private interface, right-click and select **Change Source/Dest. Check**.

Disable the Source/dest. check for the Public and Private interface.

Attach the newly created Public and Private interface to the firewall by right clicking on the interface and selecting **Attach**.
Attach the interfaces to the firewall Instance. Don’t forget to do both the Public and Private Interface.

I can view the firewall Private IP addresses by right-clicking my EC2 Palo Alto instance and selecting Manage Private IP Addresses.
Configure the Palo Alto Firewall

View the instance details of the firewall and get the Public IP.

Open a SSH client that is configured with the proper Key Pair, and connect via SSH to the PAN’s Public IP address - it’s time to reset the admin password. Login as admin, and type configure at the prompt.
Set the admin password and commit the change

```
login as: admin
Authenticating with public key "imported-openssh-key" from agent
Welcome admin.
admin@PA-VM> configure
Entering configuration mode
[edit]
admin@PA-VM# set mgt-config users admin password
Enter password : 
Confirm password : 
[edit]
admin@PA-VM# commit

..99%......100%
Configuration committed successfully

[edit]
admin@PA-VM# 
```

Login via HTTPS via your Internet browser using the new admin password.

![Login to Palo Alto](image)

It’s time to setup the interfaces on the Palo Alto firewall by selecting **Network** and then selecting **ethernet1/1**. Configure ethernet1/1 as seen below. When adding the Security Zone, select new zone.
Name the new zone **untrust** and select **OK**.

Select the IPv4 tab and select **DHCP Client**.

Select the **Advanced** tab and then the **Other Info** tab. Select **Management Profile** and **New Management Profile**. Create a new management profile for the Ping service.
Select OK and then OK again.

Select ethernet1/2 and configure this interface the same way for the private network. This time, however, create a new Security Zone named trust. Uncheck automatically create default route pointing to default gateway provided by server. Add the MGT-Ping management profile and select OK and then OK again. Commit the changes.
By selecting just left of **Dynamic-DHCP Client** you can view the interface details

Creating the Source and Destination NAT rules

Select **Policies > NAT > Add**

Create the Destination NAT to the Webserver
Source Zone: untrust

Destination Zone: untrust

Source Address: Any

Destination Address: 10.0.1.20

The destination address is the public IP of the public interface.

Add the destination address translation. The translated address is the IP address of the web server.

Now add a source NAT for the web server.
Add the Translated Packet rule using the public interface of the Palo Alto firewall, which is 10.0.1.20 in this tutorial.
Commit the changes

**Adding the Security Policies**
Browse to Policies > Security and click Add. Begin adding the first rule to allow traffic to the web server.
Now let's add a rule to allow the web server to the Internet
Now let’s adjust the implicit deny rule to log traffic that is getting blocked. Click on the interzone-default rule and click Override.
Select the **Actions** tab and check **Log at Session End**

Finally, **Commit** the changes.

**Step 3 – Creating the EC2 Linux servers behind the PAN**

We need to create an Elastic IP for the web server’s real-world IP address that will point to the Public interface of the Palo Alto (10.0.1.20).

Browse to **EC2 > Elastic IPs** and select **Allocate New Address** and work through the instructions.

Then right click the new Elastic IP and select **Associate Address**.
Select the network interface that matches the public side of the PAN and ensure the Private IP matches the IP for the public side of the PAN.

At this point, you should be able to ping the Elastic IP from your workstation, and since we didn’t allow ICMP to security policy, the Traffic Monitor on the Palo Alto will show packets being dropped.
**Building the Bastion Host**

Launch the bastion host instance that will be used to enter into the private subnet. Create this host in the Public Security Group and with two NICs - one NIC in the public subnet and another in the private subnet.

Browse to EC2 > Instances and select Launch Instance

Select one of the cheaper Linux instances.

Select **Next: Configure Instance Details**

Make sure the VDC is 10.0.0.0/16 and the subnet is the public subnet 10.0.1.0/24.

Secondly make sure you have two NICs. One in the public subnet with the bastion host public IP, and one in the private subnet with the bastion host private IP.
Click **Next**: until you get to **Configure Security Group** and place the instance in the **VPC Management SG** Security Group.
Select **Review and Launch** and then **Launch**. Finally select the security key you want to use for authentication to the server. While the instance is building, assign the Elastic IP for the bastion host, so that it is accessible from the Internet.

*This is a good time to mention security for the bastion host. In a production environment, you would want to only allow explicit access to the host from your corporate network block. For example, only allow SSH to the bastion host public IP from the corporate source NAT of your network.*

Browse to **EC2 > Elastic IPs** and allocate a new Address

Select the new Elastic IP and use the **Actions** menu to associate a new address. Select the Network Interface with the public NIC of the bastion host.

SSH to the new Elastic IP of the bastion host and login as user **ec2-user**

Use **ifconfig** to verify the server has 2 NICs and the correct IPs are configured
```
[ec2-user@ip-10-0-1-30 ~]$ ifconfig

eth0   Link encap:Ethernet  HWaddr 02:EB:71:A6:35:1A
        inet addr:10.0.1.30  Bcast:10.0.1.255  Mask:255.255.255.0
        inet6 addr: fe80::b371ff:fe61:35a5/64 Scope:Link
        UP BROADCAST RUNNING MULTICAST  MTU:9001  Metric:1
        RX packets:513  errors:0  dropped:0  overruns:0  frame:0
        TX packets:380  errors:0  dropped:0  overruns:0  carrier:0
        collisions:0  txqueuelen:1000
        RX bytes:39529 (38.1 KiB)  TX bytes:59595 (58.1 KiB)

eth1   Link encap:Ethernet  HWaddr 02:38:37:8B:5D:48
        inet addr:10.0.2.30  Bcast:10.0.2.255  Mask:255.255.255.0
        inet6 addr: fe80::b371ff:fe61:35a5/64 Scope:Link
        UP BROADCAST RUNNING MULTICAST  MTU:9001  Metric:1
        RX packets:16  errors:0  dropped:0  overruns:0  frame:0
        TX packets:17  errors:0  dropped:0  overruns:0  carrier:0
        collisions:0  txqueuelen:1000
        RX bytes:1386 (1.3 KiB)  TX bytes:1832 (1.6 KiB)
```
Creating the Web Server Instance
Like the bastion host, launch another Linux instance. This time, however, place the server in the private subnet and assign the private IP of 10.0.2.40.

Add the web server to the Private Security Group
Now, we need to ensure the bastion host in the management network can talk to the servers in the VPC Private Security Group. Browse to **EC2 > Security Groups**, right-click on the VPC Private SG and select **Edit inbound security rules**. Select **Add Rule**, and add the **VPC Management SG’s Group ID** as the Custom IP.

![Edit inbound rules](image)

Proving you have Pageant running, and your security key added to it, you can SSH to the bastion’s public IP, and then SSH to 10.0.2.40, the web server. Pageant loads the key automatically, so that you do not have to store the key on the bastion host – that would be a bad idea!

You’ll notice, however that you can’t ping out to the Internet from the web server. We need to do a couple of things first! We need to change the route for the private subnet and update the default gateway on the web server. Granted, I’ve read there are other ways to route through the Palo Alto, but I haven’t explored those at this time.

Browse to **EC2 > Network Interfaces** and copy the **Network interface ID** for the **Palo Alto Private Interface**, which should have the IP of 10.0.2.20.

Browse to **VPC > Route Tables** and select **Private VPC RT**. Select the **Routes** tab and **Edit**. Add the destination 0.0.0.0/0 to the Network interface ID of the Palo Alto Private Interface. Click **Save**.

![Route Table](image)

Login vis SSH to the bastion host, and then SSH to the web server, which is 10.0.2.40. Use the command **ip route list** to show the current routing table.

```
[root@ip-10-0-2-40 ec2-user]# ip route list
default via 10.0.2.1 dev eth0
10.0.2.0/24 dev eth0  proto kernel  scope link  src 10.0.2.40
169.254.169.254 dev eth0
```

Copyright 2015, Grant Carmichael
Make sure you are root by running the command `sudo su`.

Then use `ip route del default` to delete the default route and then `ip route add default via 10.0.2.20 dev eth0`, which makes the default route to the Palo Alto private IP. If you reboot the server, you will need to delete the default route and add the new default route again. If needed, make the routing change persistent.

The routing table should look like this:

```
[root@ip-10-0-2-40 ec2-user]# ip route show
default via 10.0.2.20 dev eth0
10.0.2.0/24 dev eth0  proto kernel  scope link  src 10.0.2.40
169.254.169.254 dev eth0
```

Now let's install the apache web server by running `yum install httpd`.

When prompted, enter `y` and hit the enter key.

```
Dependencies Resolved

Package Arch Version

Installing:
httpd x86_64 2.2.31-1.6

Installing for dependencies:
apr x86_64 1.5.0-2.11.
apr-util x86_64 1.4.1-4.17.
apr-util-ldap x86_64 1.4.1-4.17.
httpd-tools x86_64 2.2.31-1.6.
mailcap noarch 2.1.31-2.7.

Transaction Summary

Install 1 Package (+5 Dependent packages)

Total download size: 1.5 M
Installed size: 3.6 M
Is this ok [y/d/N]:
```

Start apache with `service httpd start`.
Now browse via an Internet browser to the Elastic IP that was created for the web server, and your webpage should pull up!