Check Point
SOFTWARE TECHNOLOGIES LTD.

16 October 2017

VSEC FOR OPENSTACK

R80.10

Administration Guide

Check Point
SOFTWARE TECHNOLOGIES LTD
INFINITY

Important Information

Latest Software

We recommend that you install the most recent software release to stay up-to-date with the latest functional improvements, stability fixes, security enhancements and protection against new and evolving attacks.

Latest Version of this Document

Download the latest version of this document
http://supportcontent.checkpoint.com/documentation_download?ID=58506.

To learn more, visit the Check Point Support Center
http://supportcenter.checkpoint.com.

Feedback

Check Point is engaged in a continuous effort to improve its documentation.

Please help us by sending your comments
mailto:cp_techpub_feedback@checkpoint.com?subject=Feedback on vSEC for OpenStack R80.10 Administration Guide.

Revision History

Date	Description
16 October 2017	First release of this document

Contents

Introduction

The OpenStack Foundation http://www.openstack.org is an organization that unites corporate sponsors, individual developers and a user community to develop and deploy open source software for implementing public (or private) cloud infrastructure and services, similar to (and sometimes compatible with) Amazon Web Services.

This guide shows how to deploy an R80.10 Check Point Security Gateway that protects an internal subnet, in an OpenStack-based cloud.

For successful set up, it is necessary to have expertise with:

- Check Point Security Gateway and Security Management.
- OpenStack command line components and Web UI (Dashboard). The procedures use these OpenStack command line components: `neutron` , `glance`, `cinder`, and `nova`. Some tasks can be automated with OpenStack Heat templates. Heat templates are not covered in this guide.

Important - When copying and pasting commands from this guide into a command prompt, you must manually link together in a chain the lines that end with '\'. Then, delete the '\'.

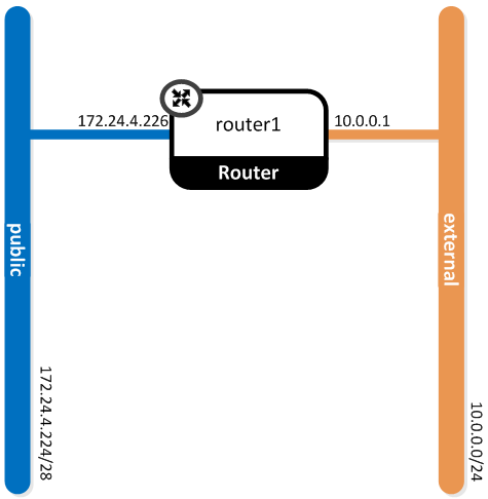
Deploying a vSEC R80.10 Gateway for OpenStack

To deploy a Check Point Gateway that secures traffic between the Internet and an internal network in an OpenStack-based cloud:

1. Deploy an internal OpenStack subnet that is routed to the Internet through a Check Point Gateway instance.
2. Install an OpenStack-ready Gaia image on the Gateway instance.

OpenStack deployment needs a router ("router1") with:

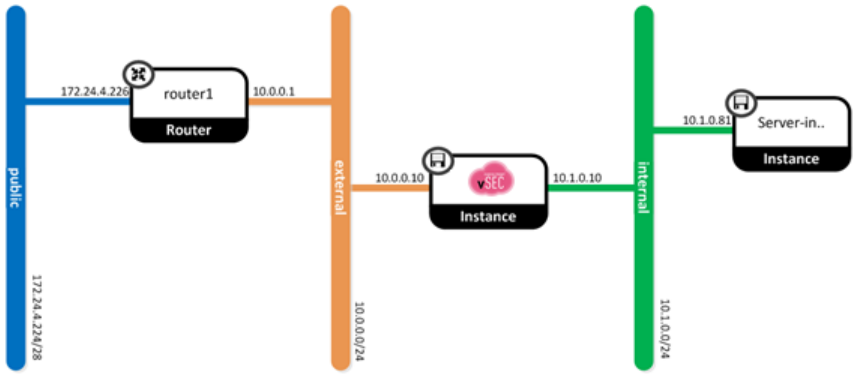
- One interface connected to a public network ("public").
- One interface connected to an external network ("external").



Defining the Network Topology

To deploy vSEC for OpenStack, make these changes to the OpenStack default deployment:

- Add a network "internal".
- Configure a subnet ("internal-subnet") with the IP addresses in the "internal" network.
- Launch a Check Point Gateway instance that connects the "external" network and the "internal" network.



Two ways to configure the gateway:

- As a Security Gateway which can then connect from a Security Management Server that is not in the cloud, over the Internet, to the Security Gateway.
- As a Security Gateway and Security Management Server (a standalone deployment) which can then connect from a SmartConsole over the Internet to the Security Management Server.

The Check Point Security Gateway inspects all traffic to and from the Internet, and protects servers that are connected to the "internal" network.

Creating a Security Group

The Gateway instance enforces the Security Policy configured using Check Point Security Management. Therefore, define the Security Group for traffic over the interfaces with a permissive policy.

1. Define the Security Group.
2. Add a permissive rule.

```
$ neutron security-group-create \
    --description 'A permissive security group to be applied to the gateway' \
    gateway-security-group

$ neutron security-group-rule-create \
    --direction ingress \
    --remote_ip_prefix 0.0.0.0/0 \
    gateway-security-group
```

Creating the Internal Network

Create the "internal" network: `$ neutron net-create internal`

Creating the Internal Subnet

Create the internal subnet and attach it to the "internal" network.

1. Replace INTERNAL-START and INTERNAL-END with the IP address range for the "internal" network subnet.
2. Replace INTERNAL-SUBNET-CIDR with the CIDR format subnet address specification.
3. Replace INTERNAL-GATEWAY_ADDRESS with an IP address from the internal subnet that will be used as the address of the Gateway interface connected to this subnet.

```
$ neutron subnet-create \
    --name internal-subnet \
    --allocation_pool start=INTERNAL-START,end=INTERNAL-END \
    --gateway INTERNAL-GATEWAY-ADDRESS \
    internal \
    INTERNAL-SUBNET-CIDR
```

Creating the Internal Gateway Port

Create the internal Gateway port. This will be the interface of the Gateway that is attached to the "internal" network.

Replace INTERNAL-GATEWAY-ADDRESS with the Security Gateway address defined in *Creating the Internal Subnet* (on page 8).

```
$ neutron port-create \
    --name internal-gw-port \
    --fixed-ip ip_address=INTERNAL-GATEWAY-ADDRESS \
    --security-group gateway-security-group \
    internal
```

Allowing Traffic from the Internet through the Internal Gateway Port

Edit the internal Gateway port to allow traffic from the Internet to the internal network. Traffic will pass through "router1", through the Gateway, to the internal network.

To allow traffic from any source IP address, edit the internal Gateway port:

```
$ neutron port-update \
    internal-gw-port \
    --allowed_address_pairs type=dict list=true ip_address=0.0.0.0/1 \
    ip_address=128.0.0.0/1
```

Creating the External Gateway Port

Create the external Gateway port. The external port is used as the interface of the Gateway that is attached to the "external" network. The "external" network is connected to the "public" network (and the Internet) through "router1".

Replace EXTERNAL-GATEWAY-ADDRESS with an IP address from the subnet of the "external" network.

```
$ neutron port-create \
    --name external-gw-port \
    --fixed-ip ip_address=EXTERNAL-GATEWAY-ADDRESS \
    --security-group gateway-security-group external
```

Allowing Traffic from the Internal Network through the External Gateway Port

Edit the external Gateway port to allow traffic from the internal subnet addresses to the external Gateway port. Traffic from the internal network is allowed through the Gateway, routed to the "external" network, to "router1" and the Internet.

To allow traffic with a source IP address in the internal subnet IP address range, edit the external Gateway port:

Replace INTERNAL-SUBNET-CIDR with the internal subnet IP address range.

```
$ neutron port-update \
    external-gw-port \
    --allowed_address_pairs type=dict list=true \
    ip_address=INTERNAL-SUBNET-CIDR
```

Adding a Route to the Internal Network

Update the router with a route to the internal network.

Replace EXTERNAL-GATEWAY-ADDRESS and INTERNAL-SUBNET-CIDR with the IP addresses defined earlier.

```
$ neutron router-update \
    router1 \
    --routes type=dict list=true nexthop=EXTERNAL-GATEWAY-ADDRESS,\
    destination=INTERNAL-SUBNET-CIDR
```

Downloading the Image

Download the R80.10 image for OpenStack from the R80.10 home page, sk111841 http://supportcontent.checkpoint.com/solutions?id=sk111841, under the **Additional Downloads and Products**.

Note - Do this step one time in an OpenStack deployment (one image can be used to launch many instances).

Run these `glance` image management commands:

```
$ glance image-create \
    --name Check-Point-R80.10-image \
    --disk-format qcow2 \
    --container-format bare \
    --file Check_Point_R80.10_for_OpenStack.qcow2
```

Creating a Machine Flavor

A machine flavor is the list of resources allocated to a virtual machine instance. You can create a flavor with the resources required to run the image, or use an existing flavor that meets the minimum resource requirements.

The 2048 parameter in the command line is a RAM quantity of 2048 MiB http://en.wikipedia.org/wiki/Mebibyte. You can set it to a different value depending on the expected load.

You must do this operation as the OpenStack admin. Use the nova Compute management component:

```
$ env OS_USERNAME=admin nova flavor-create 2048MiB-50GiB-1CPU auto 2048 50 1
```

Importing a Key Pair

Import a key pair into the OpenStack environment. Replace KEY-NAME with the name of the key. Use the Web UI or a nova Compute command:

```
$ nova keypair-add --pub_key ~/.ssh/id_rsa.pub KEY-NAME
```

Launching a Gateway Instance

Before you launch a Gateway instance, extract the identifiers of the Gateway ports created in the earlier steps.

You can launch a Gateway instance with nova Compute component in the first-time wizard or through a completely automated scripted process.

To launch a Gateway instance using the First-time Wizard:

1. Run these commands:

```
$ external_gw_port_id=`neutron port-show external-gw-port | awk '/ id
/{print $4}'`
$ internal_gw_port_id=`neutron port-show internal-gw-port | awk '/ id
/{print $4}'`
$ nova boot \
    --flavor 2048MiB-50GiB-1CPU \
    --key-name KEY-NAME \
    --image Check-Point-R80.10-image \
    --nic port-id=$external_gw_port_id \
    --nic port-id=$internal_gw_port_id \
    --config-drive=true \
    R80.10-instance
```

2. Run the First Time Wizard ("Configuring the Check Point Gateway" on page 12).

To launch a Gateway instance with an automated script:

1. Create a USER-SCRIPT that runs at the time the computer boots up for the first time. Use these commands in the script to configure the password for the OS admin user and to set the gateway as a standalone Security Management Server and Security Gateway.

   ```
   #!/bin/bash
   clish -c 'set user admin password-hash ADMIN-PASSWORD-HASH' -s
   (config_system -s
   'install_security_gw=true&install_ppak=true&install_security_managment
   =true&install_mgmt_primary=t
   rue&install_mds_primary=false&mgmt_admin_name=MANAGEMENT-ADMIN-USERNAM
   E&mgmt_admin_passwd=MANAGEME
   NT-ADMIN-PASSWORD&mgmt_gui_clients_radio=any' ; shutdown -r now &)
   ```

   Where:

   - ADMIN-PASSWORD-HASH - the hash of the OS admin user password. Generate the password hash with: `openssl passwd -1 <password>`
   - INTERNAL-GATEWAY-ADDRESS and INTERNAL-GATEWAY-MASKLEN - the internal network address and mask size in bits
   - EXTERNAL-GATEWAY-ADDRESS and EXTERNAL-GATEWAY-MASKLEN - the internal network address and mask size in bits
   - MANAGEMENT-ADMIN-USERNAME and MANAGEMENT-ADMIN-PASSWORD - the initial management administrator user name and password (not hash)
   - USER-SCRIPT - the name of the script file

2. Launch the instance with the flavor, key pair, image, and interfaces defined in the earlier steps:

   ```
   $ external_gw_port_id=`neutron port-show external-gw-port | awk '/ id
   /{print $4}'`
   $ internal_gw_port_id=`neutron port-show internal-gw-port | awk '/ id
   /{print $4}'`
   $ nova boot \
       --flavor 2048MiB-50GiB-1CPU \
       --key-name KEY-NAME \
       --image Check-Point-R80.10-image \
       --nic port-id=$external_gw_port_id \
       --nic port-id=$internal_gw_port_id \
       --user-data USER-SCRIPT \
       --config-drive=true \
       R80.10-instance
   ```

   Several minutes after you configure the Check Point gateway for the first time, you can use the Gaia WebUI to configure your Gateway.

An alternative to the **nova boot** command line, is a **heat** template. Specify the `user_data_format` property of the `OS::Nova::Server` resource to be `"RAW"`.

If you use the `admin_pass` property (or the `--admin-pass` option in **nova boot**), you can do without the clish command in the script above:

```
clish -c 'set user admin password-hash ADMIN-PASSWORD-HASH' -s
```

Configuring the Check Point Gateway

To configure the Check Point Gateway:

1. Set the admin password of the Gateway.
2. Connect to the Gaia WebUI portal of the Gateway.
3. Configure the Gateway instance as a Security Gateway, or as a Security Gateway and Security Management Server (a standalone deployment).

Note - The host IP address and the default route are set automatically. Do not change them.

To set the administrator password:

1. Set the administrator password. Run:
   ```
   set user admin password
   ```
   At the prompt, enter the administrator password.
2. Run:
   ```
   save config
   ```
3. Exit the gateway shell. Run:
   ```
   exit
   ```

To configure the Check Point Gateway:

1. Using a browser, connect to https://<GATEWAY-FLOATING-IP >
2. In the Gaia Portal window, log in using the administrator name (admin) and password that you defined earlier.
3. The WebUI shows the First Time Configuration Wizard.
   Click **Next**.
4. Set the date and time (manually, or enter the hostname or IP address of the NTP server).
   Click **Next**.
5. Set the host name for the appliance.
6. **Optional:** Set the domain name, and IPv4 addresses for the DNS servers. You can configure IPv6 DNS servers.
   Click **Next**.
7. The interface page shows the internal IPv4 address of the interface. Do not change this setting.
   Click **Next**.
8. Set the username and password for the Security Management Server administrator account.
   Click **Next**.
9. Select **Security Gateway** and/or **Security Management**
   Note: ClusterXL is not supported.
   Click **Next**.
10. Define the GUI Clients that can log in to the Security Management Server.
    Click **Next**.
11. If the **Help Check Point Improve Software Updates** window shows, click **Yes** or **No** as necessary.
12. Click **Finish** and then **OK**.

Note - You can also automate the first time configuration. To do that, refer to sk69701 http://supportcontent.checkpoint.com/solutions?id=sk69701.

Associating a Public Address with the Gateway Instance

The "external" IP address of the launched Gateway is allocated from the "external" network. This address is usually not reachable from the Internet. To make the external address reachable, select a "floating" IP address from the "public" network subnet. Then, associate this address with the address of the external interface of the Gateway.

1. Extract the external interface identifier.
2. Choose and assign a public IP address and print the address. We will refer to this IP address as GATEWAY-FLOATING-IP.

```
$ external_gw_port_id=`neutron port-show external-gw-port | awk '/ id /{print $4}'`

$ neutron floatingip-create \
    --port-id $external_gw_port_id \
    public \
  | awk '/ floating_ip_address /{print $4}'
```

Securely Accessing the Gateway Instance

Use SSH Connect to the Gateway instance as the admin user . Compare the public fingerprint from the Gateway instance, to the public key from the OpenStack Console.

If you do not compare the fingerprints, you are vulnerable to a man-in-the-middle attack on your SSH session.

Note - It can take a few minutes after the launch of an instance before the system log is available on the OpenStack Console.

To get the SSH public key fingerprints of the Gateway from the OpenStack Console:

1. Open an SSH client.
2. Run: `$ nova console-log R80.10-instance | grep '^ec2:'`

Note the fingerprint strings. One of these fingerprints will match the key fingerprint that is presented when making an SSH connection to the Gateway for the first time.

To connect to the Gateway instance over SSH:

1. Run: `$ ssh admin@GATEWAY-FLOATING-IP`

   The address GATEWAY-FLOATING-IP is the public IP address that was associated with the Gateway in Associating a Public Address with the Gateway Instance (on page 13).
2. Compare the public key fingerprint with the string sent by the Gateway.

Deploying Servers on the Internal Network

After you install a Policy on the Security Gateway from the Security Management Server, launch server instances on the "internal" network. Traffic to and from the servers is protected by the Gateway.

The servers must be from a vendor or configuration that is supported by the OpenStack deployment.

Launch instances using the command line, using a OpenStack `Heat` template, or using the OpenStack Horizon Web UI.

Note: The Havana release of OpenStack does not allow associating a public floating IP to a server that is not directly connected to the router. Therefore, you cannot give a public IP address to servers in the "internal" network. However, you can allocate additional public floating IP addresses to the external interface of the Check Point Security Gateway, and then use Static NAT rules to redirect traffic for these addresses to servers in the "internal" network.