Check Point
SOFTWARE TECHNOLOGIES LTD.

20 September 2018

CHECK POINT CLOUDGUARD IAAS HIGH AVAILABILITY FOR MICROSOFT AZURE R80.10

Deployment Guide

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.

Certifications

For third party independent certification of Check Point products, see the Check Point Certifications page https://www.checkpoint.com/products-solutions/certified-check-point-solutions/.

Latest Version of this Document

Open the latest version of this document in a Web browser https://sc1.checkpoint.com/documents/R80.10/WebAdminGuides/EN/CP_CloudGuard_IaaS_R80.10_HighAvailability_for_Azure/html_frameset.htm.

Download the latest version of this document in PDF format http://downloads.checkpoint.com/dc/download.htm?ID=67723.

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 Check Point CloudGuard IaaS High Availability for Microsoft Azure R80.10 Deployment Guide.

Revision History

Date	Description
20 September 2018	Updated: Step 1: Deploy with a Template in Azure (on page 16) - added "Standard Load Balancers and High Availability ports are not available on the Azure Government Cloud environment". Updated: Known Limitations (on page 40) - added "Standard Load Balancers and High Availability ports are not available on the Azure Government Cloud environment".
17 September 2018	First release of this document.

Contents

Terms

Active Directory (AD)

Active Directory. Microsoft® directory information service. Stores data about user, computer, and service identities for authentication and access.

Active Member

A cluster member that handles network connections that pass through the cluster. In a cluster deployment, only one cluster member is Active and can handle connections.

Availability Set

A collection of Virtual Machines that are managed together to provide application redundancy and reliability. The use of an availability set ensures that during either a planned or unplanned maintenance event at least one Virtual Machine is available. (Description from the Microsoft Azure glossary).

Azure Environment

An Azure environment an independent deployment of Microsoft Azure, such as Azure Cloud for global Azure and Azure China Cloud for Azure operated by 21Vianet in China.

Azure PowerShell

A command-line interface to manage Azure services via a command line from Windows. (Description from the Microsoft Azure glossary)

Check Point WatchDog

A process that launches and monitors critical processes such as Check Point daemons on the local machine, and attempts to restart them if they fail.

Cluster

Two or more Security Gateways that work together in a redundant configuration - High Availability.

Failover

Also, Fail-over. Transferring of a control over traffic (packet filtering) from a cluster member that suffered a failure to another cluster member (based on internal cluster algorithms).

Load Balancer

A resource that distributes incoming traffic among computers in a network. In Azure, a load balancer distributes traffic to Virtual Machines defined in a load-balancer set. A load balancer can be Internet-facing, or it can be internal. (Description from the Microsoft Azure glossary)

Resource

An item that is part of your Azure solution. Each Azure service enables you to deploy different types of resources, such as databases or Virtual Machines. (Description from the Microsoft Azure glossary)

Resource Group

A container in Resource Manager that holds related resources for an application. The resource group can include all of the resources for an application, or only those resources that are logically grouped together. You can decide how you want to allocate resources to resource groups based on what makes the most sense for your organization.

SmartConsole

Check Point main GUI client used to create and manage the security policy.

Standby

State of a cluster member that is ready to be promoted to Active state (if the current Active member fails). Applies only to ClusterXL High Availability Mode.

Subnet

A logical subdivision of an IP network.

User Defined Routing (UDR)

A route table or a set of rules to create network routes so that your Virtual Machine

can handle the traffic between subnets and to the Internet.

Virtual Machine (VM)

The software implementation of a physical computer that runs an operating system. Multiple virtual machines can run simultaneously on the same hardware. In Azure, virtual machines are available in a variety of sizes. (Definition from the Azure glossary).

Virtual Network (Virtual Network)

A network that provides connectivity between your Azure resources that is isolated from all other Azure tenants. An Azure VPN Gateway lets you establish connections between Virtual Networks and between a Virtual Network and an on-premises network. You can fully control the IP address blocks, DNS settings, Security Policies, and route tables within a Virtual Network. (Description from the Microsoft Azure glossary)

Check Point CloudGuard IaaS R80.10 High Availability for Azure

In This Section:

Prerequisites

To set up your system most efficiently, you have to be familiar with these topics:

Microsoft Azure

- Virtual Networks
- Virtual Machines
- Load Balancers
- High Availability ports
- Public IP addresses
- User Defined Rules (UDR)
- Role Based Access Control (RBAC)

Check Point

- R80.10
- Check Point with Microsoft Azure

Setting Up R80.10 Check Point High Availability Clusters in Azure

A cluster is a group of Virtual Machines that work together in High Availability Mode. One member is the Active member and the second member is the Standby member. The Active member fails over to the Standby member when necessary.

- Cluster members communicate with unicast.
- For inbound, outbound, and East-West traffic, cluster members rely on Azure load balancers to represent their external and internal virtual IP addresses. Load balancers only forward traffic to the Active member.
- For VPN traffic, members use API calls to Azure to communicate the failover of the Active cluster member. The Standby member is then promoted to Active.

  During failover, the Standby member associates the private and public cluster IP address of the Active cluster member, with its external interface.

Azure API authentication

To be able to automatically make API calls to Azure, cluster members need Azure Active Directory credentials. Use Role-Based Access Control (RBAC) to enable Active Directory.

The Check Point cluster members are managed by a Check Point Security Management Server in the Azure Cloud, or on-premises.

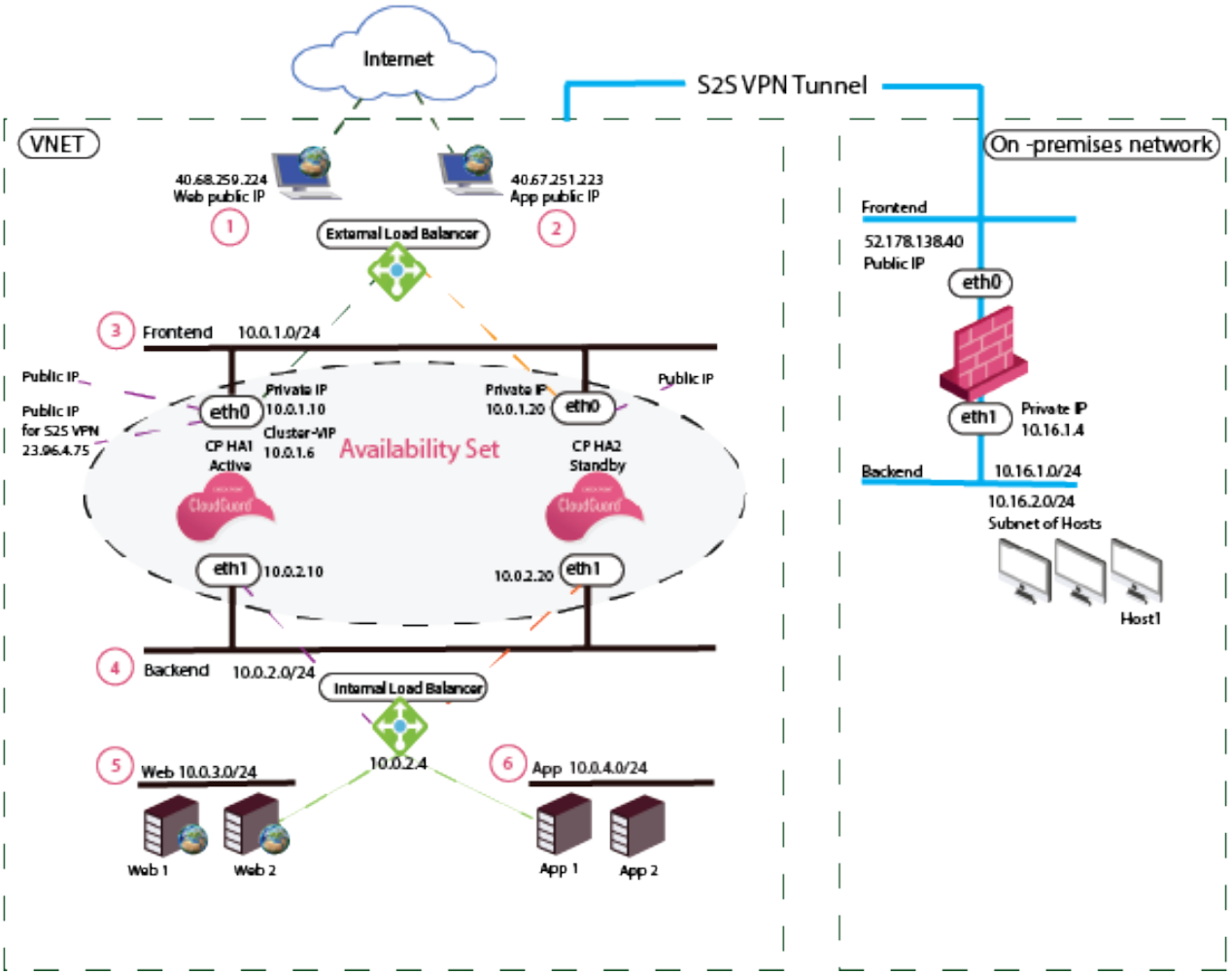
Network

In This Section:

Follow this network diagram to configure your system, but make sure to replace the IP addresses in the sample environment with the IP addresses in your environment.

Network Diagram

See the routing tables below the diagram.



Load Balancing Rules of the External Load Balancer

1	**Example 1**	**Frontend** Web:443	**Backend port** 8081
2	**Example 2**	**Frontend** App:80	**Backend port** 8083

Frontend Routing Table - User Defined Routes (UDR)

3	**Destination** 10.0.0.0/16	**Nexthop** None (Drop)
	10.0.1.0/24	Virtual Network

Backend Routing Table - UDR

4	**Destination** 0.0.0.0/0	**Nexthop** None (Drop)

Routing Table for Web and App - UDR

Web and App routing tables have the same Virtual Network address but different subnet addresses.

Web

5	**Frontend**	**Nexthop**
	10.0.0.0/16 - *Virtual Network address*	**10.0.2.4 -** *IP of the Internal Load Balancer*
	0.0.0.0/0	**10.0.2.4 -** *IP of the Internal Load Balancer*
	10.0.3.0/24 (Web) - *Subnet address*	Virtual Network

App

6	**Frontend**	**Nexthop**
	10.0.0.0/16 - *Virtual Network address*	**10.0.2.4 -** *IP of the Internal Load Balancer*
	0.0.0.0/0	**10.0.2.4 -** *IP of the Internal Load Balancer*
	10.0.4.0/24 (App) - *Subnet address*	Virtual Network

Diagram Components

The diagram shows:

- Virtual Network in Azure that is divided into four subnets
  - Frontend
  - Backend
  - Web
  - App
- On-premises network with these components
  - Security Gateway
  - Hosts

Check Point R80.10 High Availability consists of two cluster members, Member 1 and Member 2. Each member has two interfaces.

When the cluster members are in the same Availability Set, it guarantees that the two members are in separate fault domains. For more information, see Managing the availability of Virtual Machines https://docs.microsoft.com/en-us/azure/virtual-machines/windows/manage-availability?toc=%2Fazure%2Fvirtual-machines%2Fwindows%2Ftoc.json.

In the diagram:

- The R80.10 cluster is protecting two web applications.
- There is Site-to-Site VPN connectivity between the cluster members and on-premises gateways.

Each web application has:

- Public IP address
- Web server
- Application server

Manually configure these components:

- Backend hosts
- Subnets
- Routing tables for Web and App servers

Static IP Addresses

Name	Attached to	Use
Cluster public address	The external interface of the Active member.	VPN
Cluster private address	The external interface of the Active member.	VPN
Member 1 public address	The external interface of Member 1.	• External management of Member 1 • Internet and Azure API access Do not disable or delete this resource.
Member 2 public address	The external interface of Member 2.	• External management of Member 2 • Internet and Azure API access Do not disable or delete this resource.
Web	Azure Load Balancer	Public service Web
App	Azure Load Balancer	Public service App

Use the Azure Load Balancer rules to forward traffic that comes from the Internet.

Note - The following ports cannot be used:

- 80
- 443
- 444
- 8082
- 8880
- 8117

Frontend IP address	Frontend TCP ports	Destination IP address	Destination port
Web	HTTPS	Active cluster member	8081
App	HTTP	Active cluster member	8083

Failover

This is what happens at failover:

1. The member that fails, immediately stops responding to the Load Balancer health probes.
2. The member that gets promoted to Active, starts responding to the Load Balancer health probes.
3. The Azure External Load Balancer and Internal Load Balancer detect the new health status of each member, and forward traffic to the healthy member. See Azure Load Balancer probes https://docs.microsoft.com/en-us/azure/load-balancer/load-balancer-custom-probe-overview for more information.

   This usually happens in under 15 seconds based on the health probe Load Balancer configuration. This affects inbound, outbound, and East-West traffic inspection.
4. The member that gets promoted to Active uses the Azure API to associate itself with the cluster private and public IP addresses.

   This usually happens in under 2 minutes. This affects VPN tunnel failover.

These are the expected failover times based on use case:

Use Case	Expected Failover Time	Comments
Site-to-Site VPN	Under 2 minutes	Depends on the Azure API.
Inbound inspection via the External Load Balancer	Under 15 seconds	Depends on the Load Balancer health probe.
Outbound inspection via the Internal Load Balancer	Under 15 seconds	Depends on the Load Balancer health probe.
East-West inspection via the Internal Load Balancer	Under 15 seconds	Depends on the Load Balancer health probe.

Traffic Flows

If the Security Management Server is in the Virtual Network, make sure to have specific routes to allow traffic between the Management Server Virtual Machine and the cluster members.

Note - No other Virtual Machines can be deployed in the R80.10 solution subnets.

Inbound Traffic

- Traffic travels into the External Load Balancer.
- The External Load Balancer forwards the traffic to the Active cluster member.
- The Active member inspects the traffic, and forwards it to the destination.

Inbound Traffic Reply

- The traffic travels from the Web Server to the Internal Load Balancer.
- The Internal Load Balancer forwards it to the Active member.
- The Active member forwards it to the destination.

Outbound Traffic

- Traffic travels to an Internal Load Balancer based on the UDR.
- The Internal Load Balancer forwards the traffic to the Active member.
- The Active Member inspects the traffic and forwards it to the destination.

East-West Traffic

- Traffic travels from one of the internal servers to the Internal Load Balancer of the Check Point solution.
- The Internal Load Balancer forwards the traffic to the Active member.
- The Active member forwards the traffic to the destination.

Note - The Internal Load Balancer deploys by default as part of the solution template. It is automatically configured to listen and forward any TCP or UDP traffic High Availability ports. It gets an automatically assigned name: backend-lb.

Probes monitor the health of the cluster members on TCP port 8117 from the source IP address: 168.63.129.16

Inbound VPN Traffic

- Packet enters the frontend NIC of the Active member.
- Packet is decrypted by the Active member.
- Packet is forwarded to its destination.

Intra-Subnet Traffic

- Traffic travels freely in the subnet without inspection.

Workflow for Setting Up a High Availability Gateway in Azure

In This Section:

Step 1: Deploy with a Template in Azure

Deploy this solution through the Azure Portal. If you use a different environment than the Standard Azure environment, see Using a Different Azure Cloud Environment (on page 36).

- To access the Standard Azure environment, from the Azure Marketplace, see the Azure standard portal https://portal.azure.com/#create/checkpoint.vsecha.
- To access the Azure US Government environment, from the Azure Marketplace, see the Azure US Government portal https://portal.azure.us/#create/checkpoint.vsecha.

Note - Standard Load Balancers and High Availability ports are not available on the Azure Government Cloud environment.

When the template shows, enter information for these parameters:

Parameter	Description
Cluster object name	Name of the cluster object resource group.
Credentials	Public key or user name and password for SSH connections to the cluster members.
Subscription	Azure subscription into which the cluster object is deployed.
Resource group	Azure resource group into which the cluster object is deployed.
Location	Location into which the cluster object is deployed.
License	Type of license: • Bring your own license (BYOL) • Pay as you go (PAYG)
Virtual Machine size	Size of each Virtual Machine instance in the cluster object.
SIC	SIC key to the Security Management Server.
Network setting	1. Pre-existing Virtual Network and its subnets 2. Name of a new Virtual Network and subnets into which the cluster object is deployed. **Note** - When you use pre-existing subnets, make sure that: • No other Virtual Machines are deployed in those subnets. • Define UDRs properly for each subnet. See Setting up Route Tables ("Step 3: Set Up Internal Subnets and Route Tables" on page 20). • There is a Network Security Group (NSG) associated with your Frontend subnet to connect the External Load Balancer and cluster member.

Components of the Check Point Solution

The Check Point deployed solution has these components:

- Frontend subnet

  The NSG is associated with the frontend subnet and allows all inbound and outbound TCP and UDP traffic
- Backend subnet
- Two R80.10 Virtual Machines configured as a Check Point cluster
- Internal Load Balancer
- External Load Balancer
- Public IP address for each R80.10 cluster member

No other Virtual Machines can be deployed in the solution's subnet.

Notes about the template

- You can create a new Virtual Network or deploy into an existing Virtual Network.
- Web and App subnets are not deployed automatically.
- It does not deploy any other Virtual Machines in the solution's frontend and backend subnets.
- Virtual Machines that are launched in the backend subnets may require Internet access to finalize provisioning. Launch these Virtual Machines only after you have applied *Hide NAT* rules on the cluster object to support this type of connectivity.
- The Check Point First Time Configuration Wizard automatically deploys after you have set up the cluster object. The cluster object is configured based on the parameters you apply.
- After the First Time Configuration Wizard completes, the Virtual Machines automatically reboot.

Important - If you deploy the solution to an existing Virtual Network, confirm that there is an NSG associated with the frontend subnet that allows all inbound and outbound TCP and UDP traffic. An NSG is necessary to connect to members successfully.

Step 2: Set Credentials in Azure

By default, the automatic service principal is deployed. If you want to create your own service principal, make sure you set credentials and assign privileges to necessary resources. Managed service identity for Virtual Machines is only available in the Azure Cloud environment.

If you deploy in other environments, you have to create your own service principal manually. See Creating your Own Service Principal (on page 18).

Azure Credentials and the Automatic Service Principal

The R80.10 cluster template automatically creates a service principal for each Virtual Machine, and assigns a Contributor role to the cluster resource group. Therefore, there is no need to create a service principal, assign it a role, and attach it to each of your individual cluster resources. For more information, see What is Managed Service Identity for Azure resources https://docs.microsoft.com/en-us/azure/active-directory/managed-service-identity/overview.

After you deploy a Check Point cluster, the automatic credentials can be found in Azure Portal **> Resource groups > <***cluster_resource_group* **> Access control (IAM)**. There are two service principals for each cluster member, each with a Contributor role.

Notes:

- If you delete the cluster member's Virtual Machine, the credentials are also deleted.
- Service principals never expire.

Creating Your Own Service Principal

From the Azure website, see Create an Azure Active Directory Application and Service Principal https://docs.microsoft.com/en-us/azure/azure-resource-manager/resource-group-create-service-principal-portal.

Use these parameters:

Field	Parameter
Name	Application_Name **Example:** `check-point-<cluster>`
Application type	Web-App / API
Sign-on URL	`https://localhost/Application_Name` **Example:** `https:/localhost/check-point-<cluster>`

After you create the application, write down these values.

- ApplicationId
  `client_id`
- Key value
  `client_secret`
- Tenant ID (Directory ID)
  `tenant`

Best practice - We recommend that you set the key to never expire. Go to your resource.

To create a service principal:

1. Click **Access control (IAM) > Add**.
2. Select your role.
3. Select your AD application.
4. Click **Save**.
5. Set the `client_id` and `client_secret` on each of the cluster members.

   From Expert Mode, run the following command on each cluster member:

   `# azure-ha-conf --client-id` <*ApplicationId*> `--client-secret` <*key value*> `--force`

   Example:

   ```
   # azure-ha-conf --client-id '5c1896fe-26b6-4a5b-8c81-34ae07c09a24'
   --client-secret '2G6E_|]Y&|@Il(L}-O>g' --force
   ```

Note - Use single quotes to avoid shell expansion.

1. To validate the file syntax, from Expert Mode run:
   ```
   # python -m json.tool $FWDIR/conf/azure-ha.json
   ```
2. To reload the cluster Azure configuration, from Expert Mode run:
   ```
   # $FWDIR/scripts/azure_ha_cli.py reconf
   ```

To revert to your previous automatic credentials:

1. Remove your service principal.
2. From Expert Mode, run:
   ```
   # azure-ha-conf --system-assigned --force
   ```
3. Assign the two service principals to each resource and to cluster members. See Template Components ("Components of the Check Point Solution" on page 17) for more information.
4. The service principal deploys automatically. If you want to create a new service principal, assign the privileges to the necessary resources and to cluster members. See Azure Role-Based Access Control https://docs.microsoft.com/en-us/azure/role-based-access-control/role-assignments-portal for more information.

Step 3: Set Up Internal Subnets and Route Tables

You can use the Azure portal or the CLI to add internal subnets. Let's add the Web and App subnets to our Virtual Network.

For each internal subnet, you have to create an Azure routing table with these UDRs:

Web Route Table

	Name	**Address prefix**	**Nexthop-type**	**Nexthop-address**
1	<*web-subnet*>-local	*<10.0.3.0/24>*	Virtual Network	-
2	web-subnet-to-other-subnets	10.0.0.0/16	Virtual appliance	ILB-internal-address 10.0.2.4
3	web-subnet-default	0.0.0.0/0	Virtual appliance	ILB-internal-address 10.0.2.4

App Route Table

	Name	**Address prefix**	**Nexthop-type**	**Nexthop-address**
1	<*app-subnet*>-local	10.0.4.0/24	Virtual Network	
2	*<app-subnet-to-other-subnets>*	*<10.0.0.0/16>*	Virtual appliance	ILB-internal-address 10.0.2.4
3	app-subnet-default	0.0.0.0/0	Virtual appliance	ILB-internal-address 10.0.2.4

Note - If traffic inspection is required inside the Web/App subnets, override Rule 1 in the route tables above, <*web-subnet*>-local, and <*app-subnet*>-local.

Important - Associate the newly created routing table with the subnet to which it belongs.

If the subnet houses the Security Management Server that manages the cluster members, add the following routes as well. This allows the Security Management Server to communicate directly with each cluster member, without passing through the Active member.

For example:

Name	Address-prefix	Nexthop type	Nexthop address
Subnet-name-cluster_member1-management	cluster_member1-internal-address/32 *<10.0.2.10/32>*	Virtual appliance	cluster_member1-internal address *<10.0.2.10>*

Name	Address-prefix	Nexthop type	Nexthop address
Subnet-name-cluster_member2-management	cluster_member2-internal-address/32 *<10.0.2.20/32>*	Virtual appliance	cluster_member2-internal address *<10.0.2.20>*

Step 4: Set Up Routes on Cluster Members to the Internal Subnets

SSH into each of the cluster members and add this route:

```
clish -c 'set static-route <virtual-network-prefix> nexthop gateway address
<eth1-router> on' -s
```

Example:

```
clish -c 'set static-route 10.0.0.0/16 nexthop gateway address 10.0.2.1 on' -s
```

Parameters:

- *<virtual-network-prefix>* is the prefix of the entire Virtual Network.
  Example: 10.0.0.0/16
- *<eth1-router>* is the first unicast IP address on the subnet to which eth1 is connected
  Example: 10.0.2.1

Notes:

- If the Virtual Network is comprised of several non-contiguous address prefixes, repeat the command for each prefix.
- For vNET Peering:
  - Add a compatible route on each peer network.
  - Add the route for vNET Peering to each cluster member.

Step 5: Configure Cluster Objects in SmartConsole

To configure SmartConsole for the Cluster:

1. Click the **Objects** menu click **More Object types > Network Object > Gateways and Servers > Cluster > New Cluster**.
2. Select **Wizard Mode**.

   The **Check Point Installed Gateway Cluster wizard** window shows.
3. Enter a **Cluster Name**.

   Example: checkpoint-cluster
4. In the **Cluster IPv4 Address** field, enter the public address allocated for the cluster.

   Note - You can find the cluster IP address in the Azure portal when you select the Active member's primary **NIC > IP configuration > "cluster-vip"**.
5. Click **Next**.

   The **Gateway Cluster Properties** window shows.
6. Click **Add**.
   a) In the **Name** field, enter the first cluster member name.

      Example: member1

   b) In the **IPv4 address** field:

      If you are managing the cluster from the same Virtual Network, enter the member private IP address.

      Otherwise:

      Enter the member public IP address.

   c) In the **Activation Key** field, enter the SIC key you set up in Azure.

   d) In the **Confirm Activation key** field, enter the key and click initialize.

      If the activation key is confirmed, the **Trust State field** shows: **trust established**

   e) Click **OK**.
7. Now add the second cluster member.
8. Click **Next**.

   The **Cluster Topology** window shows.
9. Select **Cluster Synchronization > Primary > Next**.
10. Select **Cluster Synchronization > Secondary > Next**.
11. Select **Edit Cluster's Properties > Finish**.

12. Review the cluster configuration.
    a) Select **Network Management**.
    b) Double-click **eth0**.

       The **Network eth0** window shows.
    c) From the **General** tab, in the **Network type** field, select **Cluster + Sync**.
    d) In the **Virtual IPv4** field, enter the private VIP address and subnet mask of the cluster. In the diagram, the private VIP address is: 10.0.1.6

       Note - You can find the cluster private VIP address in the Azure portal when you select the Active member **primary NIC > IP configuration > cluster-vip**.
    e) From the **Network eth0** window **> Topology**, disable **Anti-Spoofing** from the network interface.
    f) Click **OK**.
13. Install policy on the cluster.

Step 6: Configure NAT Rules

1. From SmartConsole, create these NAT rules to provide Internet connectivity from the internal subnets.
2. Create a network object in SmartConsole, see Creating Objects in SmartConsole (on page 39).
3. Use the table below to configure the rules.

Original packet					Translated packet			Notes
	Source	**Destination**	**Service**		**Source**	**Destination**	**Service**	
1	Virtual Network	Virtual Network	Any		=Original	=Original	=Original	Avoid NAT in the Virtual Network
2	App-subnet	App-subnet	Any		=Original	=Original	=Original	Automatic rule
3	App-subnet	Any	Any		App-subnet (hidden address)	=Original	=Original	Automatic rule
4	Web-subnet	Web-subnet	Any		=original	=Original	=original	Automatic rule
5	Web-subnet	Any	Any		Web-subnet (hidden address)	=Original	=original	Automatic rule

Notes on NAT rules:

Rule 1 - You have to *manually* define this NAT rule.

Rules 2 - 5 - These are *automatic* NAT rules.

Traffic between the Web subnet and the App subnet is based on the UDR rules. Each subnet has its own routing table.

For each internal subnet, create a network object:

1. Double-click the Web-subnet object.
2. The **Web-subnet object** window shows.
3. Select the **NAT** tab **> Add automatic address translation rules**.
4. In the Translation method field, select **Hide > Hide Behind Gateway**.
5. In the **Install on Gateway** field, select the cluster object.
   The *automatic* NAT rules have been created.
6. Install policy on the cluster.

Step 7: Set Up the External Load Balancer in Azure

By default, the template you deploy creates an External Load Balancer, with the name frontend-lb, which faces the Internet. The External Load Balancer uses TCP health probes on port 8117 to determine the health of the CloudGuard IaaS Security Gateways.

Now let's create the load balancing rules to allow incoming connections. The load balancing rules are created in the Azure portal.

1. Go to **External Load Balancer > Frontend IP configuration**.
2. Click **Add**.

Notes:

- These ports cannot be used for forwarded traffic:
  - 80
  - 443
  - 444
  - 8082
  - 8080
  - 8117
- Do not change the health probe port.
- The Check Point cluster resource group includes an NSG associated with the frontend subnet. By default, the NSG allows all outbound and inbound traffic.
- The Load Balancer can be set up to listen on additional ports or on additional public IP addresses.

For more information, see Load Balancing with Multiple Frontends https://docs.microsoft.com/en-us/azure/load-balancer/load-balancer-standard-overview#multiple-front-ends. For an example, go to How to configure the Load Balancer to listen on additional public IP addresses ("Step 9: Configure the Load Balancer to Listen on Multiple IP Addresses in Azure" on page 28).

Step 8: Create LocalGatewayExternal in SmartConsole

Let's create the object *LocalGatewayExtenal* in SmartConsole. This object is dynamic, and contains the private cluster member IP addresses. In the next step you will learn the purpose of the object.

1. From the Object Explorer in SmartConsole, go to **New > More > Network Object > Dynamic Object**.
2. Enter: `LocalGatewayExternal`
3. Click **OK**.

Step 9: Configure the Load Balancer to Listen on Multiple IP Addresses in Azure

Configure the Load Balancer to listen on additional public IP addresses. This setup is useful if you want the gateway to secure multiple web applications, each with its own public IP address.

Let's setup the Load Balancer to listen on a second public IP address on TCP port 80, and then forward the traffic to the Check Point CloudGuard Security Gateway on TCP port 8083.

To configure the frontend pool:

1. Go to the Azure portal and select the **frontend-lb** Load Balancer.

   Note - The Load Balancer is in the resource group you created.
2. Allocate a new public IP address.
   a) Click **Frontend IP configuration** > **Add**.
   b) Select a **Name**.

      Example: <*cluster*>-app-2
   c) Select the public **IP address** you created.
   d) Click **OK**.
3. Add a load balancing rule.
   a) Click **Load balancing rules** > **Add**.
   b) Name the rule.

      Example: <*cluster*>-app-2-tcp-80
   c) In the **Frontend IP address** field, select the newly created Frontend IP address.
   d) In the **Protocol** field, select **TCP**.
   e) In the **Port** field, enter 80.
   f) In the **Backend port** field, enter 8083.
   g) In the next **Backend pool** field, select the pre-existing cluster pool.
   h) In the **Health probe** field, select the health probe created by default by the template (TCP, port 8117).
   i) In the **Session persistence** field, select **None**.
   j) Set the desired **Idle timeout**, in minutes.
   k) In the **Floating IP** field, select **Disabled**.
   l) Click **OK**.

Load Balancer Conditions

The Active member uses NAT to forward traffic that belongs to the two web applications, to the appropriate web server. NAT translation rules are defined with the Dynamic Object. For more information, see Creating LocalGatewayExternal ("Step 8: Create LocalGatewayExternal in SmartConsole" on page 27).

No	Original Packet			Translated packet			Install on
	Source	Destination	Service	Source	Destination	Service	
1	Any	LocalGatewayExternal	TCP 8081	=Original	sApp	shttps	Policy targets
2	Any	LocalGatewayExternal	TCP 8083	=Original	sWeb	shttps	Policy targets

LocalGatewayExternal represents the private IP addresses of the external interface of Member 1 and Member 2.

Configuring VPN

Let's use SmartConsole to create a Network Group object to represent the encryption domain for the cluster. To create an object for the VPN configuration, see Creating Objects in SmartConsole (on page 39). See the R80.10 Security Management Server Administration Guide http://downloads.checkpoint.com/dc/download.htm?ID=54842 for more information.

Step 1: Create a Network Group object to represent the encryption domain of the cluster and define the VPN domain:

From the Object Explorer in SmartConsole, select **New > Network Group**.

Step 2: Define your Network Group and the encryption domain of the cluster object:

1. Double-click the cluster object.
   The **Gateway Cluster Properties** window shows.
2. Go to **Network Management > VPN Domain.**
3. From the manually defined field, select the cluster object Network Group object.
   This Network Group object is the encryption domain of the cluster object.

Step 3: Define the outgoing interface:

1. From the **Gateways & Servers** tab in SmartConsole, double-click the network object.
2. From the menu at the left, click **IPsec VPN > Link Selection**.
   These three sections show in the **Gateway Cluster Properties** window.
   - **IP Selection by Remote Peer**
   - **Outgoing Route Selection**
   - **Tracking**
3. From **IP Selection by Remote Peer**, select **Always use this IP address > Main address**.
4. From **Outgoing Route Selection**, select **Source IP address settings > Manual > Selected address from topology table.**
5. Select the private cluster object VIP address.

6. Click **OK**.

Step 4: Add tunnels to the VPN.

1. From SmartConsole, go to the **Object Explorer > VPN Communities**.
2. Double-click the VPN community that the cluster participates in.
   The **VPN Community** window shows.
3. Go to **Tunnel Management > Set Permanent Tunnels**.

Additional Information

In This Section:

Testing and Troubleshooting

You can use the APIs to retrieve information about the cluster resource group. Use these commands on each cluster member to confirm that the cluster is operating correctly.

`cphaprob -a`

`cphaprob state`

Example:

```
Expert@HostName:0]# cphaprob state
Cluster Mode:   High Availability (Active Up) with IGMP Membership
Number     Unique Address  Assigned Load   State
1 (local)   10.0.1.10       0%              Active
2           10.0.1.20       100%           Standby
```

Use the cluster configuration test script on each cluster member to confirm the member is configured correctly.

Run the script from Expert Mode with this command (do not change the syntax):

```
# $FWDIR/scripts/azure_ha_test.py
```

The script verifies:

- The configuration file is defined in `$FWDIR/conf/azure-ha.json.` This file is created by the ARM template.
- A Primary DNS server is configured and is working.
- The machine is set up as a cluster member.
- IP forwarding is enabled on all network interfaces of the cluster member.
- You can log into Azure using the Azure credentials in `$FWDIR/conf/azure-ha.json`

If all tests were successful, this shows: `All tests were successful!` Otherwise, an error message is displayed with information to troubleshoot the problem.

A list of common configuration errors:

Message	Recommendation
`The attribute (ATTRIBUTE) is missing in the configuration`	
`Primary DNS server is not configured` `Failed to resolve (host)`	The cluster member is not configured with a DNS server.
`Failed in DNS resolving test`	Confirm that DNS resolution on the cluster member works.
`You do not seem to have a valid cluster configuration`	Make sure that the member configuration on the Check Point Security Management Server is complete and that the Security Policy is installed.
`IP forwarding is not enabled on Interface (Interface-name)`	Use PowerShell to enable IP forwarding on all the network interfaces of the cluster member.
`failed to read configuration file: /opt/CPsuite-R80/fw1/conf/azure-ha.json`	The Azure cluster member configuration is not up to date, or written correctly.
`Testing credentials`	Failed to log in with the credentials provided. See the exception text to understand why.
`Testing authorization` `(Exception)`	Make sure the Azure Active Directory service account you created is designated as a Contributor to the cluster resource group.

Simulate a cluster failover. For example, shut down the internal interface of the Active cluster member.

From Expert Mode enter:

```
# ip link set dev eth1 down
```

In a few seconds the second member has to report itself as the Active cluster member.

If you are experiencing issues:

- Make sure you have set up an Azure Active Directory Service Account. The service has to have:
  - Contributor privileges to the resource group
  - At least minimum privileges on the member deployment resources. See *Minimum Roles* in Changing Template Components (on page 38).
- To make the networking changes automatically, the members have to communicate with Azure. This requires HTTPS connections over TCP/443 to the Azure end points.
- Make sure the Security Policy that is installed on the Security Gateway allows this type of communication.

Using the Azure High Availability Daemon

The cluster solution in Azure uses the daemon to make API calls to Azure when a member failover takes place. This daemon uses a configuration file, `$FWDIR/conf/azure-ha.json` located on each cluster member.

When you deploy the solution above from the template supplied, this file is created automatically. The configuration file is in json format and contains these attributes:

Attribute name	Type	Value
`debug`	Boolean	`true` or `false`
`subscriptionId`	String	Subscription ID.
`location`	String	Resource group location.
`environment`	String	Name of the environment.
`resourceGroup`	String	Resource group name.
`credentials`	String	IAM. Indicates using automatic credentials on the member Virtual Machine.
`proxy`	String	Name of the proxy.
`virtualNetwork`	String	Name of the Virtual Network.
`clusterName`	String	Name of the cluster.
`templateName`	String	Name of the template.
`tenantId`	String	ID of the tenant.

Note - If you use your own service principal, the `credentials` attribute contains:

- Your Client-id
- Your Client-secret
- Grant type: client-credentials
- Your tenant id

You can confirm that the daemon in charge of communicating with Azure is running on each cluster member. From Expert Mode, enter:

```
# cpwd_admin list | grep -E "PID|AZURE_HAD"
```

The output should look like this:

```
APP         PID    STAT  #START  START_TIME              MON  COMMAND
AZURE_HAD   3663   E     1       [12:58:48] 15/1/2016    N    python
/opt/CPsuite-R77/fw1/scripts/azure_had.py
```

Notes:

- The script appears in the output
  - The STAT column should show **E** (executing)
  - The #START column should show **1** (the number of times this script was started by the Check Point WatchDog)

To troubleshoot issues related to this daemon, generate debugging printouts. From Expert Mode:

- To enable debug printouts

```
# azure-ha-conf --debug --force
```

- To disable debug printouts

```
# azure-ha-conf --no-debug --force
```

The debug output is written to `$FWDIR/log/azure_had.elg*`

Using a Different Azure Cloud Environment

If you want to deploy your cluster in an environment other than the standard Azure environment, make sure to edit this file: `$FWDIR/conf/azure-ha.json`

```
{
...
    "environment": "[Azure-cloud-environment]",
...
}
```

The Azure-Cloud-Environment has to be one of these:

- Azure Cloud (the default global cloud environment)
- Azure China Cloud
- Azure US Government
- Azure German Cloud

From Expert Mode, enter:

```
# azure-ha-conf --environment <Azure-cloud-environment> --force
```

Validate the file syntax. From Expert Mode run:

```
# python -m json.tool $FWDIR/conf/azure-ha.json
```

Apply the changes.

```
# $FWDIR/scripts/azure_ha_cli.py reconf
```

Note -If you deploy in the default global cloud environment, you can omit this attribute.

Important note about the service principal:

If you use any of these different environments, you have to create your own service principal. No default service principal is created.

Working with a Proxy

In some deployments, you can only access the Internet through a web proxy. To allow the cluster member to make API calls to Azure through the proxy, edit this file, `$FWDIR/conf/azure-ha.json`, and add the following attribute:

```
{
...
     "proxy": "http://[proxy-name]:[proxy-port]",
...
}
```

- *proxy-name* is the host name or IP address of the web proxy
- *proxy-port* is the port the proxy port

Example:

```
{
...
     "proxy": "http://proxy.example.com:8080",
...
}
```

Note - The URL scheme has to be HTTP and not HTTPS.

Go to Expert Mode to run these commands:

- Change this setting:

  ```
  # azure-ha-conf --proxy 'http://[proxy-name]:[proxy-port]' --force
  ```

- Confirm that the file you modified has no syntax errors:

  ```
  # python -m json.tool $FWDIR/conf/azure-ha.json
  ```

- Apply the changes:

  ```
  # $FWDIR/scripts/azure_ha_cli.py reconf
  ```

Changing Template Components

The R80.10 cluster public IP address has to be in the same resource group as the cluster member.

These resources can be in any resource group:

- Virtual Network
- Network interfaces
- Route tables
- Storage account

Note - Make sure the resources Virtual Network and External Network Interfaces use the same automatic service principal with the same permissions.

Naming Constraints

- Cluster members in Azure have to match the cluster member names with a suffix of '1' and '2'.
- The IP address of the cluster has to match the configuration file.
- By default it should match the cluster name.

Permissions

It is possible to assign service principal permissions to specific Azure resources. See sk116585 http://supportcontent.checkpoint.com/solutions?id=sk116585 for information on how to find the image version.

To allow the cluster to update the necessary Azure resources on failover, the service principal has to be assigned at least the following roles on these resources or on their respective resource group.

Resource Type	Role
Any public IP address attached to the External Load Balancer	Virtual Machine contributor
Public Load Balancer	Network contributor
CloudGuard Virtual Machines	Reader
Cluster public IP address	Network contributor
Public IP address of each cluster member	Virtual Machine contributor
Virtual Network	Virtual Machine contributor
The external network interfaces (`eth0`) used by the cluster member	Virtual Machine contributor

Creating Objects in SmartConsole

To create the Host object:

1. From the menu on the right, click **New > Host**.
   The **New Host** window shows.
2. In the **Machine** field, enter the private IP address of the machine.

To create the internal subnet object:

1. From the menu on the right, click **New > Network**.
   The **New Network** window shows.
2. Enter the **Object Name** (specifically the subnet name).
3. Enter the **Network address** and **Net mask**.

To create a port service object:

1. From the menu on the right, click **New > More > Service**.
2. Select your TCP/UDP service.
3. Enter the **Object name**.
4. In the **Enter Object Comment** field, enter the port name.
5. In the **General** field, select your **Protocol**.
6. In the **Match By** field, select the **Port** number.
7. Click **OK**.

To create a Network Group:

1. From the menu on the right, click **New > Network Group**.
   The **New Network Group** window shows.
2. Select your internal subnets.
3. Click **OK**.

For more information, see the R80.10 Security Management Server Administration Guide http://downloads.checkpoint.com/dc/download.htm?ID=54842.

Known Limitations

- For outbound and VPN traffic, you cannot delete or disable the public IP address of cluster members.
- The feature is only available in Azure Resource Manager deployments. It is not supported with Azure Service Manager (also known as classic) deployments.
- Only two members per cluster are supported.
- Only High Availability Mode (Active/Standby) is supported. Load Sharing Mode is not supported.
- VRRP is not supported.
- Only the Active member can reach services from the cluster via VPN. The Standby member can reach those services only when it becomes the Active member.
- When you use the standard Internal Load Balancer it does not support Stateful failover.
- Managed service identity for Virtual Machines is only available in the Azure Cloud environment. Other environments require a manual service identity management.
- Standard Load Balancers and High Availability ports are not available on the Azure Government Cloud environment

Related Solutions

- sk109360 http://supportcontent.checkpoint.com/solutions?id=sk109360. Check Point Reference Architecture for Azure
- sk113583 http://supportcontent.checkpoint.com/solutions?id=sk113583. How to add a network interface to a Check Point Security Gateway in Azure
- sk113476 http://supportcontent.checkpoint.com/solutions?id=sk113476. Azure Virtual Network peering