



FortiADC - Deploying High Availability on Azure

Version 7.2.0

FORTINET DOCUMENT LIBRARY

<https://docs.fortinet.com>

FORTINET VIDEO GUIDE

<https://video.fortinet.com>

FORTINET BLOG

<https://blog.fortinet.com>

CUSTOMER SERVICE & SUPPORT

<https://support.fortinet.com>

FORTINET TRAINING & CERTIFICATION PROGRAM

<https://www.fortinet.com/training-certification>

FORTINET TRAINING INSTITUTE

<https://training.fortinet.com>

FORTIGUARD CENTER

<https://www.fortiguard.com>

END USER LICENSE AGREEMENT

<https://www.fortinet.com/doc/legal/EULA.pdf>

FEEDBACK

Email: techdoc@fortinet.com



February 3, 2023

FortiADC 7.2.0 Deploying High Availability on Azure

01-544-677187-20230203

TABLE OF CONTENTS

Change Log	4
Overview of FortiADC HA pairing on the Azure platform	5
Checking the prerequisites	6
VRRP HA using API call to Azure	7
VRRP HA using Azure Load Balancers	9
Why use the Azure Load Balancer?	11
Configuring the FortiADC-VMs to use Azure Load Balancer	11
Deploying VRRP HA using Azure load balancers via the ARM template	13
Creating an Azure storage account	13
Uploading license files to Azure storage container	14
Creating an Azure Active Directory application	15
Getting the subscription ID and tenant ID	20
Deploying FortiADC HA resources from the ARM template	21
Connecting to the FortiADC GUI and CLI	26
FortiADC Virtual Server with HA VRRP mode using Azure Load Balancer	28
Example: FortiADC L7 Virtual Server with HA VRRP mode using Azure Load Balancer	
Topology	30
Example: FortiADC L4 Virtual Server with HA VRRP mode using Azure Load Balancer	
Topology	37
FortiADC Virtual Server with traffic group in HA VRRP using Azure Load Balancer	42
Virtual Server NAT Pool and Firewall NAT SNAT configuration in HA mode	48
Debugging Azure API call in HA mode	51

Change Log

Date	Change Description
May 27, 2022	Initial release

Overview of FortiADC HA pairing on the Azure platform

Microsoft Azure supports FortiADC HA (High Availability) in Active-Active-VRRP mode. Azure can only support the VRRP HA mode because other modes that require access to Layer 2 are inaccessible in Public Cloud environments. For more information on FortiADC HA modes or the Azure Virtual Network, see the [FortiADC Handbook on High Availability Deployments](#) and the [Azure Virtual Network FAQ](#).

There are 2 methods to deploy the FortiADC VRRP HA on Azure:

1. Deploy the VRRP HA using an API call to Azure
2. Deploy the VRRP HA using Azure Load Balancers

Introduced in the FortiADC 6.2.0 release, the method of using Azure Load Balancers to pair FortiADC HA on the Azure platform was developed to resolve the issues created by the Azure API call. Previously, VRRP HA on Azure used the API call to Azure to migrate the public IPs associated with the FortiADC-VM on the Azure backend in the event of HA failover. The IP migration process may take several minutes during the HA failover, causing business traffic to break until the migration is complete. With using Azure Load Balancers, the downtime for the FortiADC member may only be seconds during the HA failover because it does not require IP migration.

For more details on the two HA deployment methods, see [VRRP HA using API call to Azure on page 7](#) and [VRRP HA using Azure Load Balancers on page 9](#).

Checking the prerequisites

You need to meet the following prerequisites to deploy HA on Azure:

- A Microsoft Azure account that can be used to log into the Azure portal. If you do not have an Azure account, please follow the instructions on the [Microsoft Azure website](#) on how to obtain one.
- A Microsoft Azure subscription that allows you to purchase the FortiADC-VM and launch in a desired location.
- If you choose a BYOL image type, then you will need valid FortiADC licenses (for a minimum of two CPUs) for all HA members.

VRRP HA using API call to Azure

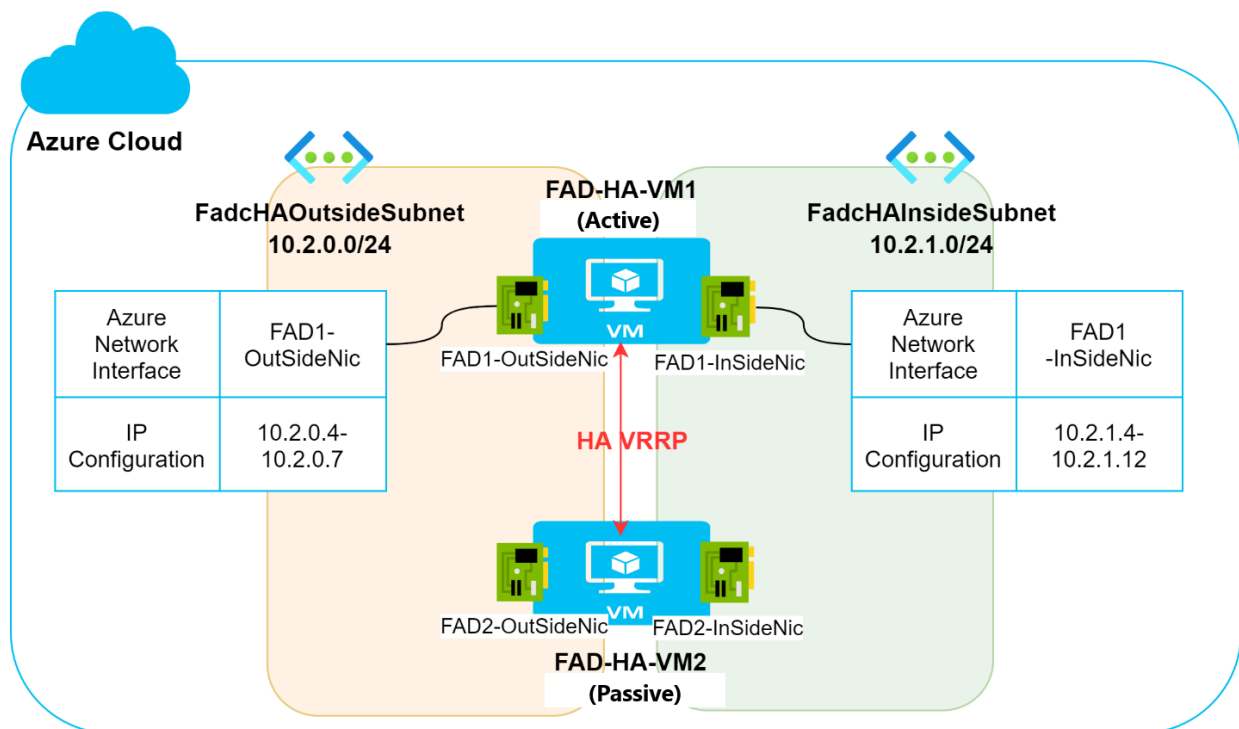
In this VRRP HA deployment mode that uses an API call to Azure, each FortiADC network interface on Azure maintains its own IP configuration table. You can add secondary IPs in the IP configuration table and use the IPs in the FortiADC configuration, such as virtual server IPs.



Prior to the FortiADC 6.2.0 release, deploying VRRP HA on Azure using an API call was the standard deployment mode. In version 6.2.0, deploying VRRP HA using Azure Load Balancers was introduced to mitigate the issues caused by the Azure API call deployment mode. We recommend deploying VRRP HA on Azure using Azure Load Balancers as the preferred deployment mode.

In the Figure 1 example below, there are two (2) FortiADC-VMs in HA VRRP mode that share the same configuration for Virtual Server, Virtual Server NAT Pool, Firewall NAT SNAT, Firewall 1-1 NAT, and the network interface floating IP. On the Azure platform side, the corresponding IP configuration is only attached to the active FortiADC-VM, which is the FAD-HA-VM1 in this example.

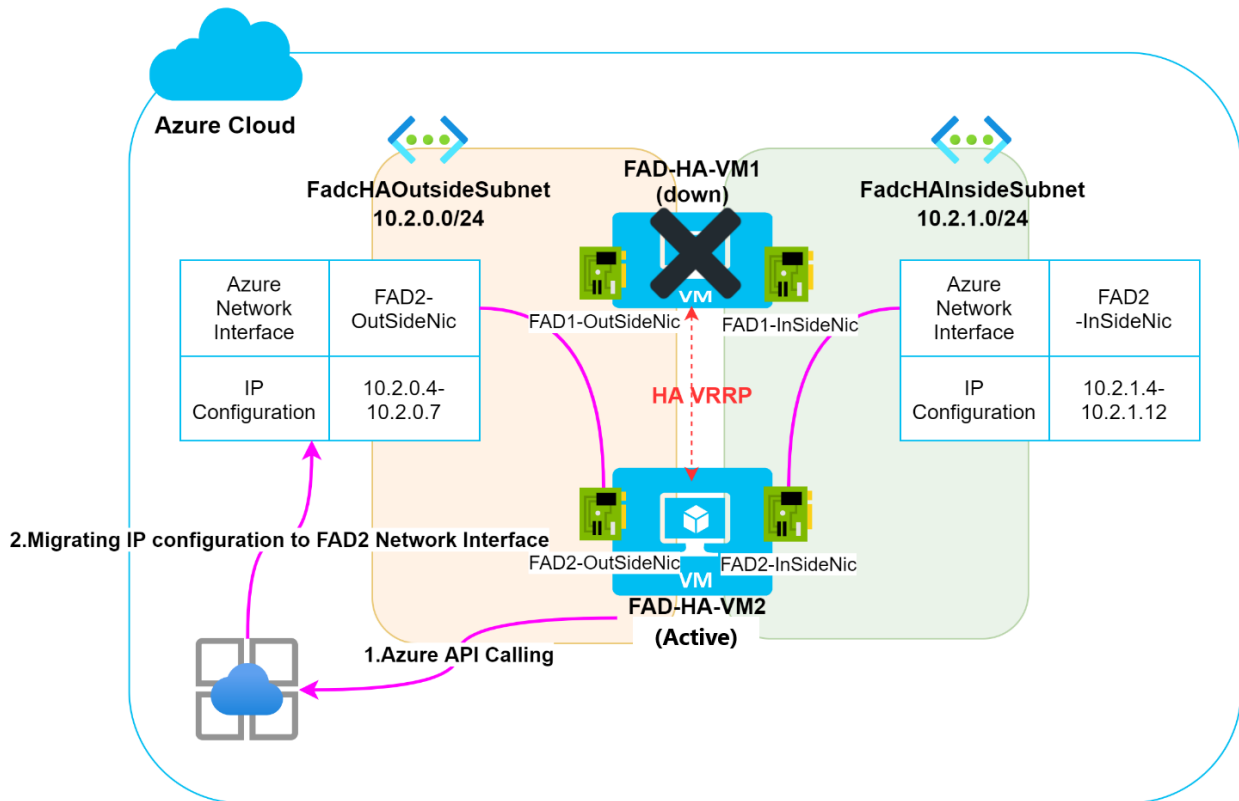
Figure 1 FortiADC HA pair on Azure platform



Shared FAD Configuration	Virtual Server	IPPool(Virtual Server NAT Pool)	Firewall NAT SNAT	Firewall 1-1 NAT	Interface FloatingIP
IP Address	10.2.0.4-10.2.0.6	10.2.1.4-10.2.1.10	10.2.0.7	10.2.1.11	10.2.1.12

In the event of HA failover, the FAD-HA-VM2 becomes the active node. This requires the IP configuration to be migrated from FAD-HA-VM1 to FAD-HA-VM2 in the Azure backend. More specifically, this disassociates the related IP configuration on the FAD1-OutSideNic / FAD1-InSideNic and associates them on the FAD2-OutSideNic / FAD2-OutSideNic. This IP migration process is executed using an Azure API call, as illustrated in the Figure 2 example below.

Figure 2 IP Migration between the Azure network interface when HA failover occurs



The major downside to the IP migration process is the significant amount of time required to propagate the change into the Azure environment. This may cause the business traffic to break for several minutes during the HA failover. Other critical issues may also occur, such as losing IP configuration on the Azure side during the failover.

For these reasons, FortiADC has developed a new deployment mode to avoid the issues created by the Azure API call. For more details on the new deployment mode, see [VRRP HA using Azure Load Balancers on page 9](#).



For the Firewall 1-1 NAT configuration, the Azure API calling method is still required to migrate the IP during HA failover.

VRRP HA using Azure Load Balancers

To avoid the issues caused by the Azure API call HA deployment mode, FortiADC has developed a new design that incorporates the use of external and internal Azure Load Balancers. The objective of this design is to eliminate the need for IP migration by making the IP configurations that are affected during an HA failover independent on each FortiADC. For more information, see [Deploying VRRP HA using Azure load balancers via the ARM template on page 13](#).

In the event of an HA failover, there are 5 configurations that will be affected:

1. Virtual Server IP
2. IP used in the VS NAT pool
3. IP used in the Firewall NAT SNAT
4. IP used as the network interface floating IP
5. IP used in the Firewall 1-1 NAT

To avoid IP migration, the IP configuration needs to be independent on each FortiADC. This means that these FortiADC IPs should not be synchronized even when HA is active. Each FortiADC should maintain its own IP configuration (for all 5 configurations listed above), both on the FortiADC and Azure side as shown in Figure 3 below.

To achieve this, the FortiADC units are configured to maintain their own IP configuration. However, this is only achievable for 4 of the total 5 configurations listed above; the 5th configuration, the IP used for the Firewall 1-1 NAT, would still require using the Azure API call method for HA failover.



For the Firewall 1-1 NAT configuration, the Azure API calling method is required to migrate the IP during HA failover.

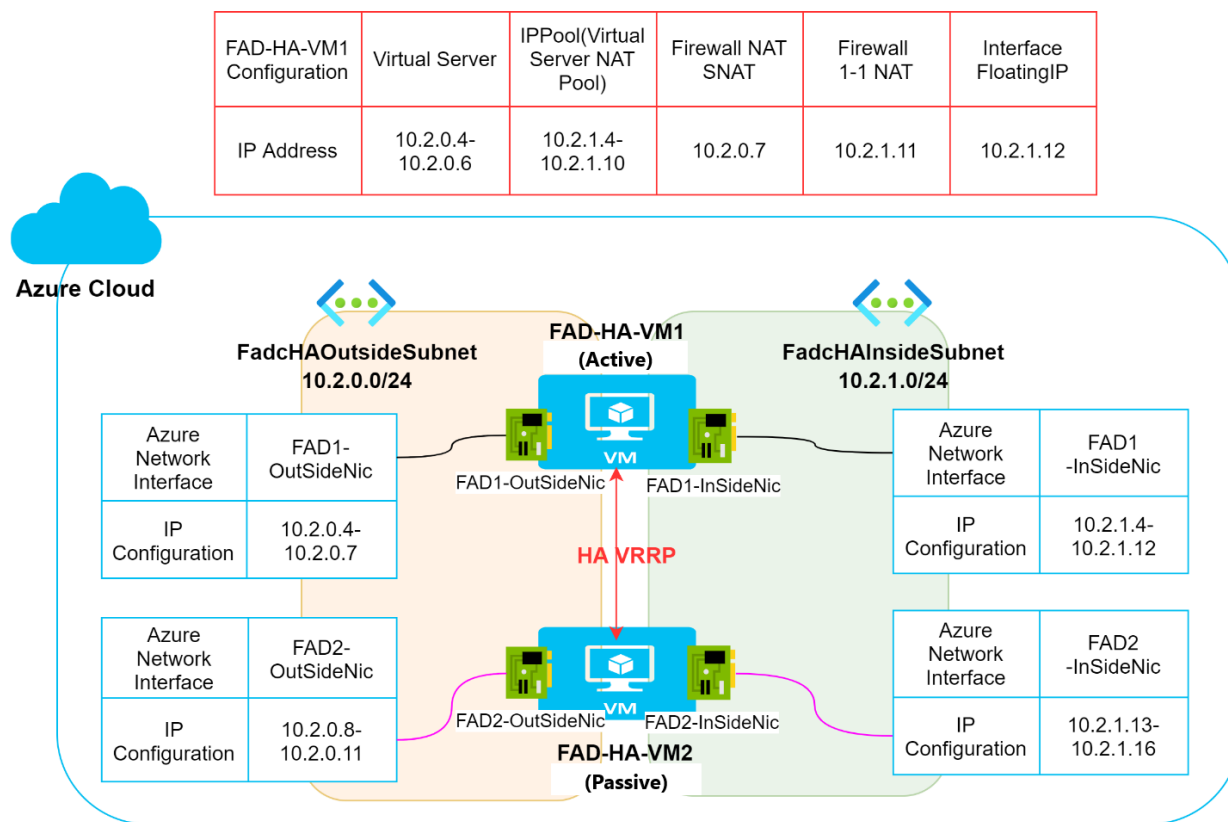


In FortiADC 6.2.0/6.2.1, the following IP configurations are not available in the VRRP HA in Azure:

- IP used in the VS NAT pool
- IP used in the Firewall NAT SNAT

In FortiADC 6.2.2, these IP configurations are available in the VRRP HA in Azure. For more information, see [Virtual Server NAT Pool and Firewall NAT SNAT configuration in HA mode on page 48](#).

Figure 3 Independent IP configuration on both FortiADC side and Azure side



FAD-HA-VM2 Configuration	Virtual Server	IPPool(Virtual Server NAT Pool)	Firewall NAT SNAT	Firewall 1-1 NAT	Interface FloatingIP
IP Address	10.2.0.8-10.2.0.10	10.2.1.13-10.2.1.14	10.2.0.11	10.2.1.15	10.2.1.16

For more details on configuring the FortiADC for this new deployment mode, see [Configuring the FortiADC-VMs to use Azure Load Balancer on page 11](#).



If you are using a virtual server IP and/or an interface floating IP, you will need to set up an Azure Load Balancer to achieve a single access endpoint.



For FortiADC version 6.2.0 or above, the VRRP HA deployment mode using Azure API call is no longer supported for virtual server IP and interface floating IP. An Azure Load Balancer is required for deploying HA for these IP configurations.

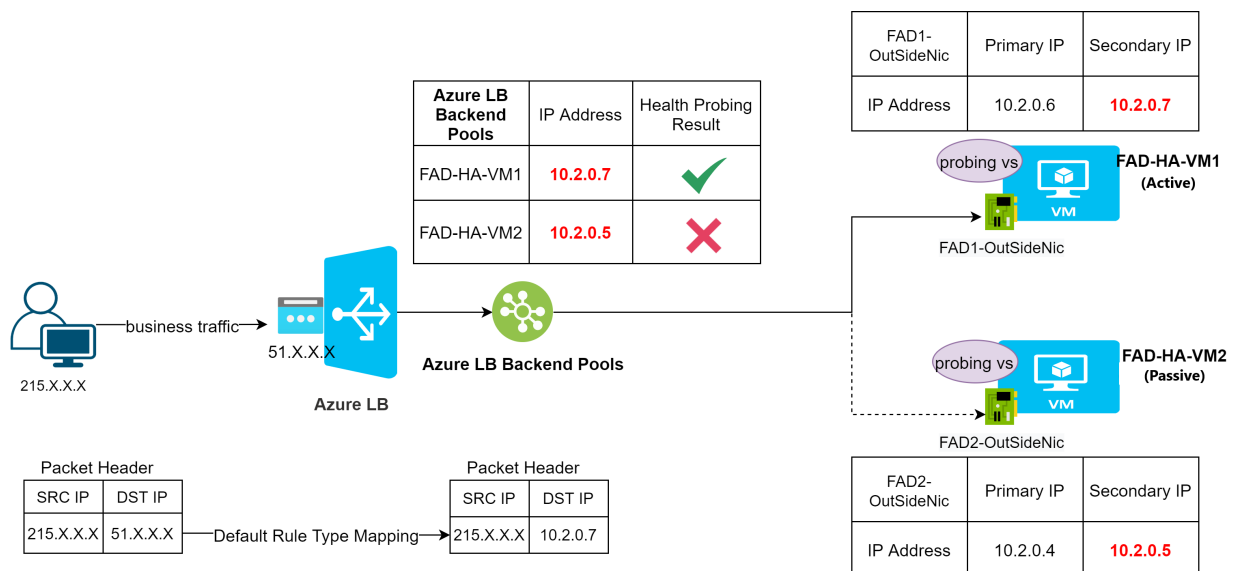
Why use the Azure Load Balancer?

The independent configuration of the virtual server IP on both FortiADCs would now be using Azure Load Balancers to provide the single access point for customers in the event of HA failover. The Azure Load Balancer publishes the IP address used by customers to provide the single point of contact for clients. Azure Load Balancer (ALB) distributes inbound flows that arrive at the load balancer's front end to backend pool instances. For deploying the FortiADC VRRP HA, the ALB distributes the inbound traffic to the FortiADC units (the backend pool instances). These flows are determined according to configured load-balancing rules and health probes.

ALB provides flexibility in defining the load-balancing rules. The function of the load-balancing rule is to declare how an address and port on the front end would be mapped to the destination address and port on the backend.

In FortiADC 6.2.0, the default rule type mapping of ALB is supported, using the secondary IP of the FortiADC in the ALB backend pool configuration. As shown in Figure 4 below, with the default rule type, Azure exposes a traditional load balancing IP address scheme for ease of use (such as the VM instances' IP). In the Figure 4 example, when receiving business traffic, ALB changes the packet destination IP from 51.x.x.x to 10.2.0.7, which is the secondary IP of the FAD1-OutSideNic attached to the HA active node, FAD-HA-VM1.

Figure 4 Azure Load Balancer with FortiADC HA pair



Azure Load Balancer uses health probes to monitor the FortiADCs in the backend pool by periodically sending the health check packet to all FortiADCs in the backend pool. The probing virtual server configured on FortiADC will respond to the health check packet from the ALB. Under known conditions in HA VRRP mode, only the probing virtual server on the active FortiADC will respond to the packet. Based on the health check result, ALB can direct the business traffic to the correct (active) FortiADC.

Configuring the FortiADC-VMs to use Azure Load Balancer

In FortiADC 6.2.0, the new configuration object "Azure LB Backend" was introduced. This can be found via the GUI in **System > Azure LB Backend**. When adding a new entry in the Azure LB Backend configuration on one node, the entry will be synchronized to the other node but excluding the IP address of the unit. On the second node, you will see the entry as 0.0.0.0. You will need to change this value to the Azure LB Backend IP of the

second node. The Azure LB Backend configuration is designed to be used in the virtual server configuration screen to prevent configuration mistakes and re-using the same IP address.



As the IP address in the Azure LB Backend configuration object would not be synchronized in HA mode, you must configure it on both the FortiADCs in the cluster.

This feature is only available in the FortiADC Azure based images.

Using the Figure 4 example, the FAD-HA-VM1 configuration should look like this:

```
FAD-HA-vm1 # show system azure-lb-backend-ip
config system azure-lb-backend-ip
    edit "FADHaLBBackendAddrPool"
        set ip 10.2.0.7
    end
```

And the FAD-HA-VM2 configuration should look like this:

```
FAD-HA-vm2 # show system azure-lb-backend-ip
config system azure-lb-backend-ip
    edit "FADHaLBBackendAddrPool"
        set ip 10.2.0.5
    end
```

Settings	Guidelines
Name	Azure LB Backend Pool Name. Valid characters are A-Z, a-z, 0-9, _, and -. No spaces. After you initially save the configuration, you cannot edit the name.
IP	IP address used in the Azure LB Backend Pool. Note: The IP address would not be synchronized in HA mode.

The Azure LB Backend configuration object can be used in the virtual server configuration to reference the Azure LB Backend IP as the virtual server IP.

```
FAD-HA-vm1 # show load-balance virtual-server probing-vs
config load-balance virtual-server
    edit "l7-vs"
        set type l7-load-balance
        set use-azure-lb-backend-ip enable
        set azure-lb-backend FADHaLBBackendAddrPool
        set interface port1
        set port 8080
        set load-balance-profile LB_PROF_HTTP
        set load-balance-method LB_METHOD_ROUND_ROBIN
        set load-balance-pool pool
        set traffic-group default
    next
end
```

Settings	Guidelines
use-azure-lb-backend-ip	Enable or disable the virtual server to reference Azure LB Backend IP as the virtual server IP.
azure-lb-backend	Select the Azure LB Backend IP Configuration used as the virtual server IP.

The Azure LB Backend IP configuration is an independent configuration on all FortiADC nodes. The virtual server references the Azure LB Backend IP as the virtual Server IP. In this way, the virtual server IP is no longer a shared configuration for FAD-HA-VM1 and FAD-HA-VM2.

Therefore, when an HA failover occurs, the traffic will be directed to the new active FortiADC by the ALB.

For more information, see [Deploying VRRP HA using Azure load balancers via the ARM template on page 13](#).

Deploying VRRP HA using Azure load balancers via the ARM template

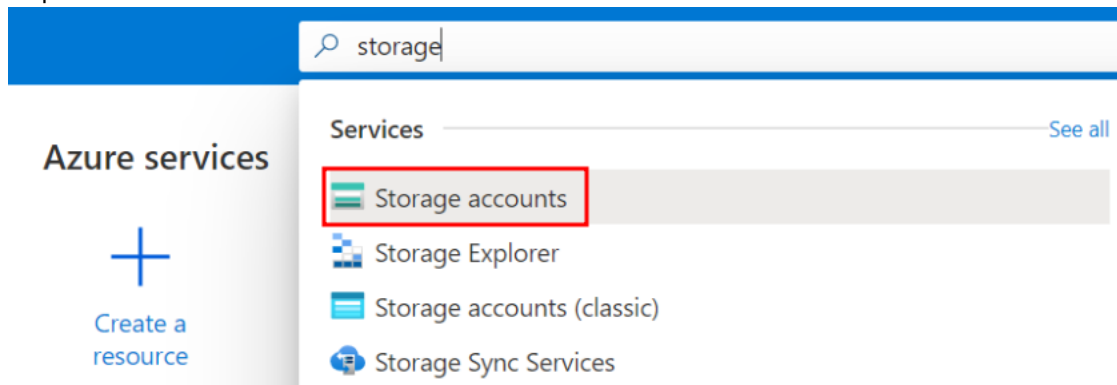
Follow the workflow below to deploy VRRP HA using Azure load balancers via the ARM (Azure Resource Manager) template.

1. [Creating an Azure storage account](#)
2. [Uploading license files to Azure storage container](#)
3. [Creating an Azure Active Directory application](#)
4. [Getting the subscription ID and tenant ID](#)
5. [Deploying FortiADC HA resources from the ARM template](#)
6. [Connecting to the FortiADC GUI and CLI](#)

Creating an Azure storage account

The Azure storage account is used for storing the boot diagnostics for the serial console usage of the FortiADC instance. If you are using a FortiADC BYOL image, you will also require an Azure storage account to upload the license files.

1. Go to the **Storage accounts** service to create a storage account or use an existing storage account. Refer to this [Azure documentation](#) for details. Take note of the **Storage account name**. It will be used in later steps.

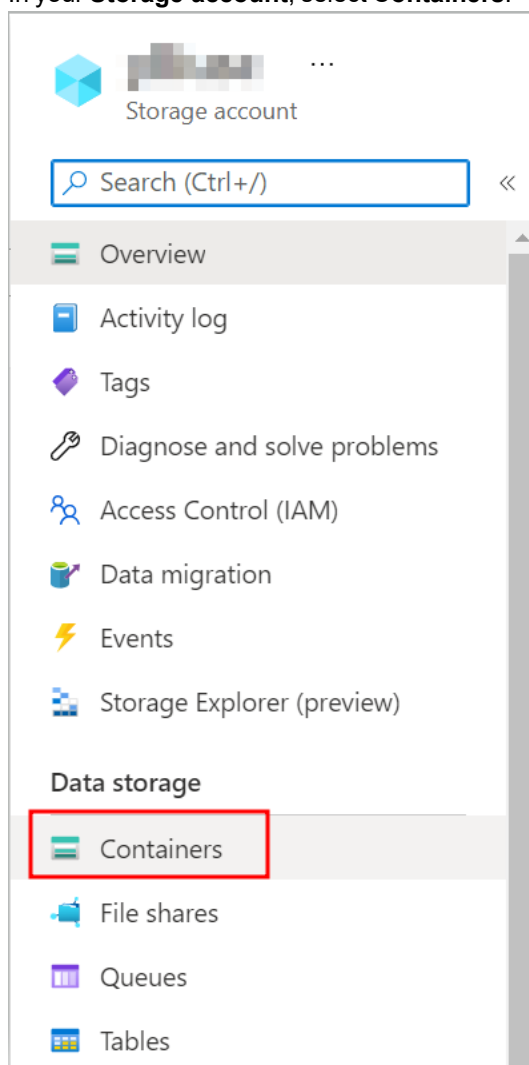


2. The **Subscription** and **Resource Group** of the storage account should be the same as the HA resources to be deployed.
3. For the **Networking** settings, it is recommended to select **Public endpoint (all networks)**. If you prefer to restrict access by allowing only selected networks or private endpoints, make sure to plan the network accordingly so that the FortiADC-VMs can successfully obtain license files from this storage account.

Uploading license files to Azure storage container

If the FortiADC-VM image type to be deployed is BYOL, prepare the license files and store the files to an Azure Storage container. If you want to deploy the PAYG image, skip this step and go directly to the next step [Creating an Azure Active Directory application on page 15](#).

1. In your **Storage account**, select **Containers**.



2. Click **+Container**.

3. Enter a name for this Container and set the **Public access level** to **blob** or **container** that is enabled for read access. Take note of this Container name. It will be used in later steps.


New container

×

Name *

Public access level ⓘ

Blob (anonymous read access for blobs only) ▼

 Blobs within the container can be read by anonymous request, but container data is not available. Anonymous clients cannot enumerate the blobs within the container.

▼ Advanced

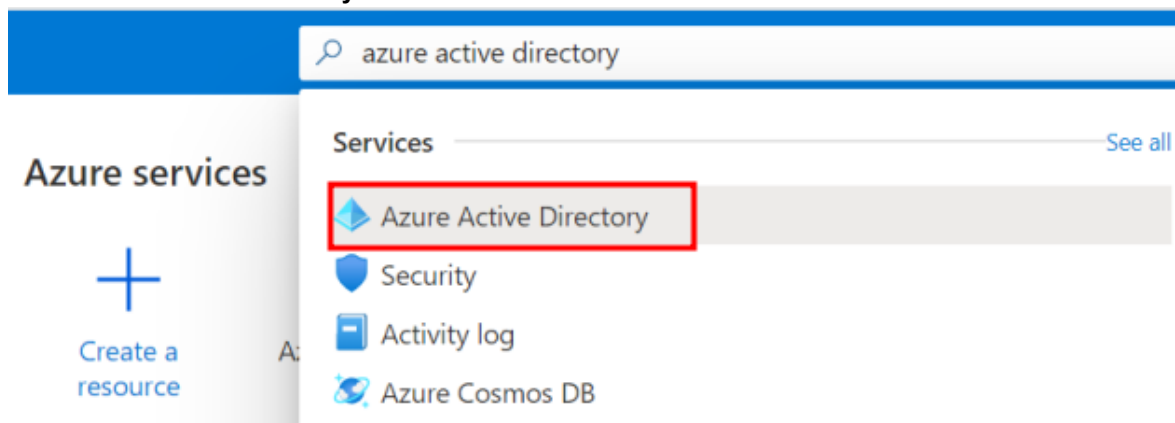
4. Click **Create**.
5. In the **Containers** list, select the container you have just created.
6. Click **Upload**.
7. Select the license files from your local directory, then click **Upload**. The number of license files should be the same as the number of FortiADC-VMs you want to deploy.

If you upload license files to an existing container, be sure to delete the used license files in the container if there are any. Otherwise, the FortiADC-VMs to be deployed in the next steps may mistakenly fetch these used license files.

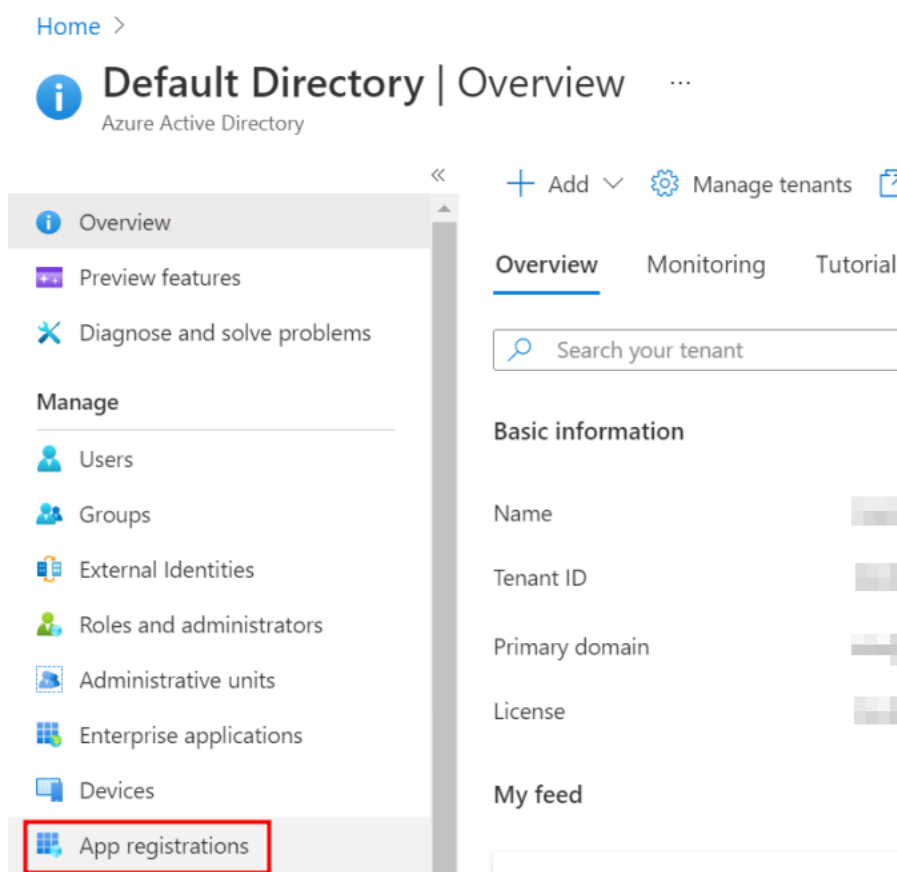
Creating an Azure Active Directory application

Create an Azure Active Directory application to authorize the function application to access the Azure resources.

1. Sign in to your Azure Account through the Azure portal.
2. Select **Azure Active Directory**.



3. Select **App registrations**.



4. Click **New registration**.
5. Provide a name and URL for the application. For more information, refer to this [Azure documentation](#). After setting the values, click **Create**. Take note of the application name. You will use it later.

Assigning the application to the owner role of the subscription

You must assign the application to the owner role of the subscription where the HA resources will be deployed to ensure it can have the privilege to authorize the HA resources in the subscription.

1. Go to **Subscriptions**, choose the subscription to assign the application to. This is the subscription in which all the resources in the HA cluster will be deployed.
2. Go to **Access control (IAM)**, and select the **Role assignments** tab.

The screenshot shows the Azure portal interface. On the left, the 'Subscriptions' page is visible, showing a list of subscriptions with filters for 'My role' (8 selected) and 'Status' (3 selected). The main area displays the 'Access control (IAM)' page for a specific subscription. The left sidebar lists various management tools: Overview, Activity log, Access control (IAM) (selected), Tags, Diagnose and solve problems, Security, Events, Cost Management (Cost analysis, Cost alerts, Budgets, Advisor recommendations). The right pane shows the 'Access control (IAM)' page with tabs for 'Check access', 'Role assignments', 'Roles', and 'Deny assignments'. The 'Check access' tab is active, showing 'My access' and 'Check access' sections. The 'Add' button is visible in the top right of the 'Access control (IAM)' page.

3. Click **Add > Add role assignment**.

The screenshot shows the 'Add role assignment' dropdown menu in the Azure portal. The menu is open, showing options: 'Add role assignment', 'Add co-administrator', and 'Add custom role'. The 'Add role assignment' option is highlighted. The background shows the 'Access control (IAM)' page with the 'Role assignments' tab selected. The 'Add' button is visible in the top right of the 'Access control (IAM)' page.

4. On the **Roles** tab, select **Owner** and click **Next**.

- On the **Members** tab, select **User group, or service principal**. In the **Members** field, select the application name you want to assign the Owner role to.

Home > Subscriptions > [Subscription Name]

Add role assignment

Got feedback?

Role **Members** Review + assign

Assign access to

☒ User, group, or service principal

☐ Managed identity

Members

[+ Select members](#)

Name	Object ID	Type
No members selected		

Description

Optional

Review + assign Previous Next

- Click **Review + assign**.

Getting the Application ID and authentication key

- Go to **Azure Active Directory > App registrations**.
- Select the function App you have just created. Take note of the **Application ID**. It will be used in later steps.

Home > App registrations >

example-app

Search (Ctrl+/) << Delete Endpoints Preview features

Overview Quickstart Integration assistant

Manage Branding Authentication

Essentials

Display name : example-app

Application (client) ID : [Application ID]

Object ID : [Object ID]

Directory (tenant) ID : [Directory ID]

Supported account types : My organization only

Copy to clipboard

3. Select **Certificates & secrets** and select the **Client secrets** tab. Click **New client secret** to create a new secret.

Home > App registrations > example-app

example-app | Certificates & secrets

Search (Ctrl+/) Got feedback?

Overview Quickstart Integration assistant

Manage

Branding Authentication

Certificates & secrets

Token configuration API permissions Expose an API App roles

Credentials enable confidential applications to identify themselves to the authentication service when receiving tokens at a web addressable location (using an HTTPS scheme). For a higher level of assurance, we recommend using a certificate (instead of a client secret) as a credential.

Application registration certificates, secrets and federated credentials can be found in the tabs below.

Certificates (0) **Client secrets (0)** Federated credentials (0)

A secret string that the application uses to prove its identity when requesting a token. Also can be referred to as application password.

+ New client secret

Description	Expires	Value	Secret ID
No client secrets have been created for this application.			

4. Enter a description for the secret, select when it would expire.

Home > App registrations > example-app

example-app | Certificates & secrets

Search (Ctrl+/) Got feedback?

Overview Quickstart Integration assistant

Manage

Branding Authentication

Certificates & secrets

Token configuration API permissions Expose an API App roles

Credentials enable confidential applications to identify themselves to the authentication service when receiving tokens at a web addressable location (using an HTTPS scheme). For a higher level of assurance, we recommend using a certificate (instead of a client secret) as a credential.

Application registration certificates, secrets and federated credentials can be found in the tabs below.

Certificates (0) **Client secrets (1)** Federated credentials (0)

A secret string that the application uses to prove its identity when requesting a token. Also can be referred to as application password.

+ New client secret

Description	Expires	Value	Secret ID
demo key	11/16/2023		

Add a client secret

Description Enter a description for this client secret

Expires Recommended: 6 months

5. Click **Add**.
6. Take note of the **Value** of the secret. It will be used in later steps.
Note: It is recommended that you copy this value as it cannot be retrieved later. This key value and the application ID is required to sign in as the application. Store the key value where your application can retrieve it.

Home > App registrations > example-app

example-app | Certificates & secrets

Search (Ctrl+/) Got feedback?

Overview Quickstart Integration assistant

Manage

Branding Authentication

Certificates & secrets

Token configuration API permissions Expose an API App roles

Credentials enable confidential applications to identify themselves to the authentication service when receiving tokens at a web addressable location (using an HTTPS scheme). For a higher level of assurance, we recommend using a certificate (instead of a client secret) as a credential.

Application registration certificates, secrets and federated credentials can be found in the tabs below.

Certificates (0) **Client secrets (1)** Federated credentials (0)

A secret string that the application uses to prove its identity when requesting a token. Also can be referred to as application password.

+ New client secret

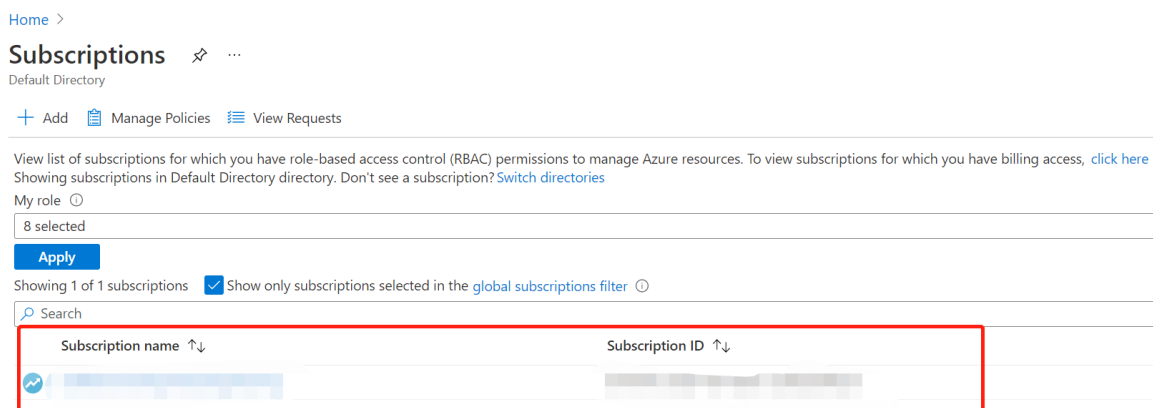
Description	Expires	Value	Secret ID
azure-ha	4/29/2022		

Getting the subscription ID and tenant ID

Follow the steps below to obtain the values for the ARM template parameters. You will use these values later when deploying the ARM template.

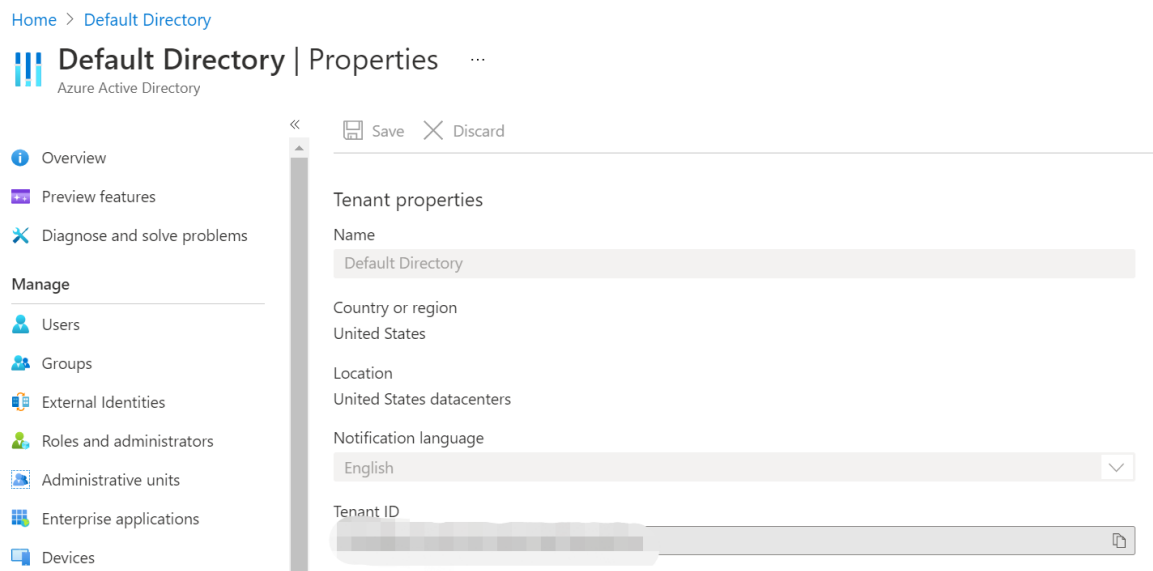
Getting the subscription ID

1. Sign in to your Azure Account through the Azure portal.
2. Select **Subscriptions**
3. Get the **Subscription ID** in which the FortiADC-VMs are deployed.
The subscription should also be the one where the storage account containing the license files is located.



Getting the tenant ID

1. Sign in to your Azure Account through the Azure portal.
2. Select **Azure Active Directory**.
3. Select **Properties**.
4. Copy the **Tenant ID**.



Deploying FortiADC HA resources from the ARM template

To deploy VRRP HA using Azure Load Balancer, the FortiADC HA resources need to be created through the ARM template. Follow the steps below to deploy the FortiADC HA resources from the ARM template.

Accessing the ARM template

You can access the ARM template through the following 2 options:

- Launching the prepared ARM template from the Fortinet GitHub to deploy directly to Azure.
- Building a custom template in Azure using the code for the ARM template as the base.

To launch the ARM template directly from the Fortinet GitHub:

1. Go to the Fortinet GitHub: <https://github.com/fortinet/fortiadc-azure-ha>.
2. Click **Deploy to Azure**.

FortiADC HA

This project contains the code and templates for the **Microsoft Azure** FortiADC HA deployments.

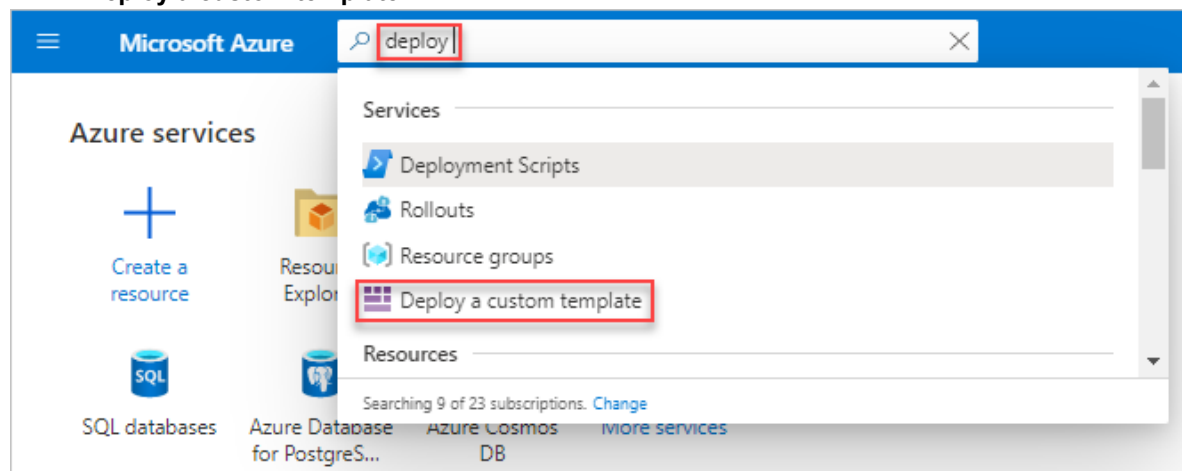
Azure Portal



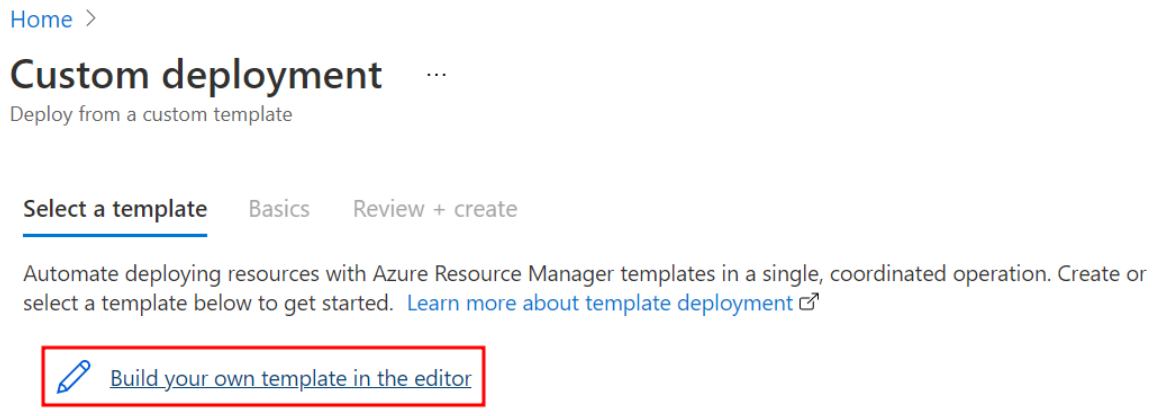
The ARM template is launched directly to Azure as a Custom deployment.

To build a custom template using the ARM template code:

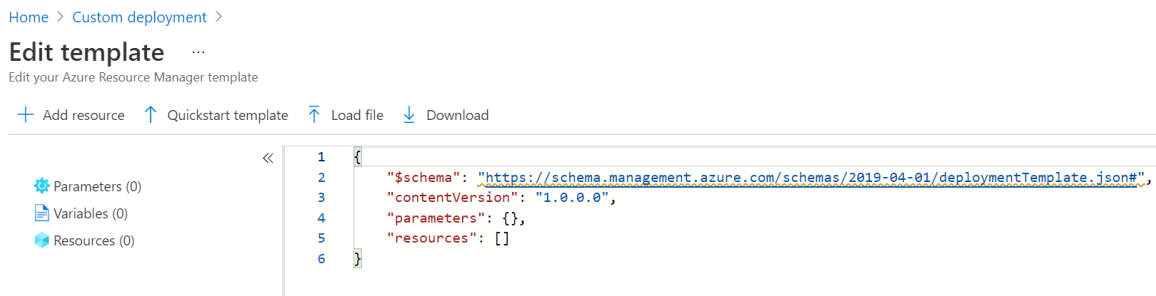
1. Sign in to your Azure Account through the Azure portal.
2. Select **Deploy a custom template**.



3. Click **Build your own template in the editor**.



4. Delete the content in the default template.



- Go to the Fortinet GitHub: https://github.com/fortinet/fortiadc-azure-ha/blob/main/templates/deploy_fadc_ha.json.
- Copy the text from **deploy_fadc_ha.json**.
- In the Azure template editor, paste the copied text. You may modify the ARM template as needed from here.
- Click **Save**.

Configuring the ARM template deployment parameters

After you have successfully launched your ARM template, configure the following parameters to complete the ARM template deployment.

- In the Azure Custom deployment where you have launched your ARM template, select the **Basics** tab.
- Under the **Project details**, select the applicable **Subscription** and **Resource group**.
Note: The Subscription and Resource group should be the same as the ones where your license files are stored. For more information, refer to [Uploading license files to Azure storage container on page 14](#).
- Under the **Instance details**, configure the following settings:

Parameter Name	Description
Region	Select the region according to the Subscription and Resource group.
Subscription Id	Apply the subscription ID in previous steps. For details, refer to Getting the subscription ID and tenant ID on page 20 .
Tenant Id	Apply the tenant ID in previous steps. For details, refer to Getting the subscription ID and tenant ID on page 20 .

Parameter Name	Description
Restapp Id	Apply the restapp ID in previous steps. For details, refer to the steps on how to get the Application ID and authentication key in Creating an Azure Active Directory application on page 15
Restapp Secret	Apply the restapp secret in previous steps. For details, refer to the steps on how to get the Application ID and authentication key in Creating an Azure Active Directory application on page 15
Region	Select the Azure server region.
Resource Name Prefix	Specify a prefix for the resources to be deployed. The names of the resources will contain the specified prefix.
Vm Sku	<p>Specify the FortiADC-VM instance types. Select from the following instance types:</p> <ul style="list-style-type: none"> • Standard_F2s_v2 • Standard_F4s_v2 • Standard_F8s_v2 • Standard_F16s_v2 • Standard_F32s_v2 <p>To ensure high performance, it is recommended to deploy a VM instance with at least 2 vCPUs and 8 GB memory. If you are using BYOL licensing type, specify an instance type that matches your FortiADC-VM licenses. For example, if your FortiADC-VM license supports 4 vCPUs, you can choose from the instance types that have 4 vCPUs.</p>
FAD Admin Username	<p>Enter an administrator username for the FortiADC instances. Note: The username cannot be "admin" or "root".</p>
FAD Admin Password	<p>Enter a password for the administrator account if you have chosen password for Authentication Type. The Azure password policy requires the password to include at least 3 of the 4 from the following:</p> <ul style="list-style-type: none"> • Lowercase characters • Uppercase characters • Numerical digits • Special characters (Regex match [W_])
FAD Image Type	Select BYOL or PAYG .
FAD Image Version	Select the image version of FortiADC-VMs. It is recommended to deploy the latest version.
FAD Count	Specify the number of virtual machines to be created in the HA group. The minimum is 1 and maximum is 2; the default is 2.
Vnet New Or Existing	Select whether to use a new or existing virtual network.

Parameter Name	Description
Vnet Resource Group	If you selected existing for Vnet New Or Existing , then specify the resource group to which the existing virtual network belongs.
Vnet Name	Specify a name for the new virtual network or enter the name of the existing virtual network.
Vnet Address Prefix	Specify the virtual network address prefix. For example, 10.2.0.0/16.
Vnet Subnet1Name	Specify a name for the public-facing subnet.
Vnet Subnet1Prefix	Specify the prefix of the public-facing subnet. For example, 10.2.0.0/24.
Vnet Subnet2Name	Specify a name for the private subnet.
Vnet Subnet2Prefix	Specify the prefix of the private subnet. For example, 10.2.1.0/24.
Internal LB Frontend IP	Specify an internal load balancer front end IP. For example, 10.2.1.6.
FAD1HAPort2IP	Specify the FAD1 HA Port2 IP. For example, 10.2.1.4.
FAD2HAPort2IP	Specify the FAD2 HA Port2 IP. For example, 10.2.1.5.
FAD1internal LB backendip	Specify the FAD1 internal load balancer IP. For example, 10.2.1.8.
FAD2internal LB backendip	Specify the FAD2 internal load balancer IP. For example, 10.2.1.9.
Fortiadc Ha Group Name	Specify a name for the FortiADC HA group.
Fortiadc Ha Group Id	Specify an ID for the FortiADC HA group. All the members in the HA group will be marked with this group ID. The minimum is 0 and the maximum is 63.
Storage Account Name	Specify the name of the storage account. Note: This is applicable for the serial console and if BYOL is selected as the FAD Image Type .
Storage License Container Name	Enter the name of the containers where the license files are stored. Note: This is applicable only if BYOL is selected as the FAD Image Type .
Storage Licensefile1	Enter one of the names of the two licenses you have uploaded into the storage license container. For example, FADXXXlic.
Storage Licensefile2	Enter one of the names of the two licenses you have uploaded into the storage license container. For example, FADXXXlic.

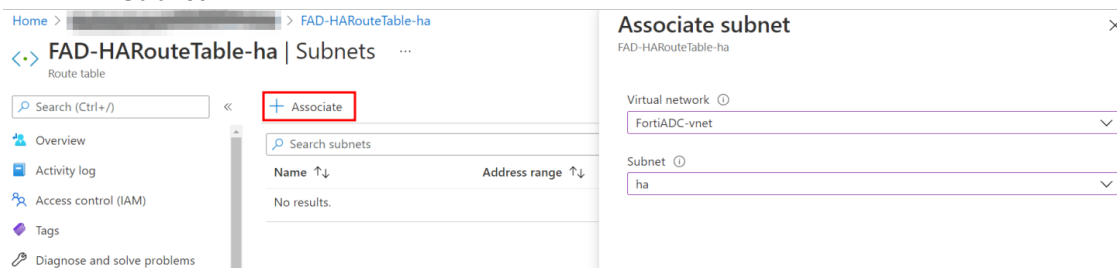
- Click **Review + create**.
- Check the resource group with all the deployment resources. The following lists the major deployed resources with the resource prefix "FAD-HA-example".

Deployed Resource	Description
FAD-HA-example-vm1	FortiADC in the HA group.
FAD-HA-example-vm2	FortiADC in the HA group.

Deployed Resource	Description
FAD-HA-example-external-nic1	External interface of the FAD-HA-example-vm1 for external access.
FAD-HA-example-internal-nic1	Internal interface of the FAD-HA-example-vm1 for internal access to the protected server. The primary IP of this network interface is also used as the HA VRRP unicast IP.
FAD-HA-example-external-nic2	External interface of the FAD-HA-example-vm2 for external access.
FAD-HA-example-internal-nic2	Internal interface of the FAD-HA-example-vm2 for internal access to the protected server. The primary IP of this network interface is also used as the HA VRRP unicast IP.
FAD-HA-example-loadbalance-internal	Internal Azure Load Balancer. This is used in the FortiADC L4 virtual server topology. For more information, see Example: FortiADC L4 Virtual Server with HA VRRP mode using Azure Load Balancer Topology on page 37 .
FAD-HA-example-loadbalance-external	External Azure Load Balancer.
FAD-HA-example-nicPublic-IP1	Provides public access to the FAD-HA-example-vm1.
FAD-HA-example-nicPublic-IP2	Provides public access to the FAD-HA-example-vm2.
FAD-HA-example-loadbalance-IP	External access to the ALB FAD-HA-example-loadbalance-external. This provides the single access endpoint for the FortiADC virtual servers.
FAD-HA-exampleRouteTable-FadCHInsideSubnet	Routing table for L4 virtual server topology. For more information, see Example: FortiADC L4 Virtual Server with HA VRRP mode using Azure Load Balancer Topology on page 37 .
FAD-HA-example-availabilitySet	Provided for redundancy and availability.
FAD-HA-example-securityGroup	Access rules for the external subnet.
FAD-HA-example-securityGroup2	Access rules for the internal subnet.
FortiADC-vnet-example	Virtual network where the FortiADCs are located.

6. If you are using an existing virtual network, you will need to manually associate the subnet2 to the route table for the FAD-HA-exampleRouteTable-FadCHInsideSubnet.
 - a. From the list of deployed resources in the existing virtual network, select **FAD-HA-exampleRouteTable-FadCHInsideSubnet**.
 - b. In the **Settings** section, select **Subnets**.
 - c. Click **+Associate**.

d. Select the **Subnet** to associate.

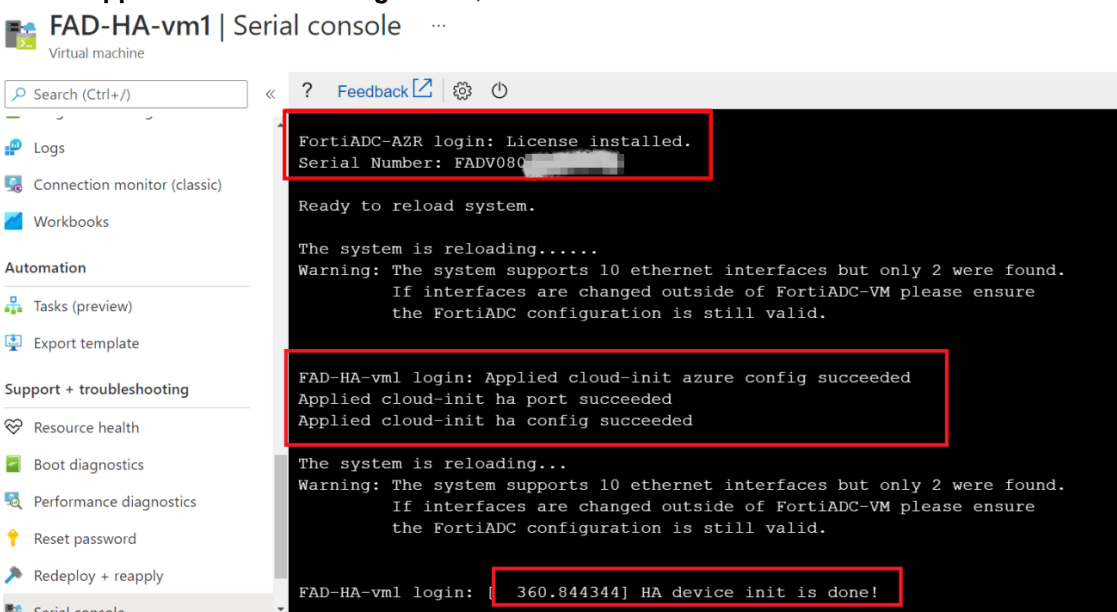


e. Click **OK**.

7. Check the FortiADC console to ensure the license (BYOL) is installed and the ha init is done.

a. Select the FortiADC-VM in the HA group. For example, the FAD-HA-example-vm1.

b. In the **Support + troubleshooting** section, select **Serial console**.



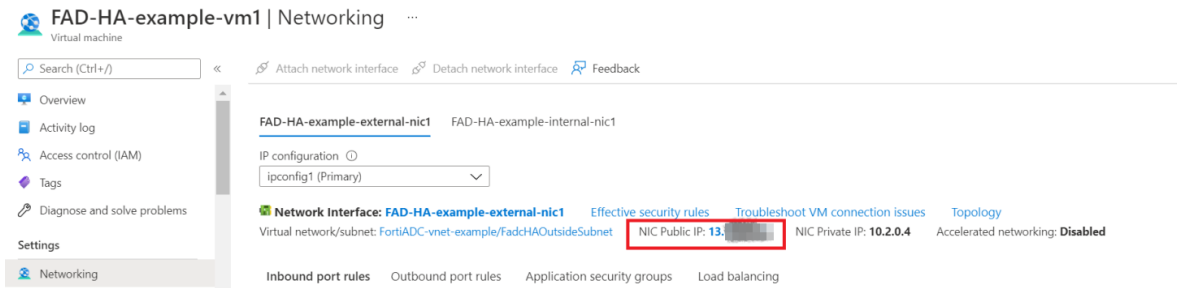
Connecting to the FortiADC GUI and CLI

After deploying the FortiADC HA resources on Azure, you will need to access the FortiADC to configure for the L7 or L4 virtual server scenario.

Obtaining the public IP address of the FortiADC-VMs

1. Go to the Azure portal to manage your resource group. Search for and select **Resource groups**.
2. Locate the resource group you have selected for the ARM template deployment and click the resource group name.
3. Locate the FortiADC-VMs by the Resource Name Prefix you have defined in the ARM template parameters and click the name.
4. In the **Settings** section, select **Networking**.

5. Take note of the **NIC Public IP**.

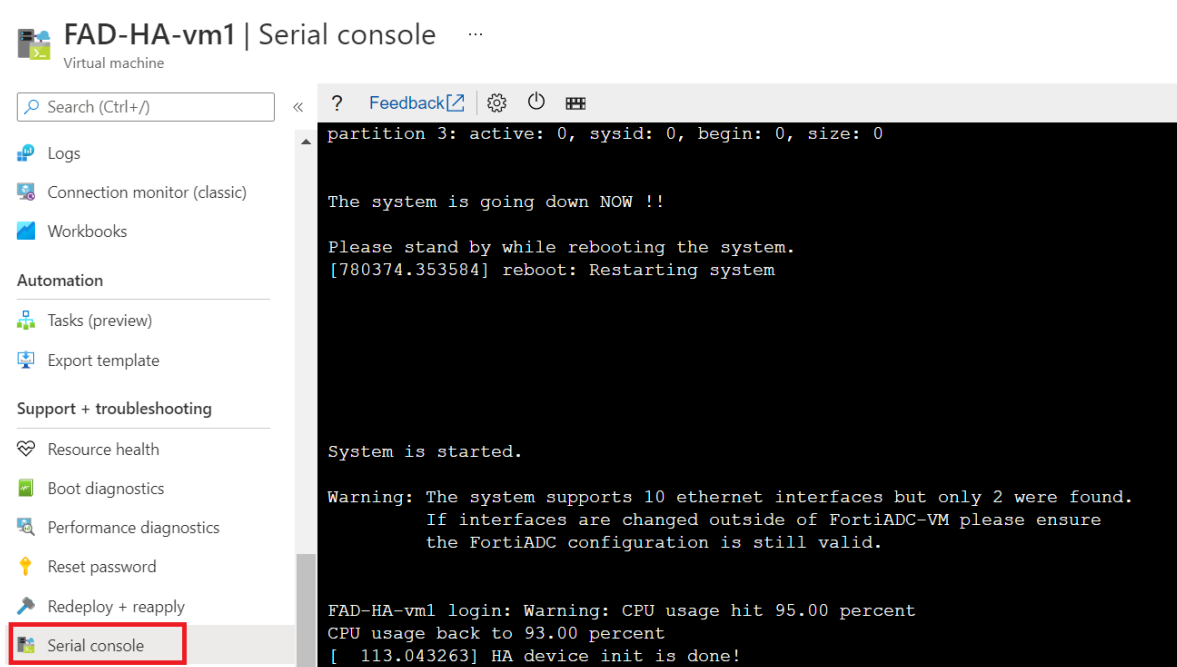


To connect to the FortiADC web UI:

1. Enter the FortiADC-VM's public IP address in a web browser's address field. For example, <https://13.x.x.x>. The HTTP access to the FortiADC GUI will be automatically redirected to HTTPS, so if you enter the HTTP port number (e.g., 80), it will be redirected to the HTTPS port (e.g., 443).
2. Log in using the **FAD Admin Username** and **FAD Admin Password** that were specified during the ARM template deployment.

To connect to the FortiADC CLI via console:

1. Go to the Azure portal to manage your resource group. Search for and select **Resource groups**.
2. In the **Support + troubleshooting** section, select **Serial console**.



To connect to the FortiADC CLI via SSH:

Follow the steps below to connect to the FortiADC-VM for Azure using the PuTTY terminal emulation software.

1. On your management computer, start [PuTTY](#).

2. To ensure that your configuration does not use environment variables that can interfere with the connection, in the **Category** tree, expand **Connection**, and then click **Data**. Remove any environment variables.
3. Click **Session**. For the **Host Name (or IP Address)**, enter the public IP address of the FortiADC-VM. For example, 13.x.x.x.
4. In **Port**, type 22.
5. For **Connection type**, select **SSH**.
6. Click **Open**.
The SSH client connects to the FortiADC appliance.
The SSH client may display a warning if this is the first time you are connecting to the FortiADC appliance and its SSH key is not yet recognized by your SSH client, or if you have previously connected to the FortiADC appliance but it used a different IP address or SSH key.
7. Click **Yes** to verify the fingerprint and accept the FortiADC appliance's SSH key. You cannot log in until you accept the key.
The CLI displays a login prompt.
8. Enter the **FAD Admin Username** and **FAD Admin Password** that were specified during the ARM template deployment.



If 3 incorrect login or password attempts occur in a row, FortiADC temporarily blacklists your IP address from the GUI and CLI. This action protects the appliance from brute force login attacks. Wait 1 minute, and then attempt the login again.

The CLI displays a prompt, such as:

```
FortiADC#
```

Managing the admin account

In its factory default configuration, FortiADC has one administrator account named `admin`. This administrator has permissions that grant full access to FortiADC's features.

The account you have created in the VM basic settings is not the `admin` account. To use the `admin` account, you need to log in to FortiADC's CLI using the account you have created, then set the password for the `admin` account.

```
config system admin
  edit "admin"
    set password "P@ssw0rd"
  next
end
```

The password for the `admin` account on Azure should not be empty.

FortiADC Virtual Server with HA VRRP mode using Azure Load Balancer

After you have deployed the ARM template, you will see the following objects in your resource group (this list includes only a small subset of all the objects you will see):

- An external Azure Load Balancer (FAD-HA-loadbalance-external)
- Azure Virtual Network (FadCHAOutsideSubnet/FadCHAInsideSubnet)
- 2 FortiADC-VMs (FAD-HA-VM1 and FAD-HA-VM2)

The FortiADC nodes are already configured with HA VRRP.

The external Azure Load Balancer is also configured with load-balancing rules that use port 80/443 for the HTTP(S) services since these are the most commonly used services in the FortiADC virtual server.

In addition to HTTP(S) services, using the Azure Load Balancer also enables the FortiADC virtual server in HA VRRP mode to provide TCP/UDP services. The protected servers located in the FadCHAInsideSubnet can be used as a non-webserver as well, such as for a FTP server. For this, you will need to adjust the protocol/port on both the Azure and FortiADC sides accordingly to accommodate the configuration in your scenario.

The following will need to be configured to allow the Azure Load Balancer to work with the FortiADC virtual server in HA VRRP mode to enable TCP/UDP services:

On the Azure side

- ALB load-balancing rules for the TCP/UDP service
- ALB health probes
- The network security group attached to the external NIC of the FortiADC to allow TCP/UDP service traffic

On the FortiADC side

- The Azure LB Backend
- The virtual server/ real server providing the TCP/UDP service
- The probing virtual server to respond to the ALB health probe packets

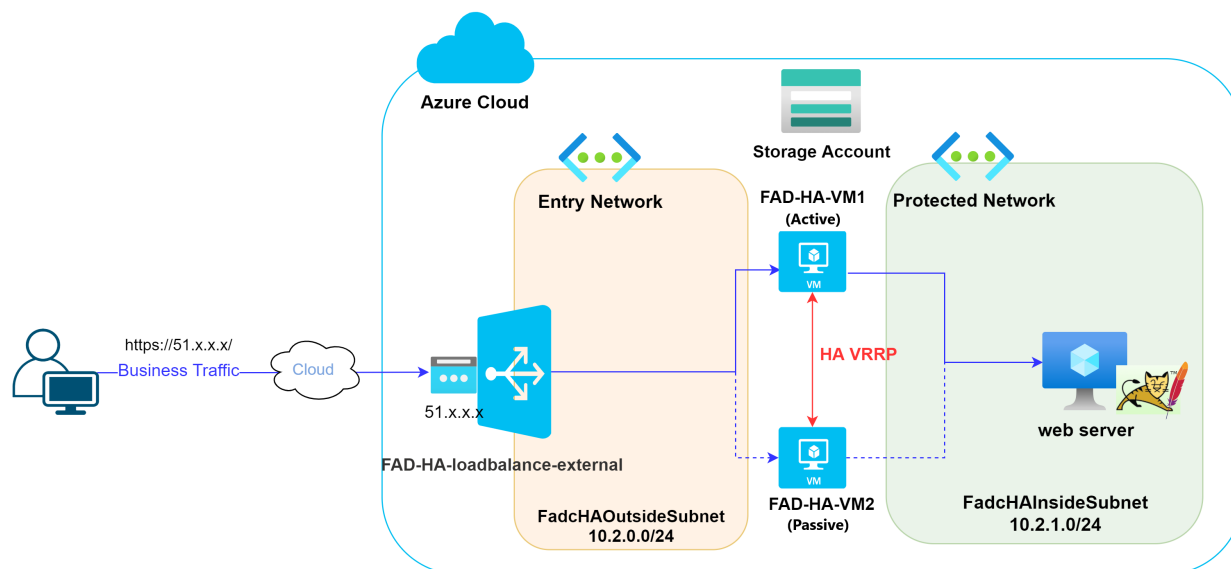
For example, if a FTP service is provided, the service link will look like ftp://51.x.x.x/file. On the Azure side, the service port used in the ALB load-balancing rule and the TCP port used in the ALB health probe are 21. The inbound traffic for port 20/21 should be allowed in the network security group attached to the FortiADC's external interface card. On the FortiADC side, configure the Azure LB Backend and the FTP virtual server with port 21. The FTP virtual server itself also acts as the probing virtual server.

For more detailed examples, see the following:

[Example: FortiADC L7 Virtual Server with HA VRRP mode using Azure Load Balancer Topology on page 30](#)

[Example: FortiADC L4 Virtual Server with HA VRRP mode using Azure Load Balancer Topology on page 37](#)

Example: FortiADC L7 Virtual Server with HA VRRP mode using Azure Load Balancer Topology



Creating an L7 virtual server to allow user access to protected resources from the internet

Follow the steps below to create an L7 virtual server to let the user access the protected resources from the internet.

In this example, we assume the following:

- The protected server is a web server.
- The web server is configured to use the VNET named FadcHAInsideSubnet.

In the example below, we will show you how to enable secure access from internet clients to a published Apache Tomcat server.

1. Prepare your protected resource (the web server) and ensure it is connected to the VNET, FadcHAInsideSubnet. For this scenario, we have prepared an Apache Tomcat server.

2. Check the Frontend IP configuration of the ALB, FAD-HA-loadbalance-external. This IP configuration, such as 51.x.x.x, is the Public IP for users to access the protected resource.

FAD-HA-loadbalance-external | Frontend IP configuration ...

Load balancer

Search (Ctrl+/) << + Add Refresh Give feedback

Filter by name...

Name	IP address
LBHaFrontEnd	51.101.101.101 (FAD-HA-loadbalance-IP)

Overview

Activity log

Access control (IAM)

Tags

Diagnose and solve problems

Settings

Frontend IP configuration

3. Check the Backend pools of the ALB, FAD-HA-loadbalance-external. You will see both the FortiADC-VMs are in the FADHaLBBackendAddrPool with IP allocated from FadcOutSideSubnet.

FAD-HA-loadbalance-external | Backend pools ...

Load balancer

Search (Ctrl+/) << + Add Refresh Give feedback

Filter by name...

Backend pool == all Resource Name == all Resource Status == all IP address == all Network interface

Group by Backend pool

Backend pool	Resource Name	Resource Status	IP Address	Network interface
▼ FADHaLBBackendAddrPool				
FADHaLBBackendAddrPool	FAD-HA-vm2	Running	10.2.0.5	FAD-HA-external-nic2
FADHaLBBackendAddrPool	FAD-HA-vm1	Running	10.2.0.7	FAD-HA-external-nic1

Overview

Activity log

Access control (IAM)

Tags

Diagnose and solve problems

Settings

Frontend IP configuration

Backend pools

Health probes

4. Check the Health Probe in ALB FAD-HA-loadbalance-external.
When receiving business traffic from the user, FAD-HA-loadbalance-external directs the business traffic to the HA VRRP active FortiADC-VM, FAD-HA-VM1. ALB determines which FortiADC-VM is the active node by sending the health check packet to both FortiADC-VMs to see which will respond to the health check packet.

FAD-HA-loadbalance-external | Health probes

Search (Ctrl+ /)

Add
Refresh
Give feedback

Filter by name...

Name ↑↓	Protocol ↑↓	Port ↑↓
tcpProbe	TCP	8080
tcpProbe1	TCP	10443

Overview
Activity log
Access control (IAM)
Tags
Diagnose and solve problems

Settings
Frontend IP configuration
Backend pools
Health probes

You can see there are 2 TCP Health Probes with port 8080 and 10443 respectively.

5. To enable the health check flow to work, the probing virtual server needs to be configured to respond to the health check packet on the FortiADC side.
 - a. Configure the backend pool IP information of the FAD-HA-loadbalance-external on both FortiADC-VMs.

FortiADC FAD-HA-vm1
HA: VRRP (Working) V6.2.0 Build0210

System
Settings
Azure LB Backend
High Availability
Traffic Group

Delete
Create New
Add Filter

Name	IP
ext-FADHaLBBackendAddrPool-1	10.2.0.7

Showing 1 to 1 of 1 entries 0 rows selected Show 25 entries

FortiADC FAD-HA-vm2
HA: VRRP (Not working) V6.2.0 Build0210

Dashboard
Security Fabric
FortiView
System
Settings
Azure LB Backend

Delete
Create New
Add Filter

Name	IP
ext-FADHaLBBackendAddrPool-1	10.2.0.5

Showing 1 to 1 of 1 entries 0 rows selected Show 25 entries

- b. Configure the probing VS which references the ext-FADHaLBBackendAddrPool-1 as the virtual server IP.
Set the probing VS with port 8080 and profile LB_PROF_TCP according to the tcpprobe as defined in the Health Probes of FAD-HA-loadbalance-external.



For FortiADC version 6.2.0 or later, LB_PROF_TCP is deprecated.
Please use LB_PROF_L7_TCP instead.

FortiADC FAD-HA-vm1 HA: VRRP (Working)

Virtual Server

Basic General Security Monitoring

Configuration

Use Azure LB Backend IP ☒

Azure LB Backend IP ext-FADHaLBBackendAddrPool-1

Port 8080

Connection Limit 0

Interface port1

Resources

Profile LB_PROF_TCP

Now, we have the probing VS using FADHaLBBackendAddrPOOL IP on both FortiADC-VMs.

FortiADC FAD-HA-vm1 HA: VRRP (Working) V6.2.0 Build0210

Virtual Server Content Rewriting Content Routing NAT Source Pool Schedule Pool Clone Pool

Delete + Create New Search

Add Filter

Name	Type	Address	Port	Profile	Status	Availability
ext_probing_vs1	Layer 7	10.2.0.7	8080	LB_PROF_TCP	Enable	✓

FortiADC FAD-HA-vm2 HA: VRRP (Not working) V6.2.0 Build0210

Virtual Server Content Rewriting Content Routing NAT Source Pool Schedule Pool Clone Pool

Delete + Create New Search

Add Filter

Name	Type	Address	Port	Profile	Status	Availability
ext_probing_vs1	Layer 7	10.2.0.5	8080	LB_PROF_TCP	Enable	✓

- Create the L7 virtual server with port 443 and LB_PROF_HTTPS profile on the FortiADC-VM1. This virtual server is going to serve the HTTPS service. We only support the ALB default route type. We also reference the azure loadbalancer backend IP in the L7 virtual server.

FortiADC FAD-HA-vm1 HA: VRRP (Working) V6.2.0 Build0210

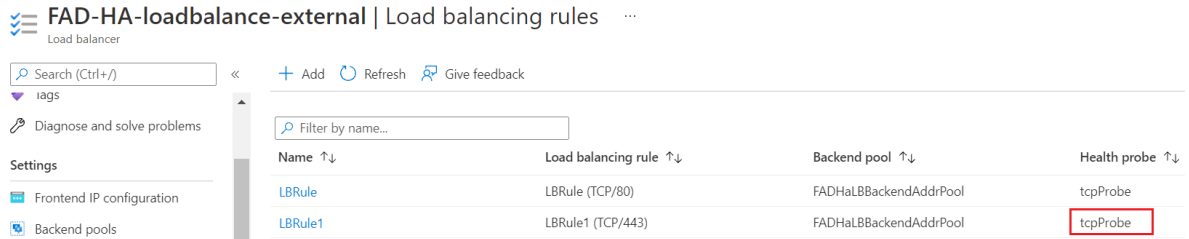
Virtual Server Content Rewriting Content Routing NAT Source Pool Schedule Pool Clone Pool

Delete + Create New Search

Add Filter

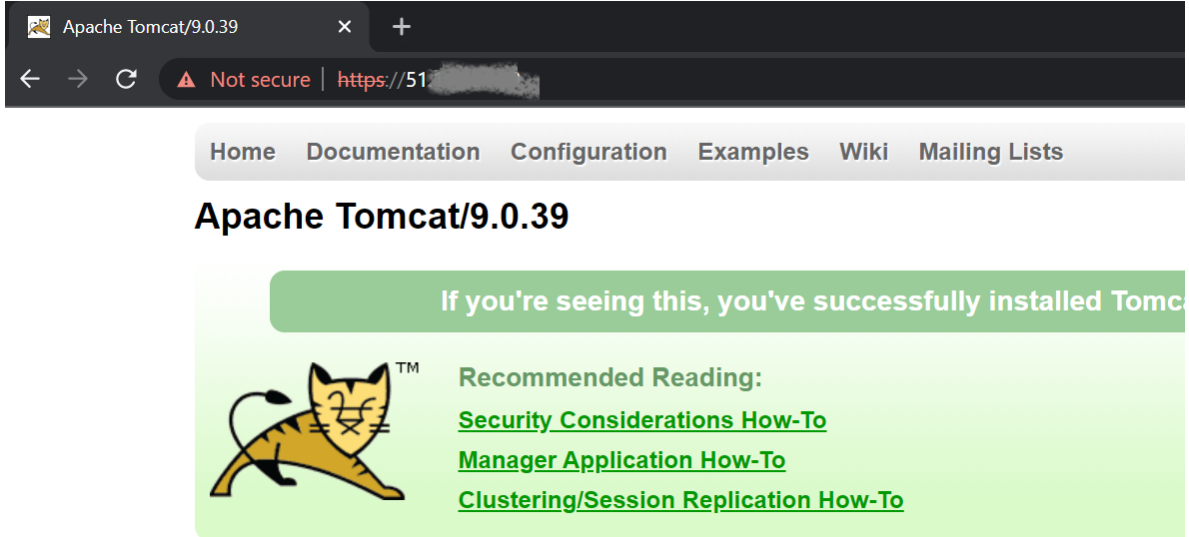
Name	Type	Address	Port	Profile	Status	Availability
ext_probing_vs1	Layer 7	10.2.0.7	8080	LB_PROF_TCP	Enable	✓
l7-vs	Layer 7	10.2.0.7	443	LB_PROF_HTTPS	Enable	✓

7. Check the Load balancing rules in FAD-HA-loadbalance-external. Change the LBRule1 to use tcpprobe as the Health probe.



Name	Load balancing rule	Backend pool	Health probe
LBRule	LBRule (TCP/80)	FADHaLBackendAddrPool	tcpProbe
LBRule1	LBRule1 (TCP/443)	FADHaLBackendAddrPool	tcpProbe

8. Try to connect to https://51.x.x.x to get apache tomcat service.



Home Documentation Configuration Examples Wiki Mailing Lists

Apache Tomcat/9.0.39

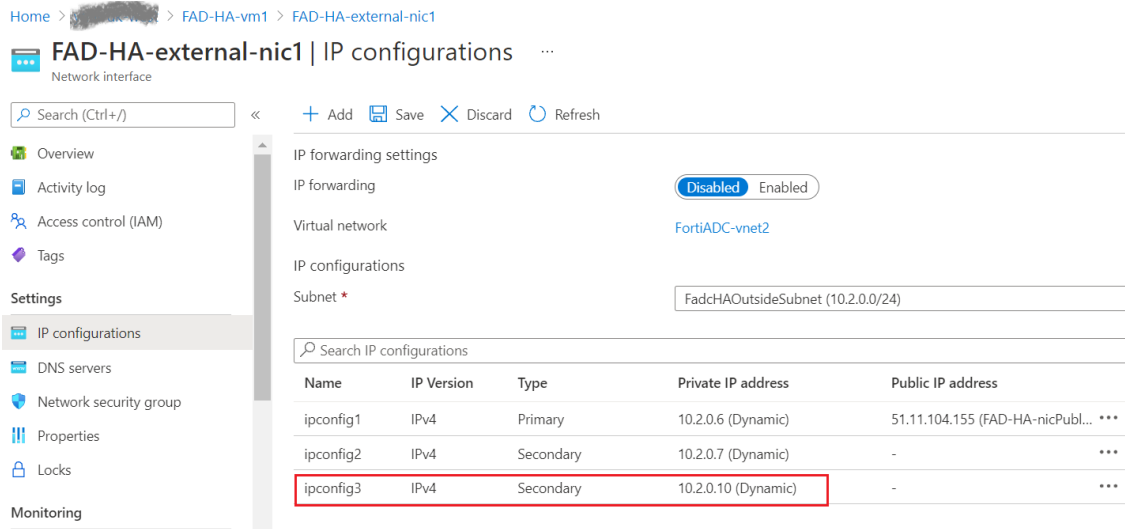
If you're seeing this, you've successfully installed Tomcat

Recommended Reading:

- [Security Considerations How-To](#)
- [Manager Application How-To](#)
- [Clustering/Session Replication How-To](#)

Now, FAD-HA-VM1 is the active node. When HA failover occurs, the health check packet returns from FAD-HA-VM2, and the FAD-HA-loadbalance-external directs the business traffic to FAD-HA-VM2. The connection to https://51.x.x.x may lose a few seconds during the failover, but will recover soon after.

9. If you want to reuse the port on the backend FortiADC, you can add more backend pools in the ALB, FAD-HA-loadbalance-external.
 - a. Add a third IP to the IP Configuration of the external interface for both of the FortiADC-VMs associated with the FadcOutSideSubnet.



Home > FAD-HA-vm1 > FAD-HA-external-nic1

FAD-HA-external-nic1 | IP configurations

Network interface

IP forwarding settings

IP forwarding: Disabled Enabled

Virtual network: FortiADC-vnet2

IP configurations

Subnet: FadcHAOutsideSubnet (10.2.0.0/24)

Name	IP Version	Type	Private IP address	Public IP address
ipconfig1	IPv4	Primary	10.2.0.6 (Dynamic)	51.11.104.155 (FAD-HA-nicPubl...)
ipconfig2	IPv4	Secondary	10.2.0.7 (Dynamic)	-
ipconfig3	IPv4	Secondary	10.2.0.10 (Dynamic)	-

Home > FAD-HA-external-nic2

FAD-HA-external-nic2 | IP configurations

Network interface

Search (Ctrl+/) << + Add Save Discard Refresh

Overview

Activity log

Access control (IAM)

Tags

Settings

IP configurations

DNS servers

Network security group

Properties

Locks

IP forwarding settings

IP forwarding: Disabled Enabled

Virtual network: FortiADC-vnet2

IP configurations

Subnet *: FadcHAOutsideSubnet (10.2.0.0/24)

Search IP configurations

Name	IP Version	Type	Private IP address	Public IP address
ipconfig1	IPv4	Primary	10.2.0.4 (Dynamic)	51.11.104.115 (FAD-HA-nicPubl...)
ipconfig2	IPv4	Secondary	10.2.0.5 (Dynamic)	-
ipconfig3	IPv4	Secondary	10.2.0.11 (Dynamic)	-

- b. Add the ipconfig3 of both FortiADC-VMs to a new Backend pool for the ALB, FAD-HA-loadbalance-external.

FAD-HA-loadbalance-external | Backend pools

Load balancer

Search (Ctrl+/) << + Add Refresh Give feedback

Filter by name...

Backend pool == all Resource Name == all Resource Status == all IP address == all Network interface == all

Group by Backend pool

Backend pool	Resource Name	Resource Status	IP Address	Network interface	Availabil
FADHaLBBBackendAddrPool					
FADHaLBBBackendAddrPool	FAD-HA-vm2	Running	10.2.0.5	FAD-HA-external-nic2	
FADHaLBBBackendAddrPool	FAD-HA-vm1	Running	10.2.0.7	FAD-HA-external-nic1	
FADHaLBBBackendAddrPool2					
FADHaLBBBackendAddrPool2	FAD-HA-vm1	Running	10.2.0.10	FAD-HA-external-nic1	
FADHaLBBBackendAddrPool2	FAD-HA-vm2	Running	10.2.0.11	FAD-HA-external-nic2	

- c. Configure the backend pool IP information of the FAD-HA-loadbalance-external on both FortiADC-VMs.

FortiADC FAD-HA-vm1 HA: VRRP (Not working) V6.2.0 Build0

Dashboard > Security Fabric > FortiView > System > Settings > Azure LB Backend > High Availability

Delete + Create New Add Filter

Name	IP
ext-FADHaLBBBackendAddrPool-1	10.2.0.7
ext-FADHaLBBBackendAddrPool-2	10.2.0.10

Showing 1 to 2 of 2 entries 0 rows selected Show 25 entries

FortiADC FAD-HA-vm2 HA: VRRP (Working) V6.2.0 Build0

Dashboard > Security Fabric > FortiView > System > Settings > Azure LB Backend

Delete + Create New Add Filter

Name	IP
ext-FADHaLBBBackendAddrPool-1	10.2.0.5
ext-FADHaLBBBackendAddrPool-2	10.2.0.11

Showing 1 to 2 of 2 entries 0 rows selected Show 25 entries

- d. Create the probing virtual server and L7 virtual server to reference ext-FADHaLBBBackendAddrPool-2 on the FortiADC-VM2. This time we create a L7 HTTP virtual server.

FortiADC FAD-HA-vm2 HA: VRRP (Working) V6.2.0 Build0210

System > Settings > Azure LB Backend > High Availability > Traffic Group > Administrator > SNMP > Replacement Messages > FortiGuard > Debug

Virtual Server Content Rewriting Content Routing NAT Source Pool Schedule Pool Clone Pool

Delete + Create New Search Add Filter

Name	Type	Address	Port	Profile	Status	Availability
I7-vs	Layer 7	10.2.0.5	443	LB_PROF_HTTPS	Enable	✓
I7-vs2	Layer 7	10.2.0.11	80	LB_PROF_HTTP	Enable	✓
ext_probing_vs1	Layer 7	10.2.0.5	8080	LB_PROF_TCP	Enable	✓
ext_probing_vs2	Layer 7	10.2.0.11	8080	LB_PROF_TCP	Enable	✓

- e. Create a Load balancing rule to use the FADHaLBBBackendAddrPool2 on FAD-HA-loadbalance-external. Map the front end port 1080 to backend port 80.

FAD-HA-loadbalance-external | Load balancing rules

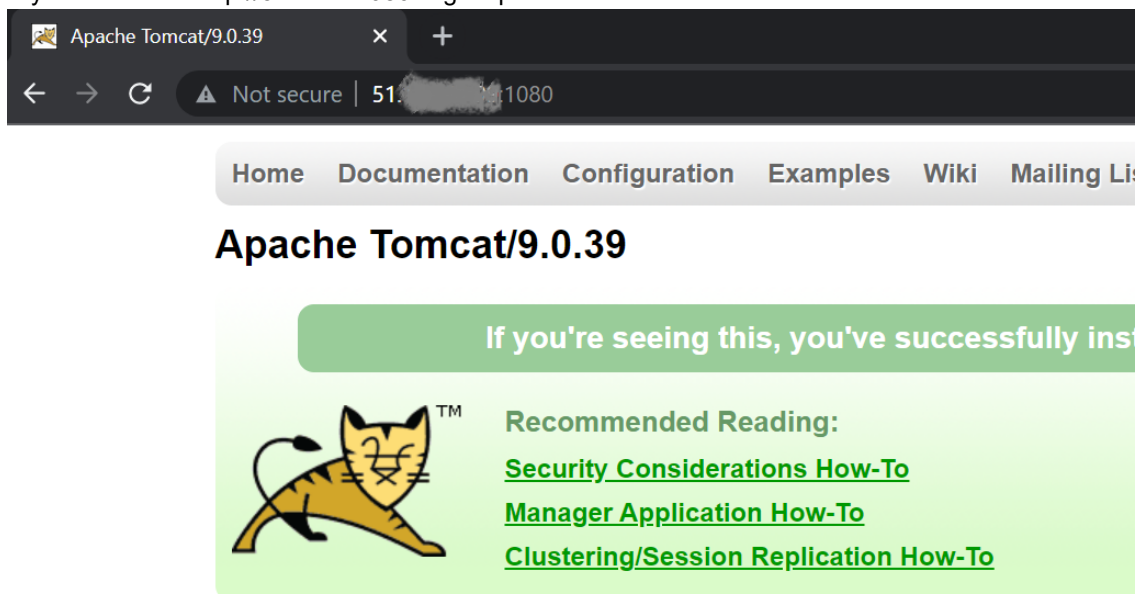
Search (Ctrl+/) iags Diagnose and solve problems Settings Frontend IP configuration Backend pools Health probes Load balancing rules

+ Add Refresh Give feedback

Filter by name...

Name	Load balancing rule	Backend pool	Health probe
LBRule	LBRule (TCP/80)	FADHaLBBBackendAddrPool	tcpProbe
LBRule1	LBRule1 (TCP/443)	FADHaLBBBackendAddrPool	tcpProbe
LBRule2	LBRule2 (TCP/1080 to TCP/80)	FADHaLBBBackendAddrPool2	tcpProbe

- f. Try to connect to `http://51.x.x.x:1080` to get apache tomcat service.

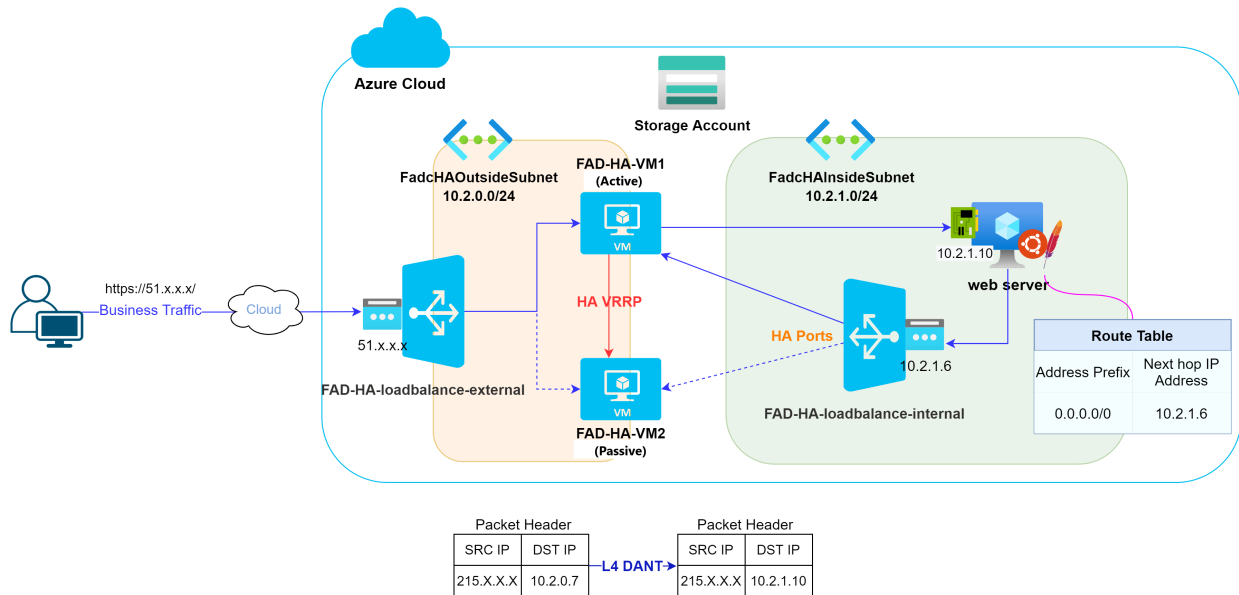


Example: FortiADC L4 Virtual Server with HA VRRP mode using Azure Load Balancer Topology

Typically, an L4 virtual server using DNAT as the packet forwarding method requires the backend real server to have a route back to FortiADC. For this, the FortiADC uses the floating IP in the network interface as the default gateway for the backend real server.

In the Azure platform, the floating IP needs to be migrated in the event of an HA failover by using an Azure API call. To avoid the issues caused by the IP migration process, the internal Azure Load Balancer was introduced into the design.

Figure 5 FortiADC HA using Azure Load Balancer for L4 VS using DNAT topology



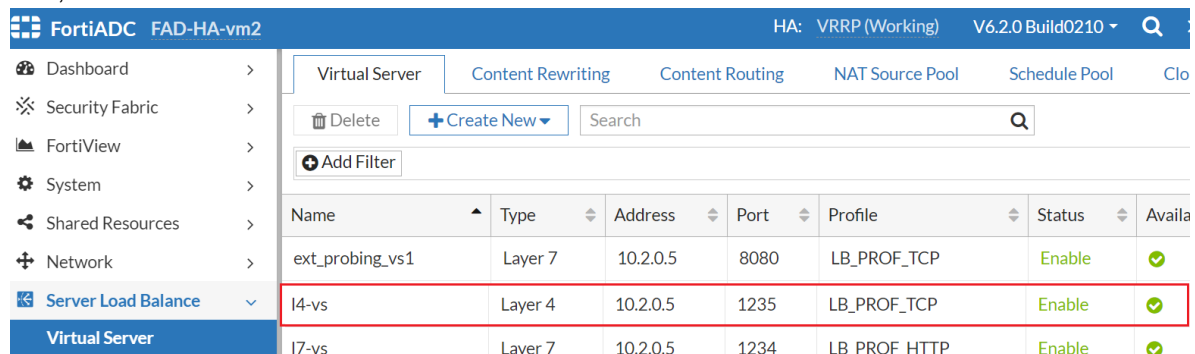
The Figure 5 example above shows the business traffic flow from the client → external ALB (FAD-HA-loadbalance-external) → the active FortiADC in HA VRRP mode → Ubuntu Apache web server → internal ALB (FAD-HA-loadbalance-internal) → the active FortiADC in HA VRRP mode → external ALB (FAD-HA-loadbalance-external) → the client.

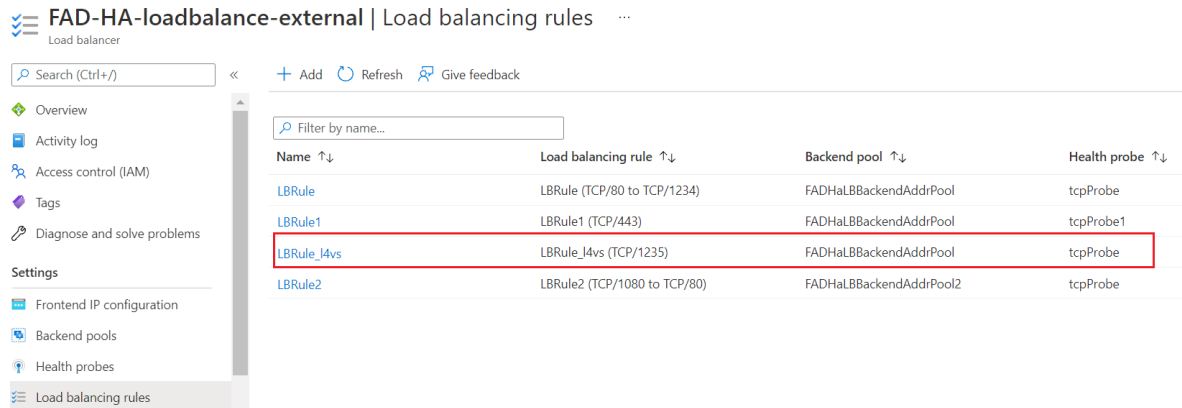
In this deployment, we use the front end IP of the internal Azure Load Balancer instead of the floating IP in the FortiADC network interface as the web server default gateway. By doing so, there is no need to migrate the floating IP during an HA failover.

Creating an L4 virtual server using DNAT as the packet forwarding method with Azure Load Balancer

Similar to the steps taken for the [Example: FortiADC L7 Virtual Server with HA VRRP mode using Azure Load Balancer Topology on page 30](#), follow the steps below to build an L4 virtual server using DNAT as the packet forwarding method on FortiADC.

1. Create the L4 virtual server on FortiADC and add the corresponding Load balancing rule on the external ALB, FAD-HA-loadbalance-external.





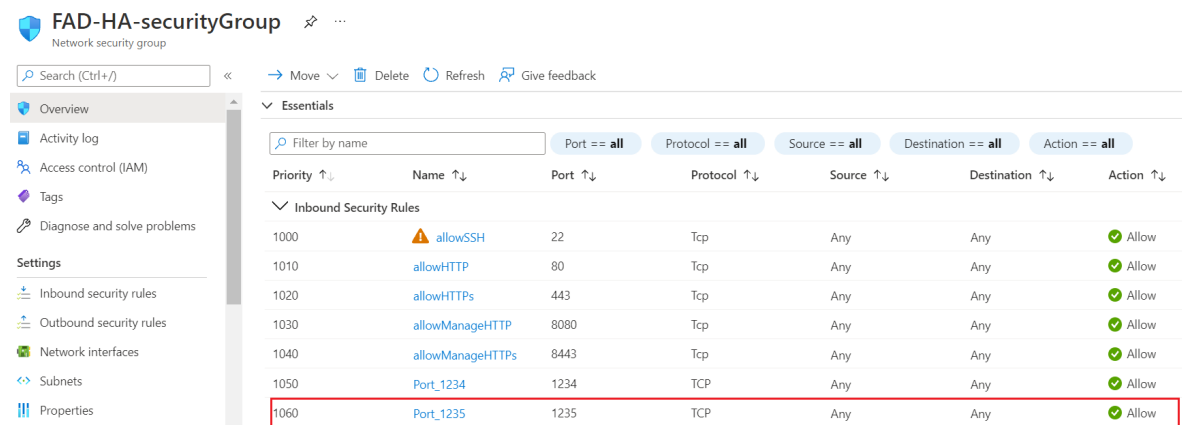
FAD-HA-loadbalance-external | Load balancing rules

Search (Ctrl+/) << + Add Refresh Give feedback

Filter by name...

Name ↑↓	Load balancing rule ↑↓	Backend pool ↑↓	Health probe ↑↓
LBRule	LBRule (TCP/80 to TCP/1234)	FADHaLBBBackendAddrPool	tcpProbe
LBRule1	LBRule1 (TCP/443)	FADHaLBBBackendAddrPool	tcpProbe1
LBRule14vs	LBRule14vs (TCP/1235)	FADHaLBBBackendAddrPool	tcpProbe
LBRule2	LBRule2 (TCP/1080 to TCP/80)	FADHaLBBBackendAddrPool2	tcpProbe

- Check the inbound rule to allow access to port 1235 in the security group FAD-HA-Security-Group that is attached to both the FortiADC-VMs' network interface in the FadchAOOutsideSubnet.



FAD-HA-securityGroup

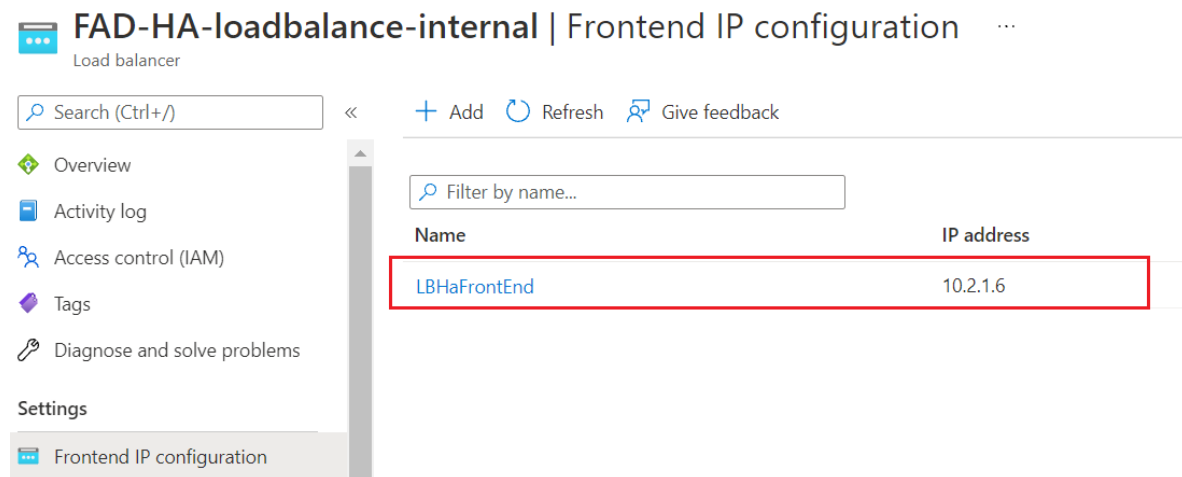
Search (Ctrl+/) << Move Delete Refresh Give feedback

Filter by name

Port == all Protocol == all Source == all Destination == all Action == all

Priority ↑↓	Name ↑↓	Port ↑↓	Protocol ↑↓	Source ↑↓	Destination ↑↓	Action ↑↓
1000	allowSSH	22	Tcp	Any	Any	Allow
1010	allowHTTP	80	Tcp	Any	Any	Allow
1020	allowHTTPs	443	Tcp	Any	Any	Allow
1030	allowManageHTTP	8080	Tcp	Any	Any	Allow
1040	allowManageHTTPs	8443	Tcp	Any	Any	Allow
1050	Port_1234	1234	TCP	Any	Any	Allow
1060	Port_1235	1235	TCP	Any	Any	Allow

- In the FAD-HA-loadbalance-internal, check the front end IP, which is the default gateway for the backend web server.



FAD-HA-loadbalance-internal | Frontend IP configuration

Search (Ctrl+/) << + Add Refresh Give feedback

Filter by name...

Name	IP address
LBHaFrontEnd	10.2.1.6

- Check the default route gateway of the backend web server. In this example, we use the Ubuntu Apache2 server as the backend web server. The default gateway should be configured to 10.2.1.6 to ensure the responding business traffic will go through the internal ALB, FAD-HA-loadbalance-internal.

```

root@ubuntu:/# route -n
Kernel IP routing table
Destination      Gateway         Genmask         Flags Metric Ref    Use Iface
0.0.0.0          10.2.1.6        0.0.0.0         UG    0      0      0 eth1
10.2.1.0         0.0.0.0         255.255.255.0   U      0      0      0 eth1

```

- a. Check the route table attached to the FadCHALInsideSubnet on Azure Cloud.

If you are using the ARM template to create the FortiADC HA pair with an **existing** network, you will need to manually associate the route table FAD-HARouteTable-FadCHALInsideSubnet to FadCHALInsideSubnet.

5. Check the Backend pools of FAD-HA-loadbalance-internal and add the ALB backend configuration on both FortiADC-VMs.

FortiADC FAD-HA-vm1
HA: VRRP (Not working) V6.2.0 Build0210

Dashboard >
Security Fabric >
FortiView >
System >
Settings

Delete Create New Add Filter

Name	IP
ext-FADHaLBBBackendAddrPool-1	10.2.0.7
int-FADHaLBBBackendAddrPool-1	10.2.1.8

Showing 1 to 2 of 2 entries 0 rows selected Show 25 entries

FortiADC FAD-HA-vm2
HA: VRRP (Working) V6.2.0 Build0210

Dashboard >
Security Fabric >
FortiView >
System >
Settings

Delete Create New Add Filter

Name	IP
ext-FADHaLBBBackendAddrPool-1	10.2.0.5
int-FADHaLBBBackendAddrPool-1	10.2.1.9

Showing 1 to 2 of 2 entries 0 rows selected Show 25 entries

- Check the Health Probe of the FAD-HA-loadbalance-internal and add the internal probing VS.

FAD-HA-loadbalance-internal | Health probes

Search (Ctrl+/)
Add Refresh Give feedback

Overview
Activity log
Access control (IAM)
Tags
Diagnose and solve problems
Settings
Frontend IP configuration
Backend pools
Health probes

Filter by name...

Name	Protocol	Port	Used By
lbprobe	TCP	8080	lbruleFE2all

FortiADC FAD-HA-vm2
HA: VRRP (Working) V6.2.0 Build0210

Dashboard >
Security Fabric >
FortiView >
System >
Shared Resources >
Network >
Server Load Balance >
Virtual Server

Delete Create New Search Add Filter

Name	Type	Address	Port	Profile	Status	Availability
ext_probing_vs1	Layer 7	10.2.0.5	8080	LB_PROF_TCP	Enable	✓
int-probing-vs1	Layer 7	10.2.1.9	8080	LB_PROF_TCP	Enable	✓
I4-vs	Layer 4	10.2.0.5	1235	LB_PROF_TCP	Enable	✓
I7-vs	Layer 7	10.2.0.5	1234	LB_PROF_HTTP	Enable	✓

- Check the Load balancing rule of FAD-HA-loadbalance-internal.
The HA port is enabled so the FAD-HA-loadbalance-internal can load balance on all ports for TCP and UDP protocols. This means that the FAD-HA-loadbalance-internal can accept any TCP and UDP service

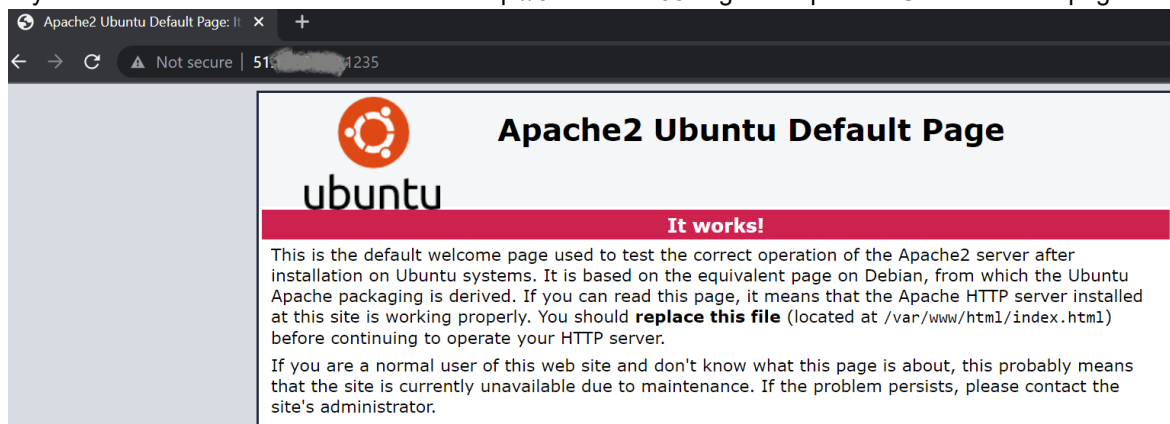
traffic and direct them to the active FortiADC-VM based on the health probe.

lbruleFE2all

FAD-HA-loadbalance-internal

Name	lbruleFE2all
IP Version *	<input checked="" type="radio"/> IPv4 <input type="radio"/> IPv6
Frontend IP address *	LBHaFrontEnd (10.2.1.6)
	<input checked="" type="checkbox"/> HA Ports
Backend pool *	FADHaLBBackendAddrPool
Health probe *	lbprobe (TCP:8080) Create new
Session persistence	None
Idle timeout (minutes) *	<input type="range"/> 15
TCP reset	<input checked="" type="radio"/> Disabled <input type="radio"/> Enabled

8. Try to connect to the L4 virtual server with `http://51.x.x.x:1235` to get an Apache2 Ubuntu default page.



FortiADC Virtual Server with traffic group in HA VRRP using Azure Load Balancer

In FortiADC, the virtual server is specified with a default traffic group. In the traffic group, there will be one active node among all nodes in the group. The virtual servers located in the active node serve the services. Therefore,

if you have multiple traffic groups with multiple virtual servers assigned to each traffic group, ensure that there is at least one probing virtual server active in each traffic group. For more information, see [FortiADC Handbook on traffic groups](#).

In the example scenario below, the following specifications are defined for the traffic groups *default* and *trafficGroup2*:

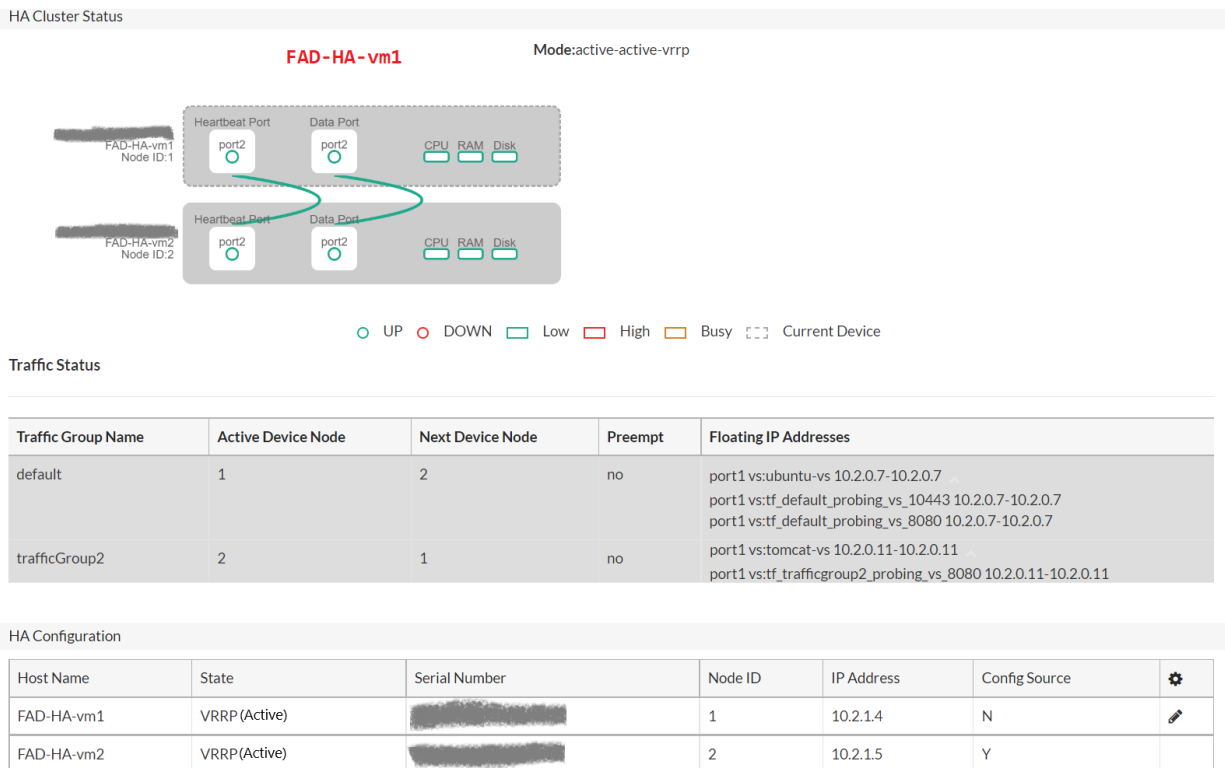
default

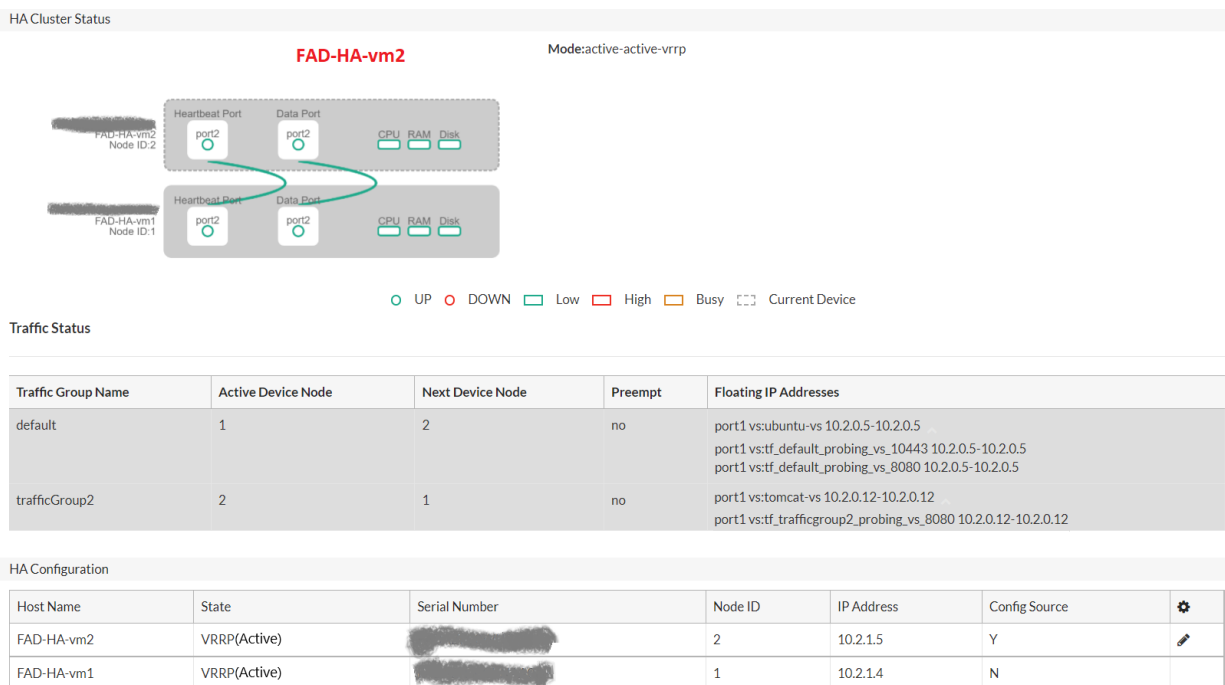
- Virtual server: ubuntu-vs/tf_default_probing_vs_10443/tf_default_probing_vs_8080
- Probing virtual server: tf_default_probing_vs_10443/tf_default_probing_vs_8080
- Active node: FAD-HA-vm1

trafficGroup2

- Virtual server: tomcat-vs/tf_trafficgroup2_probing_vs_8080
- Probing virtual server: tf_trafficgroup2_probing_vs_8080
- Active node: FAD-HA-vm2

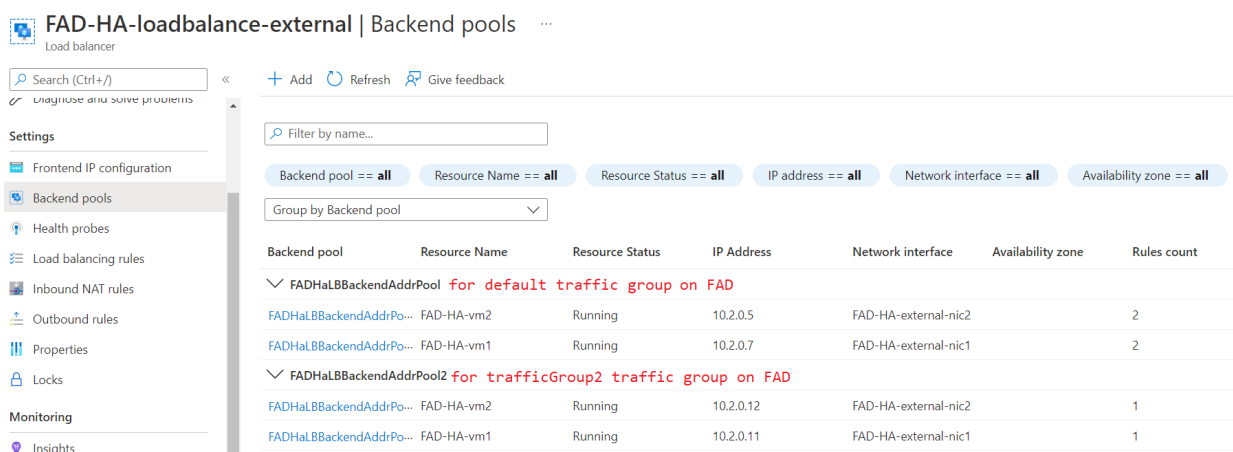
Below is the virtual server and HA traffic group configuration on the FAD-HA-vm1 and FAD-HA-vm2 respectively.





Compare the backend pool in the external ALB shown in the table and figure below. The virtual servers in the default traffic group references the IP configuration from FADHaLBBackendAddrPool and the virtual servers in the trafficGroup2 traffic group references the IP configuration from FADHaLBBackendAddrPool2.

ALB Backend Pool	Traffic Groups on FAD
FADHaLBBackendAddrPool	default(active device node: FAD-HA-vm1)
FAD-HA-vm1(10.2.0.7)	
FAD-HA-vm2(10.2.0.5)	
FADHaLBBackendAddrPool2	trafficGroup2(active device node: FAD-HA-vm2)
FAD-HA-vm1(10.2.0.11)	
FAD-HA-vm2(10.2.0.12)	



Important Note:

- The IP (floating IP) cannot be cross assigned in two different traffic groups. For example, IP 10.2.0.5/10.2.0.7 is used in the default traffic group, then it cannot be used in trafficGroup2. Therefore, when creating a new traffic group on FortiADC, you must create the corresponding ALB backend pool on the Azure side, and configure the new ALB backend pool on the FortiADC Azure LB Backend for further usage.
- When changing the IP references in the virtual server, you must change the traffic group together. If the change of IP references and traffic group in the virtual server is not performed in one step, there will be abnormal HA behavior. In this case, you have to reboot both FortiADC nodes to allow HA to function correctly.

Cross reference the above traffic group information with the ALB Load balancing rules and Health probes configuration: the LBRule for ubuntu-vs and LBRule3 for tomcat-vs uses the ALB health probe, tcpProbe, which sends the TCP health check packet with the destination port 8080 to the ALB backend pool, FADHaLBBackendAddrPool and FADHaLBBackendAddrPool2. And LBRule1 uses the ALB health probe, tcpProbe1, which sends the TCP health check packet with the destination port 10443 to the ALB backend pool, FADHaLBBackendAddrPool.

The table and figures below illustrates the relationship of the ALB load balancing rules with the FortiADC virtual server, ALB Backend Pool and ALB Health probe.

ALB Load Balancing Rule/FAD VS	ALB Backend Pool	ALB Health Probe
LBRule(TCP/80)/ubuntu-vs	FADHaLBBackendAddrPool	tcpProbe(8080)
LBRule1(TCP/443)	FADHaLBBackendAddrPool	tcpProbe(10443)
LBRule(TCP/8081)/tomcat-vs	FADHaLBBackendAddrPool2	tcpProbe(8080)

FAD-HA-loadbalance-external | Load balancing rules

Load balancer

<<
+ Add
↺ Refresh
🗨 Give feedback

Diagnose and solve problems

Settings

- Frontend IP configuration
- Backend pools
- Health probes
- Load balancing rules**

Name ↑↓	Load balancing rule ↑↓	Backend pool ↑↓	Health probe ↑↓
LBRule	LBRule (TCP/80)	FADHaLBBBackendAddrPool	tcpProbe
LBRule1	LBRule1 (TCP/443)	FADHaLBBBackendAddrPool	tcpProbe1
LBRule3	LBRule3 (TCP/8081)	FADHaLBBBackendAddrPool2	tcpProbe

FAD-HA-loadbalance-external | Health probes

Load balancer

<<
+ Add
↺ Refresh
🗨 Give feedback

Diagnose and solve problems

Settings

- Frontend IP configuration
- Backend pools
- Health probes**

Name ↑↓	Protocol ↑↓	Port ↑↓	Used By ↑↓
tcpProbe	TCP	8080	2 rules
tcpProbe1	TCP	10443	LBRule1

Figure 6 Business traffic flow for multiple traffic groups

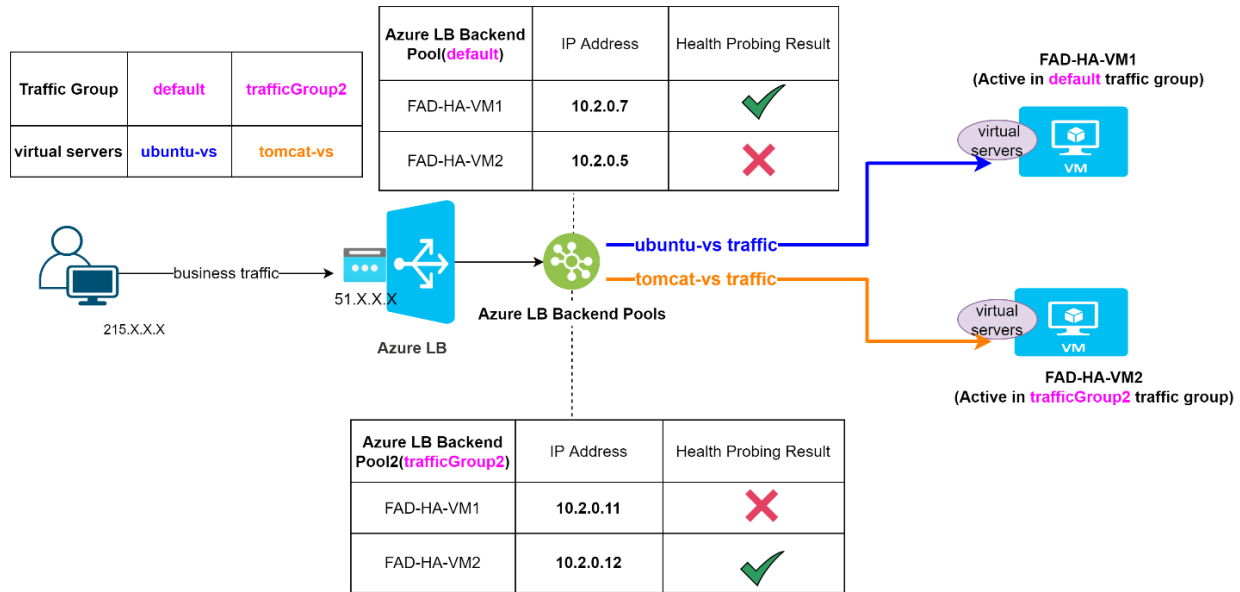
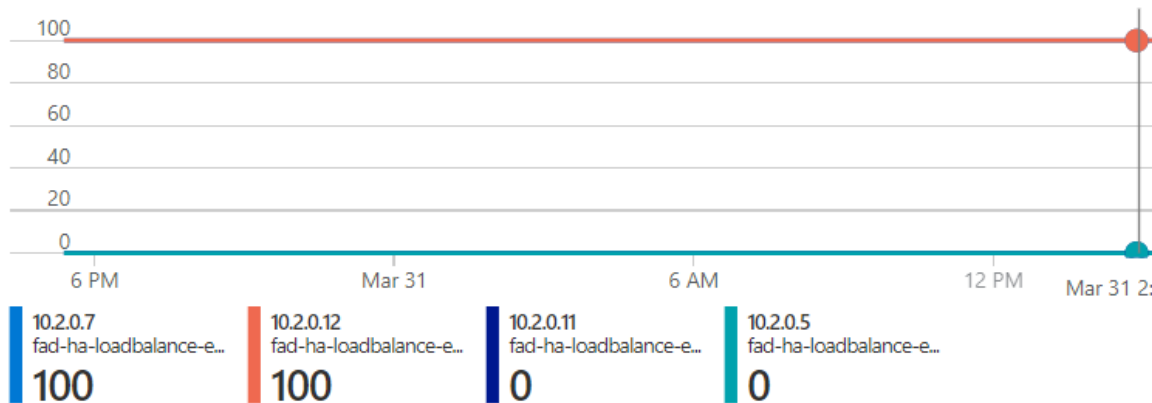


Figure 6 shows the business traffic flow in this example scenario, in which the active node in default traffic group is FAD-HA-vm1. When ALB receives the business traffic for ubuntu-vs that is assigned to the default traffic group, the ALB directs traffic to FAD-HA-vm1 based on the instant health probing status. See the ALB health probing status in the figure below, where FAD-HA-vm1(IP:10.2.0.7) responds to the health probe packets with 100% success rate, and FAD-HA-vm2(IP:10.2.0.5) does not respond because it is in HA passive mode in the default traffic group.

Likewise, the business traffic for tomcat-vs assigned to trafficGroup2 is directed to the active node, FAD-HA-vm2.

Health Probe Status by Backend IP Address

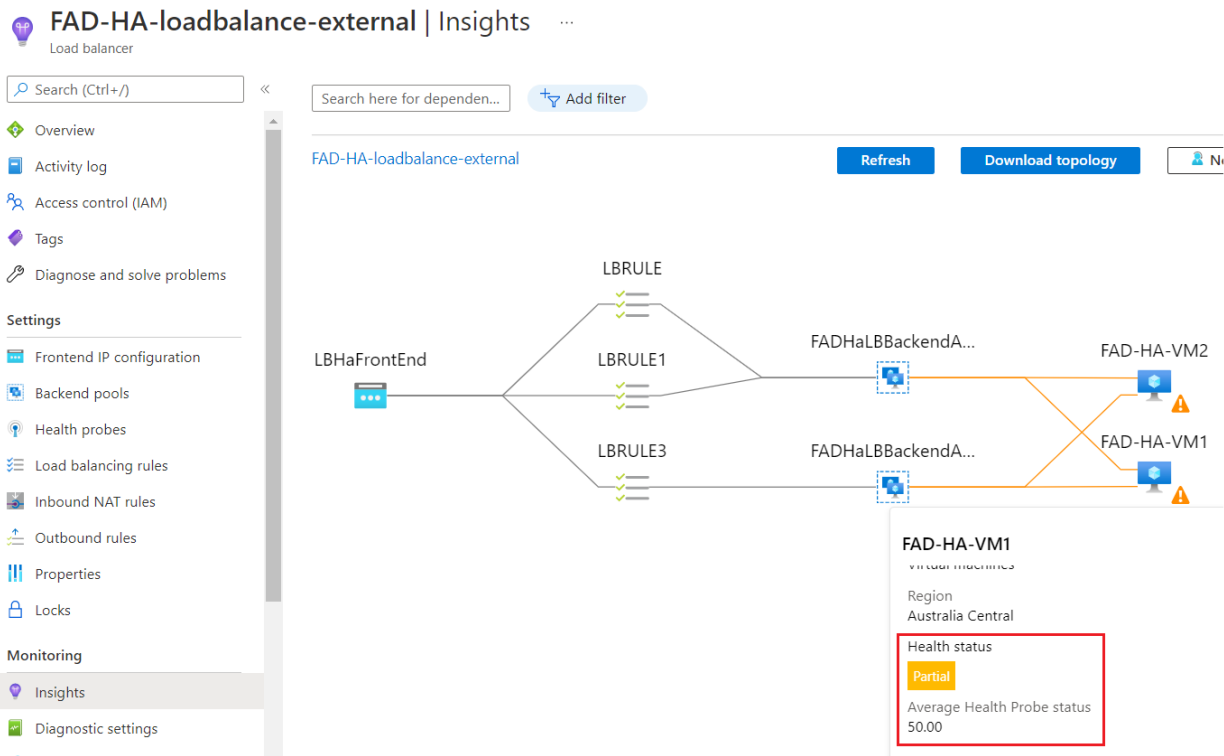


Backend Pool - Traffic Group - Nodes in the group (red means active)

FADHaLBBackendAddrPool - default - FAD-HA-vm2(10.2.0.5), **FAD-HA-vm1(10.2.0.7)**

FADHaLBBackendAddrPool2 - trafficGroup2 - **FAD-HA-vm2(10.2.0.12)**, FAD-HA-vm1(10.2.0.11)

Since FAD-HA-vm2(10.2.0.5) in the default traffic group and FAD-HA-vm1(10.2.0.11) in trafficGroup2 are in HA passive mode, the ALB Health Probe status shows 0% success rate for these two IP configurations. Therefore, you will see an average health probe status with success rate 50% for FAD-HA-vm1 and FAD-HA-vm2.



Note:

ALB directs business traffic based on the instant health probe status, not the average one. Therefore, in the case where FAD-HA-vm1 is the active node in multiple ALB backend pools, and FAD-HA-vm2 is active in only one ALB backend pool, the average health probe status for FAD-HA-vm1 should be healthy/partial and for FAD-HA-vm2 is unhealthy. Regardless of what the average health probe status shows, the business traffic can still be forwarded to the correct FortiADC node based on the instant health probe status.

Virtual Server NAT Pool and Firewall NAT SNAT configuration in HA mode

For FortiADC 6.2.2, each FortiADC-VM instance maintains its own IP configuration in the HA VRRP mode for the virtual server NAT pool and firewall NAT SNAT on both the FortiADC side and the Azure platform. However, this means the IP configuration used in the VS NAT pool and the firewall NAT SNAT **will not be** synchronized in the HA cluster when HA VRRP mode is enabled. When these IP configurations are no longer synchronized in the HA cluster, it will impact the attributes in the IP configurations.

The following attributes will no longer be synchronized in the HA cluster for each IP configuration:

IP configuration	Attribute
VS NAT pool (IPPool)	ip-min, ip-max, ip6-min, ip6-max
Firewall NAT SNAT	trans-to-ip, trans-to-ip-start, trans-to-ip-end



The independent IP configuration design for the VS NAT pool and the firewall NAT SNAT in HA VRRP mode is only applicable to Azure based FortiADC images.

After upgrading to 6.2.2, if you are using the VS NAT pool and/or firewall NAT SNAT IP configurations in your HA VRRP deployment, you must reconfigure these IP configurations on the passive FortiADC.

To reconfigure the VS NAT pool/ firewall NAT SNAT IP configurations on the passive FortiADC:

1. Check the IP configuration of the VS NAT pool/ firewall NAT SNAT of the active FortiADC instance on both the FortiADC side and Azure platform.
In the example below, the FAD-HA-vm1 is the active node. The address range of the virtual server NAT pool, nat is configured as 10.2.1.15 - 10.2.1.16.

The screenshot shows the FortiADC web interface for FAD-HA-vm1. The 'NAT Source Pool' tab is selected. A table lists the NAT source pools:

Name	Interface	Address Type	Address Range
nat	port2	IPv4	10.2.1.15 - 10.2.1.16

The 'Address Range' for the 'nat' pool is highlighted with a red box.

The corresponding address range should be configured on the associated network interface, FAD-HA-internal-nic1 on the Azure platform.

FAD-HA-internal-nic1 | IP configurations ...

Network interface

Search (Ctrl+/) << + Add Save Discard Refresh

Overview

Activity log

Access control (IAM)

Tags

Settings

IP configurations

DNS servers

Network security group

Properties

Locks

Monitoring

Insights

Alerts

Gateway Load balancer None

IP configurations

Subnet FadCHAIInsideSubnet1 (10.2.1.0/24)

The associated virtual machine 'FAD-HA-vm1' must be either stopped or deallocated in order to be able to edit the subnet.

Search IP configurations

Name	IP Version	Type	Private IP address	Public IP address
ipconfig1	IPv4	Primary	10.2.1.4 (Static)	-
ipconfig2	IPv4	Secondary	10.2.1.8 (Static)	-
ipconfig3-nat	IPv4	Secondary	10.2.1.16 (Static)	-
ipconfig4-nat	IPv4	Secondary	10.2.1.15 (Static)	-

2. Add the new IP configuration of the network interface attached to the passive FortiADC instance on the Azure platform.

Continuing with the previous example, the FAD-HA-vm2 is the passive node. So, you will add the new IP configuration 10.2.1.17 - 10.2.1.18 for FAD-HA-internal-nic2 on the Azure platform. This address range will be used in the virtual server NAT pool, nat on the FortiADC, FAD-HA-vm2.

FAD-HA-internal-nic2 | IP configurations ...

Network interface

Search (Ctrl+/) << + Add Save Discard Refresh

Overview

Activity log

Access control (IAM)

Tags

Settings

IP configurations

DNS servers

Network security group

Properties

Locks

Monitoring

Insights

Alerts

IP forwarding Disabled **Enabled**

Virtual network FortiADC-vnet3

IP configurations

Subnet FadCHAIInsideSubnet1 (10.2.1.0/24)

The associated virtual machine 'FAD-HA-vm2' must be either stopped or deallocated in order to be able to edit the subnet.

Search IP configurations

Name	IP Version	Type	Private IP address	Public IP address
ipconfig1	IPv4	Primary	10.2.1.5 (Static)	-
ipconfig2	IPv4	Secondary	10.2.1.9 (Static)	-
ipconfig3-nat	IPv4	Secondary	10.2.1.17 (Static)	-
ipconfig4-nat	IPv4	Secondary	10.2.1.18 (Static)	-

3. According to the IP allocated on the Azure platform in step 2, reconfigure the IP configuration in the VS NAT pool/ firewall NAT SNAT for the passive FortiADC.
In the example below, reconfigure the virtual server NAT pool, nat with the new address range 10.2.1.17 -

10.2.1.18 on FAD-HA-vm2.

FortiADC FAD-HA-vm2 HA: VRRP (Not working) V6.2.2 Build0500

Virtual Server Content Rewriting Content Routing NAT Source Pool Schedule Pool

Delete Create New Add Filter

Name	Interface	Address Type	Address Range
nat	port2	IPv4	10.2.1.17 - 10.2.1.18

Showing 1 to 1 of 1 entries 0 rows selected Show 25 entries

Server Load Balance

Debugging Azure API call in HA mode

The Azure API call is still required to migrate the IP configuration on the Azure platform for the Firewall 1-1 NAT during an HA failover. To check the migration process, you can use the following command on the new active FortiADC-VM to get the migration status.

```
FAD-HA-vm2 # diagnose debug enable  
FAD-HA-vm2 # diagnose debug module netd all set
```



FORTINET



Copyright© 2023 Fortinet, Inc. All rights reserved. Fortinet®, FortiGate®, FortiCare® and FortiGuard®, and certain other marks are registered trademarks of Fortinet, Inc., in the U.S. and other jurisdictions, and other Fortinet names herein may also be registered and/or common law trademarks of Fortinet. All other product or company names may be trademarks of their respective owners. Performance and other metrics contained herein were attained in internal lab tests under ideal conditions, and actual performance and other results may vary. Network variables, different network environments and other conditions may affect performance results. Nothing herein represents any binding commitment by Fortinet, and Fortinet disclaims all warranties, whether express or implied, except to the extent Fortinet enters a binding written contract, signed by Fortinet's General Counsel, with a purchaser that expressly warrants that the identified product will perform according to certain expressly-identified performance metrics and, in such event, only the specific performance metrics expressly identified in such binding written contract shall be binding on Fortinet. For absolute clarity, any such warranty will be limited to performance in the same ideal conditions as in Fortinet's internal lab tests. In no event does Fortinet make any commitment related to future deliverables, features or development, and circumstances may change such that any forward-looking statements herein are not accurate. Fortinet disclaims in full any covenants, representations, and guarantees pursuant hereto, whether express or implied. Fortinet reserves the right to change, modify, transfer, or otherwise revise this publication without notice, and the most current version of the publication shall be applicable.