Cloudbrink Connector - Azure

Cloudbrink's Software Defined Mobility enables enterprises to deliver best-in-class quality of experience and security for their end users in the new mobile-first and cloud-native world. Cloudbrink achieves this through three simple components:

1. The Brink Agent is installed on end user devices, with all major platforms supported.
2. Enterprise access points are automatically created via machine learning in close proximity to the end user, enabling Cloudbrink's revolutionary overlay protocol to overcome the most challenging last-mile network conditions, delivering best-in-class, hi fidelity quality of experience for the end-user no matter the network they are connected to.
3. To provide end-to-end security, a Cloudbrink Connector is deployed in the customer's data center or cloud environment, creating a dark tunnel connection from the end user to their applications.

This document covers deploying the Cloudbrink Connector in an Azure environment.

Introduction

This document will guide you to create an active-standby Cloudbrink Connector in Azure. Steps 7 through 10 must be completed twice in order to create two instances for the active-standby pair.

Prerequisites

- A hosting Linux VM with free space greater than 30 GB in Azure region where the Cloudbrink connector image needs to be downloaded is recommended to speed up transfer times, however it is not required
- Azure Cli package must be installed & configured on the hosting Linux VM, or your local machine
- A Storage Account service with container in the same region where the image needs to be created

Connector VM Requirements

- OS: Ubuntu 20.04
- Compute: 4 CPU and 8GB RAM
- Networking:
  - Inbound port 22 for SSH and active-standby keepalive
  - Outbound TCP port 443, and UDP ports 9993 and 9994 to Cloudbrink SaaS and Edges
- Username: `cbrink` (required for hardening scripts)
- Custom Data cloud-init script

Create Connector VM from VHD file

1. Download the Cloudbrink Connector managed disk snapshot from the following URL (Azure Public Storage Account) using **curl** command.

   Managed Disk Image zip file download link url
   https://distrouswest.blob.core.windows.net/distro-uswest/connector-v11.8.0.vhd.zip?sp=r&st=2022-01-27T17:02:12Z&se=2022-02-10T18:29:59Z&spr=https&sv=2020-08-04&sr=b&sig=Nt3c9xD%2BSJTkYLKBLOsmX8LZUePw9IFR%2FbzQeVBHvyc%3D

   Command:
   ```
   $ curl -l <connagent-image-download-url> -o <MANAGED-DISK-IMAGE-VERSION.vhd.zip>
   --retry 999 --retry-max-time 0 -C -
   ```

   Example:
   ```
   $ curl -l "https://distrouswest.blob.core.windows.net/distro-uswest/connector-v11.8.0.vhd.zip?sp=r&st=2022-01-27T17:02:12Z&se=2022-02-10T18:29:59Z&spr=https&sv=2020-08-04&sr=b&sig=Nt3c9xD%2BSJTkYLKBLOsmX8LZUePw9IFR%2FbzQeVBHvyc%3D" -o connector-v11.8.0.vhd.zip --retry 999 --retry-max-time 0 -C -
   ```

2. After the successful download, deflate/unzip the compressed zip archive to get the managed disk snapshot file. **Note: this creates a 30 GB disk on the host machine.**

   Command:
   ```
   $ unzip <zip-archive-file-name>
   ```

   Example:
   ```
   $ unzip connector-v11.8.0.vhd.zip
   ```

3. Upload the deflated managed disk image VHD file to the container in the existing storage account using Azure CLI.

   Command:
   ```
   $ az storage blob upload --account-name <qos-storage-account-name> --container-name
   <qos-container-name> --name <file-name-for-uploaded-file>
   ```

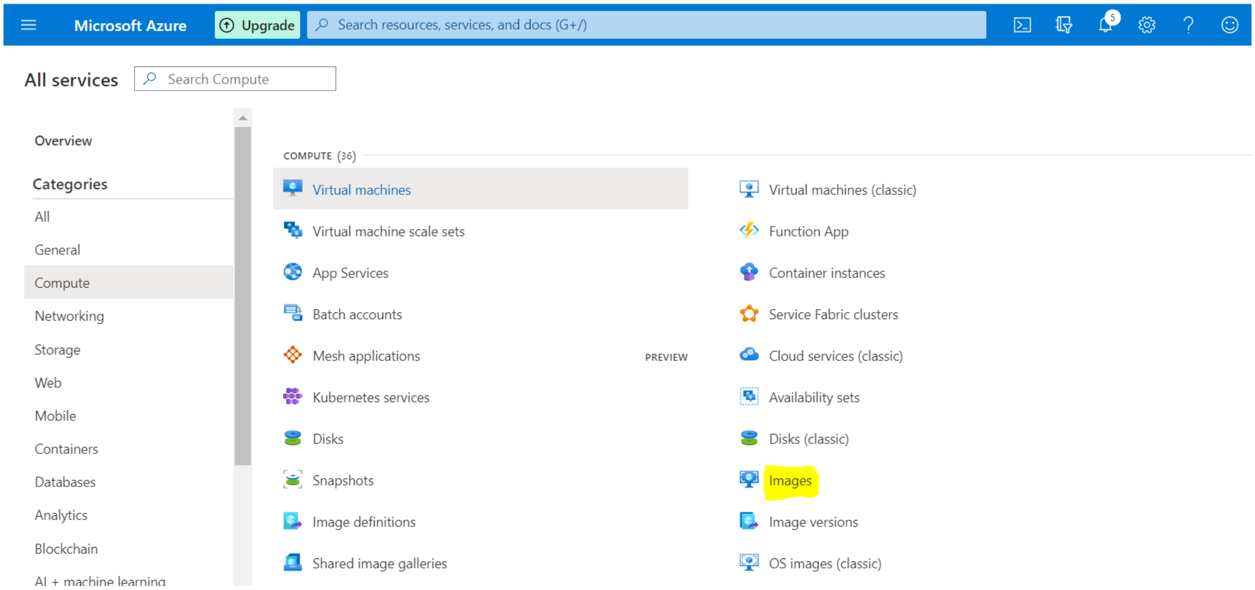
   Example:
   ```
   $ az storage blob upload --account-name blobuploadtest --container-name testcontainer --name connector-v11.8.0.vhd --file connagent-prod-v052.vhd
   ```

4. Create an image from the uploaded VHD file in the required region using Azure CLI.

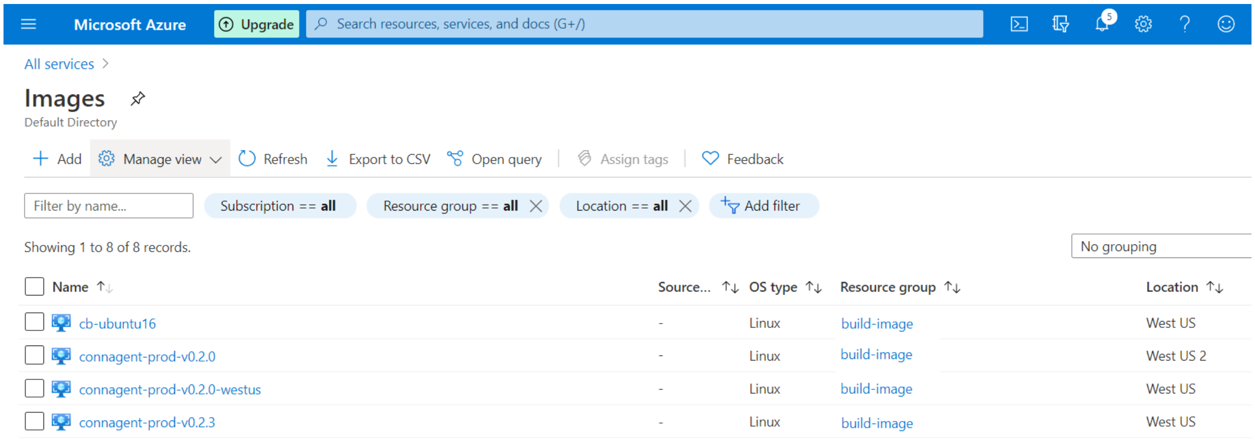
   Command:
   ```
   $ az image create --resource-group <qos-resource-group> --name <image-name> --source
   <qos-sa-blob-url> --location <required-region> --os-type Linux
   ```

   Example:
   ```
   $ az image create --resource-group ven-test-rg --name connector-v11.8.0.vhd --source
   https://blobuploadtest.blob.core.windows.net/testcontainer/connector-v11.8.0.vhd
   --location eastus2 --os-type Linux
   ```

5. In Azure portal, navigate to **All Services > Compute> Images**.



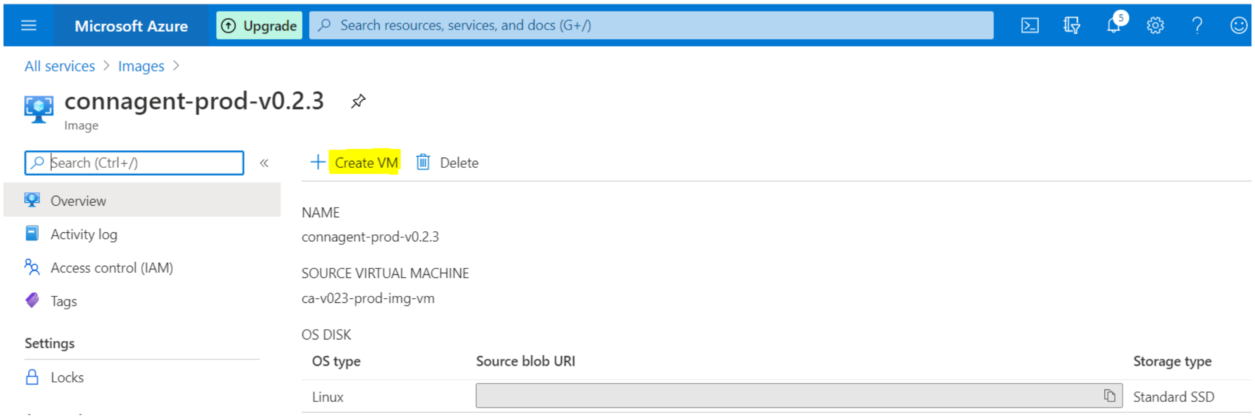
6. On the **Images** page, select the newly created image and navigate to its overview page.



Showing 1 to 8 of 8 records.

□ Name ↑↓	Source... ↑↓	OS type ↑↓	Resource group ↑↓	Location ↑↓
□ cb-ubuntu16	-	Linux	build-image	West US
□ connagent-prod-v0.2.0	-	Linux	build-image	West US 2
□ connagent-prod-v0.2.0-westus	-	Linux	build-image	West US
□ connagent-prod-v0.2.3	-	Linux	build-image	West US

7. Click on the **Create VM** link located in the header to create a new connector VM.



connagent-prod-v0.2.3

Image

NAME

connagent-prod-v0.2.3

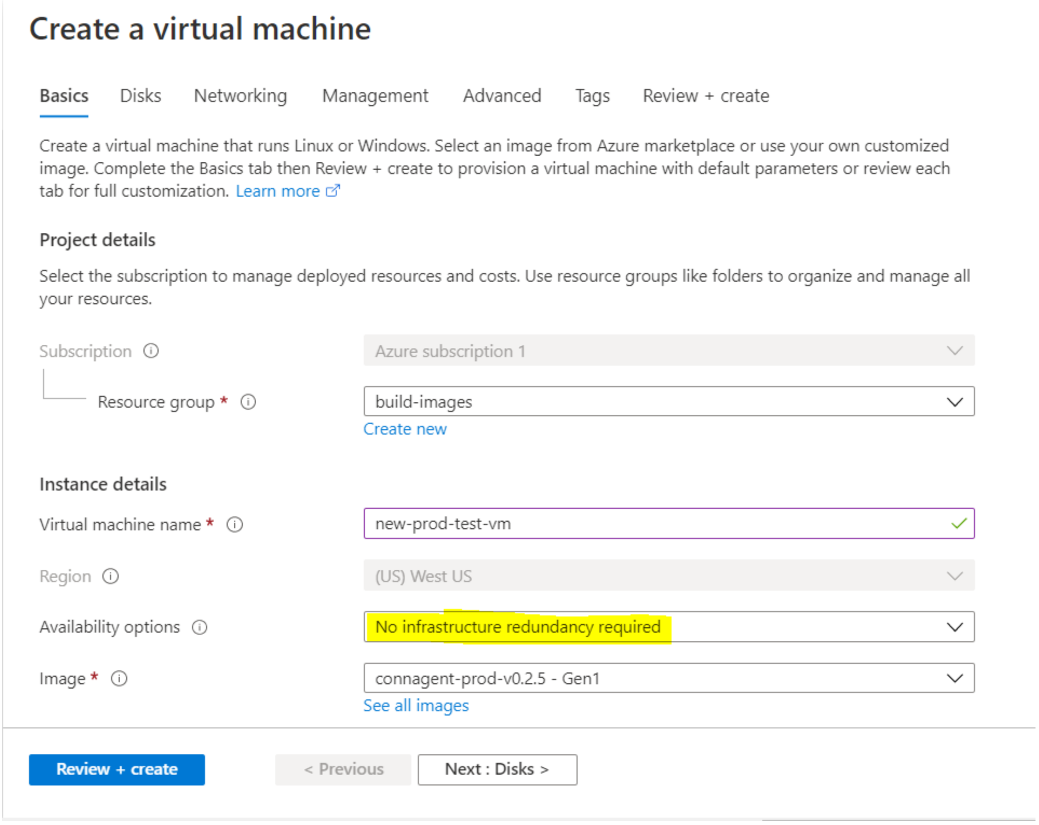
SOURCE VIRTUAL MACHINE

ca-v023-prod-img-vm

OS DISK

OS type	Source blob URI	Storage type
Linux		Standard SSD

8. Within the **Instance Details** section of the **Basics** tab, ensure the **Availability Options** has "No infrastructure redundancy required" selected.



Create a virtual machine

Create a virtual machine that runs Linux or Windows. Select an image from Azure marketplace or use your own customized image. Complete the Basics tab then Review + create to provision a virtual machine with default parameters or review each tab for full customization. Learn more

Project details

Select the subscription to manage deployed resources and costs. Use resource groups like folders to organize and manage all your resources.

Subscription: Azure subscription 1

Resource group *: build-images

Instance details

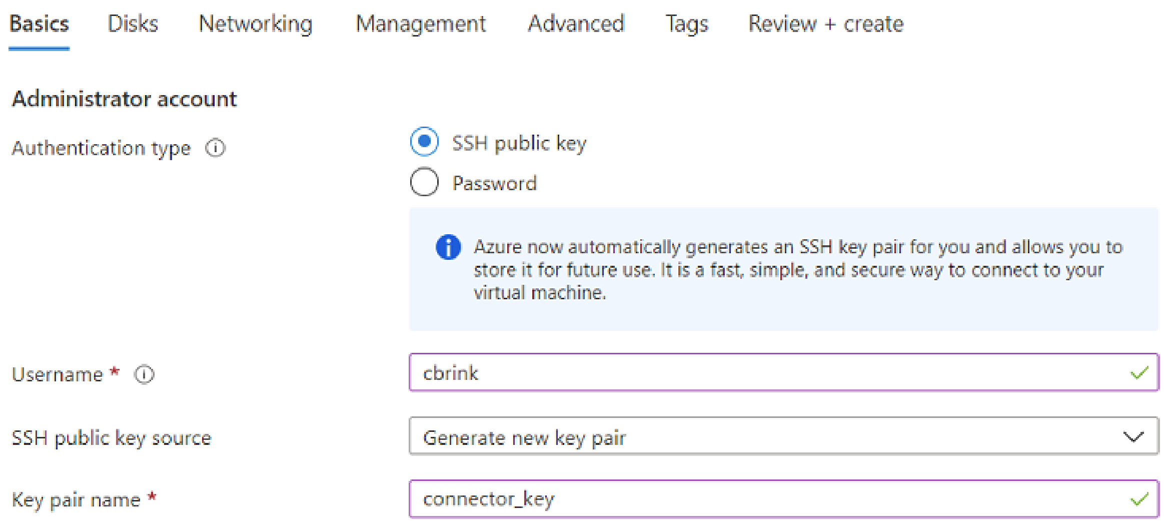
Virtual machine name *: new-prod-test-vm

Region: (US) West US

Availability options: No infrastructure redundancy required

Image *: connagent-prod-v0.2.5 - Gen1

9. Within the **Administrator Account** section of the **Basics** tab, ensure the **Username** is set to **`cbrink`** (this is required for our hardening script).



Administrator account

Authentication type ⓘ ● SSH public key ○ Password

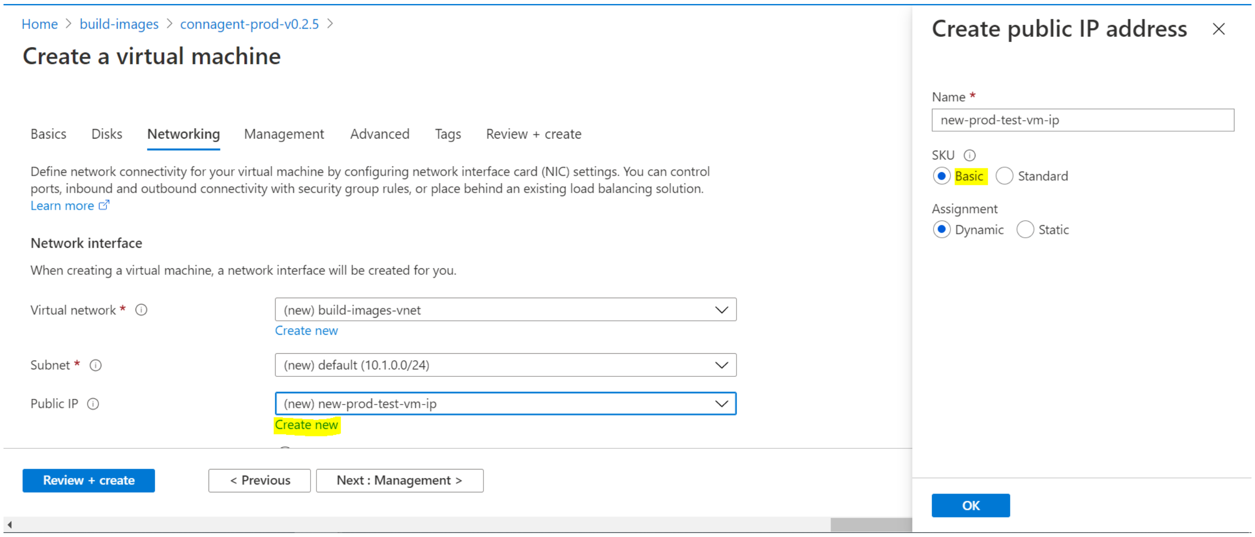
Azure now automatically generates an SSH key pair for you and allows you to store it for future use. It is a fast, simple, and secure way to connect to your virtual machine.

Username * ⓘ cbrink ✓

SSH public key source: Generate new key pair

Key pair name * connector_key ✓

10. In the **Networking** section of VM creation, ensure that the **Public IP** type is set to "Basic" to ensure the Cloudbrink Connector retrieves its public ip details from Azure's metadata.



Create a virtual machine

Define network connectivity for your virtual machine by configuring network interface card (NIC) settings. You can control ports, inbound and outbound connectivity with security group rules, or place behind an existing load balancing solution.

Network interface

When creating a virtual machine, a network interface will be created for you.

Virtual network * ⓘ (new) build-images-vnet

Subnet * ⓘ (new) default (10.1.0.0/24)

Public IP ⓘ (new) new-prod-test-vm-ip

Create public IP address

Name * new-prod-test-vm-ip

SKU ⓘ ● Basic ○ Standard

Assignment ● Dynamic ○ Static

11. In the **Advanced** tab under the **Custom-Data** section with the below script, substituting in the provided OTP value.

```
#cloud-config
runcmd:
 - [bash, /opt/scripts/brink_connector_deploy_cloud.sh, -o, "<OTP_VALUE>", -a, "1",
-v, "", -p, "", -f, ""]
```

cloudbrink
Software Defined Mobility