

- Use Managed Identities with CycleCloud
- Install Azure CycleCloud CLI
- Run CycleCloud in an Azure Container Instance
- Start a Cluster
- Use Managed Identities with CycleCloud
- Install Azure CycleCloud CLI
- Run CycleCloud in an Azure Container Instance
- Start a Cluster
- Connect to a Cluster Node
- Use Managed Identities with CycleCloud
- Install Azure CycleCloud CLI
- Run CycleCloud in an Azure Container Instance
- Start a Cluster
- Install Azure CycleCloud CLI
- Run CycleCloud in an Azure Container Instance
- Start a Cluster
- Connect to a Cluster Node
- CycleCloud
- Use Managed Identities with CycleCloud
- Install Azure CycleCloud CLI
- Run CycleCloud in an Azure Container Instance
- Start a Cluster
- Install Azure CycleCloud CLI
- Run CycleCloud in an Azure Container Instance
- Start a Cluster
- Connect to a Cluster Node
- Terminate a Cluster
- Work with Custom Images
- Install CycleCloud Manually
- Prepare Your Azure Subscription for CycleCloud
- Use Managed Identities with CycleCloud
- Install Azure CycleCloud CLI
- Run CycleCloud in an Azure Container Instance
- Start a Cluster
- Install Azure CycleCloud CLI
- Run CycleCloud in an Azure Container Instance
- Start a Cluster
- Connect to a Cluster Node
- Terminate a Cluster
- Work with Custom Images
- Use HB and HC VMs with CycleCloud
- Install Jetpack
- Enable Return Proxy
- Use Tags
- Use Images
- > Reference
- Download PDF

Supported Versions

Supported versions of the CycleCloud Container Image can be found in the product [dockerhub page](#). The image can be launched as an Azure Container instance (using existing resource group, location, and preferred container and dns names). CycleCloud has SSL certificate generation included, so if you specify the arguments twice (once for az cli and again to set environment variables), then the container is able to establish valid SSL certificates automatically.

```
sample Copy
#!/bin/bash
ResourceGroup="rg-name"
Location="westus2"
CIName="ci-name"
CIDNSName="ci-name"

az container create -g ${ResourceGroup} \
  --location ${Location} \
  --name ${CIName} \
  --dns-name-label1 ${CIDNSName} \
  --image mcr.microsoft.com/hpc/azure-cyclecloud \
  --ip-address public \
  --ports 80 443 \
  --cpu 2 --memory 4 \
  -e JAVA_HEAP_SIZE=2048
```

In the above example, the container and the cyclecloud UI will be available at [https://\\${CIDNSName}.\\${Location}.azurecontainer.io](https://${CIDNSName}.${Location}.azurecontainer.io).

Optionally, you can add an additional environment variable for the fully qualified domain name. In this case, CycleCloud will try to create valid a SSL certificate:

```
sample Copy
FQDN="https://${CIDNSName}.${Location}.azurecontainer.io"
...
-e JAVA_HEAP_SIZE=2048 FQDN=${FQDN}
```

Recover the SSH Keypair

CycleCloud creates a keypair to be used for administrative access to nodes. This keypair will be printed to the `stdout` of the container image, and should be retained.

A unique ssh keypair for the container appears in the standard output of the container process. Retain this keypair for admin access to the CycleCloud clusters. In the Azure Container Instance, this can be found in the **Container** menu under the **Logs** tab.

```
sample Copy
Private Key for admin access to nodes. Retain for cyclecloud cli and ssh access.
-----BEGIN RSA PRIVATE KEY-----
MIIEKAIBAKAgEAAhdf1rGEEsps2R+EZPSZq/2TLA/JQPNYwFtcTvA0cJ300
wRR/U8HdDswFpAvj2T00ptQqHf7prMB1/Sua1KFjYkJ/7AZxx13F-qWh3z14dDq
xmUPhQ1eZ9XPaIAYDewSeGibxuaFbkXmmxWscw1K9hXfwXng58Rs23Q/x4/xw08
FdcIvh7FjR6h13z0j6He0sRw7z0myRgJ88nPziwYB5pm9jykhNuw1YwYssSuDX
...
IFDYB41MRwK1jdxIs773U6JtuorWj5IbcIjxdK6YzayyTzJjw3ejEW12F6aSrMvs
W7d1Hj1Az0LMqNLV3LTTXxxK5d0BbExDyvE2KQe/6Wf9Z5FLAr8BcZe+PXPEX
mVa3tFI9Hfsz2qjsB1YLRFZY1MR+BzCI9uOyu9b1u2VLUX1fjg1DJ6XYtc0QAJP0
6ySHC9t1sZuh1aYHqvk0YU1LZeJch4BCzd9EknsccHxEJUFbF8CVjm1ZU=
-----END RSA PRIVATE KEY-----
```

Autoconfig

Autoconfig

To bring up an Azure CycleCloud container with a user pre-created, add the following environment variables to the docker run command:

```
Azure CLI Copy Try It
docker run -m 4G -p 80:80 -p 443:443 \
-e "JAVA_HEAP_SIZE=2048" \
-e CYCLECLOUD_AUTOCONFIG=true \
-e CYCLECLOUD_USERNAME=$YOUR-USER-NAME \
-e CYCLECLOUD_PASSWORD=$PASSWORD \
-e CYCLECLOUD_USER_PUB_KEY=$SSH_PUBLIC_KEY \
mcr.microsoft.com/hpc/azure-cyclecloud
```

With this, all clusters nodes started will have this user created and the SSH public key staged in their authorized_keys file. You can also login to the CycleCloud web interface using the username and password as credentials.

