

ARO-management

A collection of scripts to manage Azure Redhat Openshift (ARO) clusters

Azure Red Hat Openshift (ARO) Deployment Code / Scripts

createaro.bat

This windows based script will deploy Azure Red Hat OpenShift 4.x and create the necessary group/network infrastructure required. The process takes roughly 55 minutes. Until the 'az aro' command becomes GA within the Azure Linux CLI, you must ensure your Azure CLI has the extension included: **az extension add -n aro --index <https://az.aroapp.io/stable>** and continue to keep it updated: **az extension update -n aro --index <https://az.aroapp.io/stable>**

delaro.bat

This windows based script will remove the existing ARO

aro4-build.sh

This script will deploy Azure Red Hat OpenShift 4.x and create the necessary group/network infrastructure required. The process takes roughly 35 minutes. Until the 'az aro' command becomes GA within the Azure Linux CLI, you must ensure your Azure CLI has the extension included: **az extension add -n aro --index <https://az.aroapp.io/stable>** and continue to keep it updated: **az extension update -n aro --index <https://az.aroapp.io/stable>**

The usage is as follows:

./aro4-build.sh to create an ARO 4.x cluster with a standard aroapp.io domain.

./aro4-build.sh blah.foo.com to create an ARO 4.x cluster with a custom domain of blah.foo.com

Notes:

- This script now supports Red Hat Cloud Access pull secrets. Just save your pull secret in a file called "pull-secret.txt" in the same directory in which you invoke this script.
- Custom domains will error on an invalid SSL certificate since the certificate is self-signed. You will need to upload a signed SSL certificate for your domain to address this.

- The build script will look for the DNS Zone and A records for the custom domain. If either don't exist, it will create the zone and/or associated A records.
 - Using the example above, it will be your responsibility to create an NS record from the **foo.com** zone to point to **blah.foo.com**. The nameservers for **blah.foo.com** will be provided by the script during build.
-

aro4-aad-connect.sh

This script will connect Azure Red Hat OpenShift to Azure Active Directory. It will create a new Azure Application & Service Principal within AAD and subsequently configure an OAuth based Authentication Provider to bind to it using the subscription and tenant which are active when the script is run. The script is compatible with standard "aroapp.io" deployments and custom domains.

The usage is as follows:

./aro4-aad-connect.sh (ARO Cluster Name) (ARO Resource Group Name)

cleanappsp.sh

This script is a housekeeping script that mercilessly deletes all Azure Active Directory applications and service principals with an **aro-** prefix. It is particularly useful when standing up / tearing down multiple ARO clusters.

The usage is as follows:

./cleanappsp.sh

aro4-replace-pull-secret.sh

This script will allow you to replace the global pull secret currently existing on your ARO cluster.

Pre-requisites:

- Logged in a cluster-admin user (kubeadmin) with oc
- Obtain a revised pull secret (ex. from cloud.redhat.com) and save to a text file

The usage is as follows:

./aro4-replace-pull-secret.sh (filename-of-pull-secret.json)

Note: This script assumes that you are logged in as kubeadmin - if you have created a new cluster-admin user, you will need to change the script to reflect this. Your cluster will also become unavailable for several minutes while the revised pull-secret is propogated across all nodes.
