LDAP Group Sync

(Simon Reap <[sreap@redhat.com](mailto:sreap@redhat.com)>, 19th December 2019)

# Overview

LDAP Group Sync is used to manage OpenShift groups, by ensuring that if a user is a member of a specific group in Active Directory, then they are in the corresponding group in OpenShift. These groups are then used to assign permissions and privileges to those users.

The code used below is stored in https://xescm.xeop.de/openshift/group-ldap-sync

# Permissions used

The Project Provisioner assigns specified roles (often admin, edit and view) to specific groups when it creates a project. For example, when the hedgehog-acceptance project was created, the “hedgehog-act-deployer” OpenShift group was given “edit” permissions on that Project. If a user is a member of the “xeops-ocp-hedgehog-act-deployer” group in AD, LDAP Group Sync will make them members of the “hedgehog-act-deployer” group in OpenShift, so they will have “edit” permission on that Project.

Separately, the cluster-admin ClusterRole has been assigned to the atc-admin group using the following command:

$ oc adm policy add-cluster-role-to-group cluster-admin act-admin

# Group Sync control files

We have used the Whitelist approach for this process. A file contains a full list of all of the groups which should be transferred. This includes the xeops-ocp-act-admin group, which contains all of the system administrators for the cluster. It also includes all of the groups which give specific permissions to specific projects (here, admin, deployer and operator groups have admin, edit and view roles in a project).

The connection to the AD is by an insecure connection, since the machines on which it runs may not have the CA for the LDAP server (and it is not possible to ignore certificate errors using this tool).

Note that different whitelist and ldapsync files will be required for different sets of clusters. For example, the list of groups to be created will be different in the Acceptance and Production clusters.

The whitelist.txt file is currently as follows (a couple of lines have split because they are so long). Note that the xeops-ocp-act-users group is not included - that is to be used only for authentication - these groups are concerned with authorisation:

CN=xeops-ocp-act-admin,OU=XEOPS,OU=Appl Groups,OU=Accounts,DC=oa,DC=pnrad,DC=net

CN=xeops-ocp-hedgehog-act-admin,OU=XEOPS,OU=Appl Groups,OU=Accounts,DC=oa,DC=pnrad,DC=net

CN=xeops-ocp-hedgehog-act-deployer,OU=XEOPS,OU=Appl Groups,OU=Accounts,DC=oa,DC=pnrad,DC=net

CN=xeops-ocp-hedgehog-act-operator,OU=XEOPS,OU=Appl Groups,OU=Accounts,DC=oa,DC=pnrad,DC=net

CN=xeops-ocp-scs-act-admin,OU=XEOPS,OU=Appl Groups,OU=Accounts,DC=oa,DC=pnrad,DC=net

CN=xeops-ocp-scs-act-deployer,OU=XEOPS,OU=Appl Groups,OU=Accounts,DC=oa,DC=pnrad,DC=net

CN=xeops-ocp-scs-act-operator,OU=XEOPS,OU=Appl Groups,OU=Accounts,DC=oa,DC=pnrad,DC=net

The LDAP sync control file ldapsync.yaml is (again, the "CN=xeops-ocp-" lines have wrapped):

kind: LDAPSyncConfig

apiVersion: v1

url: ldap://FRA-LDAP-OAAD.oa.pnrad.net

bindDN: "CN=xeops bind,OU=Technical Accounts,OU=Accounts,DC=oa,DC=pnrad,DC=net"

bindPassword: "KlOJ5OZhK7\*1\_a"

insecure: true

groupUIDNameMapping:

"CN=xeops-ocp-act-admin,OU=XEOPS,OU=Appl Groups,OU=Accounts,DC=oa,DC=pnrad,DC=net": act-admin

"CN=xeops-ocp-hedgehog-act-admin,OU=XEOPS,OU=Appl Groups,OU=Accounts,DC=oa,DC=pnrad,DC=net": hedgehog-act-admin

"CN=xeops-ocp-hedgehog-act-deployer,OU=XEOPS,OU=Appl Groups,OU=Accounts,DC=oa,DC=pnrad,DC=net": hedgehog-act-deployer

"CN=xeops-ocp-hedgehog-act-operator,OU=XEOPS,OU=Appl Groups,OU=Accounts,DC=oa,DC=pnrad,DC=net": hedgehog-act-operator

"CN=xeops-ocp-scs-act-admin,OU=XEOPS,OU=Appl Groups,OU=Accounts,DC=oa,DC=pnrad,DC=net": scs-act-admin

"CN=xeops-ocp-scs-act-deployer,OU=XEOPS,OU=Appl Groups,OU=Accounts,DC=oa,DC=pnrad,DC=net": scs-act-deployer

"CN=xeops-ocp-scs-act-operator,OU=XEOPS,OU=Appl Groups,OU=Accounts,DC=oa,DC=pnrad,DC=net": scs-act-operator

activeDirectory:

groupsQuery:

derefAliases: never

pageSize: 0

groupUIDAttribute: dn

groupNameAttributes: [ cn ]

groupMembershipAttributes: [ memberOf ]

usersQuery:

baseDN: "OU=Users,OU=Sync,OU=Accounts,DC=oa,DC=pnrad,DC=net"

scope: sub

derefAliases: never

filter: (objectClass=organizationalPerson)

pageSize: 0

userNameAttributes: [ sAMAccountName ]

tolerateMemberNotFoundErrors: false

tolerateMemberOutOfScopeErrors: false

# Group sync process

To run the LDAP Group Sync process, be in the directory with these two files,

oc login as a cluster administrator

Run the following command to check that the process is working. It will list out the groups and their members:

$ oc adm groups sync --whitelist=whitelist.txt --sync-config=ldapsync.yaml

This will report errors if, for example, one or more of the groups was not created by this process (e.g. a group was created manually), or if the LDAP server being used changes. If that happens, delete the group(s) it complains about (oc delete group <group\_name>) and run the command again. If you do delete groups, you will need to add roles to the groups again (the above command for the act-admin group, and the Project Provisioner for the project-related permissions)

If the call works, you can add the --confirm flag to update the groups

$ oc adm groups sync --whitelist=whitelist.txt --sync-config=ldapsync.yaml --confirm

You can verify the roles assigned to the act-admin group by issuing the following command (again, as a Cluster Administrator) and getting a line similar to that shown (the name at the start of the line may be different) :

* $ oc get clusterrolebindings | grep act-admin  
  cluster-admin-3 /cluster-admin act-admin

You can verify the roles in projects by the following command (the “-n” parameter is the project you are interested in). Some internal roles have been omitted. Here the usual three groups were given admin, edit and view permissions, and a single extra user was given administrator permission.

$ oc get rolebindings -n hedgehog-acceptance

NAME ROLE USERS GROUPS

hedgehog-act-admin-admin /admin hedgehog-act-admin

hedgehog-act-deployer-edit /edit hedgehog-act-deployer

hedgehog-act-operator-view /view hedgehog-act-operator

Kv082-admin /admin kv082

# Automated processing

If the above process is to be automated, an OpenShift service account will need to be created, to allow token-based login in, for example, a cron job.

The service account must have “get”, “list”, “create” and “update” verbs on the “groups” resource in the “user.openshift.io” apiGroup.

An alternative approach, using a POD to make the “oc adm groups sync” call, is documented in <https://github.com/redhat-cop/openshift-management/tree/master/jobs>, under cronjob-ldap-group-sync.yml, though some changes would have to be made to the configuration file to reflect the Active Directory structure used here.