# Prerequisites for K3s AWX Setup

To successfully set up AWX using K3s, the following virtual machine (VM) requirements must be provisioned:

4vms for awx kubernetes

1vm for postgres sql

Hosts details:

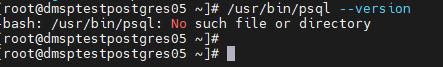
|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **FQDN** | **Static IP's** | **CPU** | **RAM** | **HDD** | **VLAN** | **OS** |
| dmsptestawx01.idm.dev.entrust.com | 10.1.214.7 | 4 | 16G | 58G | [VLAN514](javascript:void(0)) | RHEL9.5 |
| dmsptestawx02.idm.dev.entrust.com | 10.1.214.8 | 4 | 16G | 58G | [VLAN514](javascript:void(0)) | RHEL9.5 |
| dmsptestawx03.idm.dev.entrust.com | 10.1.214.9 | 4 | 16G | 58G | [VLAN514](javascript:void(0)) | RHEL9.5 |
| dmsptestawx04.idm.dev.entrust.com | 10.1.214.16 | 4 | 16G | 58G | [VLAN514](javascript:void(0)) | RHEL9.5 |
| dmsptestpostgres05.idm.dev.entrust.com | 10.1.214.57 | 4 | 16G | 58G | [VLAN514](javascript:void(0)) | RHEL9.5 |

# PostgreSQL Installation

we can install postgres sql on **dmsptestpostgres05** server

check which version is there in postgres server

[root@ dmsptestpostgres05 ~]# /usr/bin/psql --version



If postgres version 13 is not there run below commands

[root@dmsptestpostgres05 ~]# dnf remove postgres\\*

A screen shot of a computer

AI-generated content may be incorrect.

[root@dmsptestpostgres05 ~]# dnf install <https://download.postgresql.org/pub/repos/yum/reporpms/EL-9-x86_64/pgdg-redhat-repo-latest.noarch.rpm>

A screenshot of a computer

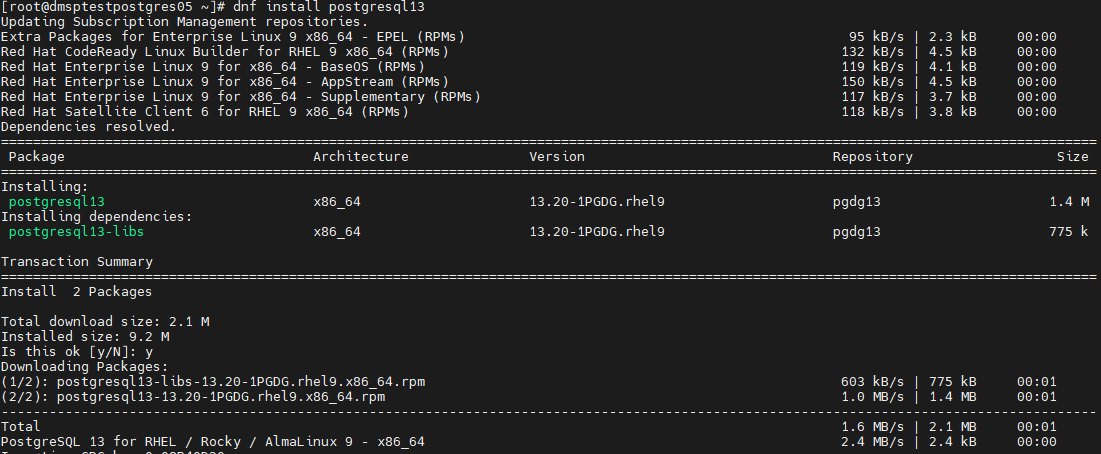
AI-generated content may be incorrect.

[root@dmsptestpostgres05 ~]# dnf -qy module disable postgresql

A screenshot of a computer

AI-generated content may be incorrect.

[root@dmsptestpostgres05 ~]# dnf install postgresql13



[root@dmsptestpostgres05 ~]# dnf install postgresql13-server

A screenshot of a computer

AI-generated content may be incorrect.

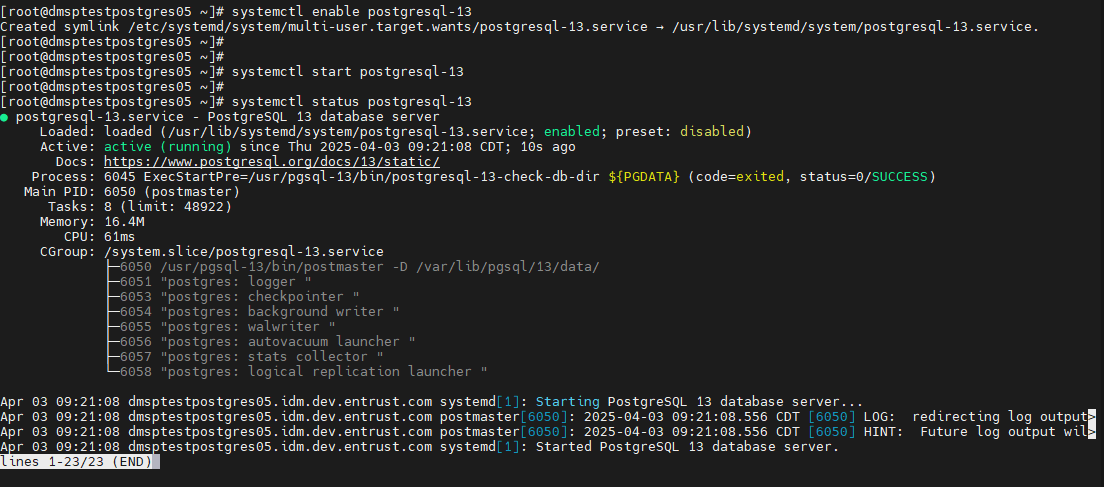
[root@dmsptestpostgres05 ~]# /usr/pgsql-13/bin/postgresql-13-setup initdb

A black screen with white text

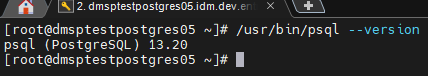
AI-generated content may be incorrect.

[root@dmsptestpostgres05 ~]# systemctl enable postgresql-13

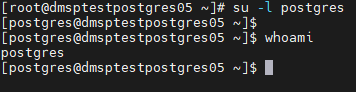
[root@dmsptestpostgres05 ~]# systemctl start postgresql-13



[root@dmsptestpostgres05 ~]# /usr/bin/psql --version

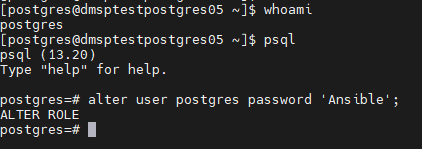


[root@dmsptestpostgres05 ~]# su -l postgres



[postgres@dmsptestpostgres05 ~]$ psql

postgres=# alter user postgres password 'Ansible';



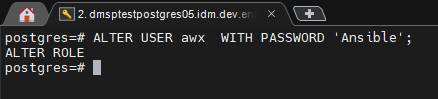
postgres=# CREATE USER awx;



postgres=# ALTER USER awx WITH LOGIN ENCRYPTED PASSWORD 'XXXXXXXXX';

or

ALTER USER awx WITH PASSWORD 'Ansible';



postgres=# CREATE DATABASE awx;



postgres=# GRANT ALL PRIVILEGES ON DATABASE awx TO awx;

A screenshot of a computer

AI-generated content may be incorrect.

postgres=# ALTER USER awx WITH SUPERUSER;

A black screen with white text

AI-generated content may be incorrect.

postgres=# \du

A screenshot of a computer screen

AI-generated content may be incorrect.

postgres=# \q >>>>>>>>>> quit from postgres and exit

[postgres@dmsptestpostgres05 ~]# exit

A black background with white text

AI-generated content may be incorrect.

[root@dmsptestpostgres05 ~]#

[root@dmsptestpostgres05 ~]# cd /var/lib/pgsql/13/data

A screen shot of a computer code

AI-generated content may be incorrect.

[root@dmsptestpostgres05 data]# vi pg\_hba.conf

UPDATE BELOW Lines end of the file

host replication all ::1/128 ident

host all all 0.0.0.0/0 md5

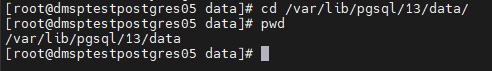
host all all 10.1.214.0 255.255.255.0 trust

A black screen with white text

AI-generated content may be incorrect.

And then save it

[root@dmsptestpostgres05 data]# cd /var/lib/pgsql/13/data



[root@dmsptestpostgres05 data]# vi postgresql.conf

update below lines

listen\_addresses = '\*'

port = 5432

max\_connections = 150

A screen shot of a computer

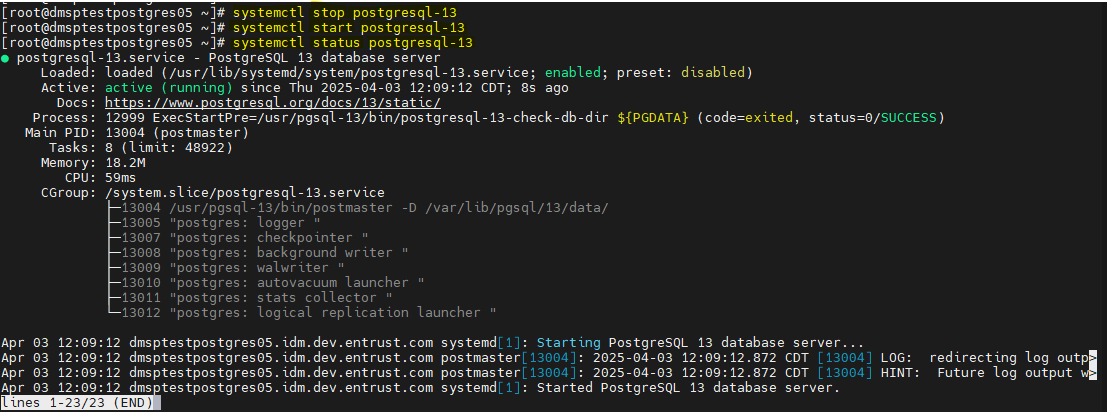
AI-generated content may be incorrect.

And then save it

[root@dmsptestpostgres05 ~]# systemctl stop postgresql-13

[root@dmsptestpostgres05 ~]# systemctl start postgresql-13

[root@dmsptestpostgres05 ~]# systemctl status postgresql-13



It is end for postgres db configuration

# AWX kuberntes setup

Login to dmsptestawx01, dmsptestawx02, dmsptestawx03 and dmsptestawx4 servers run the below commands

1st server

[root@ dmsptestawx01 ~]$ sudo su –

[root@ dmsptestawx01 ~]$ dnf -y update

[root@ dmsptestawx01 ~]$ systemctl disable firewalld --now

[root@ dmsptestawx01 ~]$ sudo reboot

2nd server

[root@ dmsptestawx02 ~]$ sudo su –

[root@ dmsptestawx02 ~]$ dnf -y update

[root@ dmsptestawx02 ~]$ systemctl disable firewalld --now

[root@ dmsptestawx02 ~]$ sudo reboot

3rd server

[root@ dmsptestawx03 ~]$ sudo su –

[root@ dmsptestawx03 ~]$ dnf -y update

[root@ dmsptestawx03 ~]$ systemctl disable firewalld --now

[root@ dmsptestawx03 ~]$ sudo reboot

4th server

[root@ dmsptestawx04 ~]$ sudo su –

[root@ dmsptestawx04 ~]$ dnf -y update

[root@ dmsptestawx04 ~]$ systemctl disable firewalld --now

[root@ dmsptestawx04 ~]$ sudo reboot

######################################################################

# Login 1st server dmsptestawx01 servers run below commands

[adityam@ dmsptestawx01 ~]$ sudo su –

[root@ dmsptestawx01 ~]# MYSECRET=Zj87KlmpqUUgh1

[root@ dmsptestawx01 ~]# curl -fL https://get.k3s.io | K3S\_TOKEN=${MYSECRET} \

sh -s - server --cluster-init

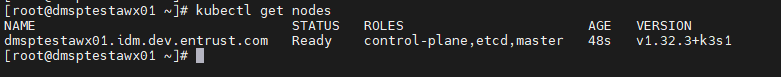
A screen shot of a computer

AI-generated content may be incorrect.

A screen shot of a computer

AI-generated content may be incorrect.

[root@ dmsptestawx01 ~]# kubectl get nodes



[root@ dmsptestawx01 ~]# setenforce 0

[root@ dmsptestawx01 ~]# sed -i 's/^SELINUX=.\*/SELINUX=permissive/g' /etc/selinux/config

[root@ dmsptestawx01 ~]# cat /etc/selinux/config | grep SELINUX=

A screen shot of a computer

AI-generated content may be incorrect.

# Login 2nd server dmsptestawx02 server run below commands

[adityam@ dmsptestawx02 ~]$ sudo su –

[root@dmsptestawx02 ~]# MYSECRET=Zj87KlmpqUUgh1

[root@dmsptestawx02 ~]# curl -fL https://get.k3s.io | K3S\_TOKEN=${MYSECRET} \

sh -s - server --server https://10.1.214.131:6443

10.1.214.131 IP is dmsptestawx01 server IP

A screenshot of a computer screen

AI-generated content may be incorrect.

[root@dmsptestawx02 ~]# kubectl get nodes

A computer screen shot of a computer code

AI-generated content may be incorrect.

[root@dmsptestawx02 ~]# setenforce 0

[root@dmsptestawx02 ~]# sed -i 's/^SELINUX=.\*/SELINUX=permissive/g' /etc/selinux/config

[root@dmsptestawx02 ~]# cat /etc/selinux/config | grep SELINUX=

A black screen with white text

AI-generated content may be incorrect.

# Login 3rd server dmsptestawx03 server run below commands

[adityam@dmsptestawx03 ~]$ sudo su –

[root@ dmsptestawx03~]# MYSECRET=Zj87KlmpqUUgh1

[root@dmsptestawx03 ~]# curl -fL https://get.k3s.io | K3S\_TOKEN=${MYSECRET} \

sh -s - server --server https://10.1.214.131:6443

10.1.214.131 IP is dmspawxXX1 server IP

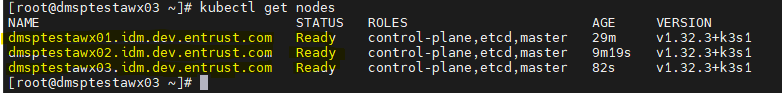
A screenshot of a computer

AI-generated content may be incorrect.

A computer screen shot of a black screen

AI-generated content may be incorrect.

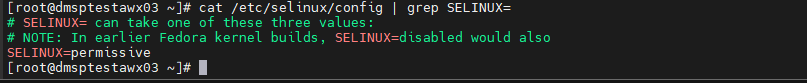
[root@dmsptestawx03 ~]# kubectl get nodes



[root@dmsptestawx03 ~]# setenforce 0

[root@dmsptestawx03 ~]# sed -i 's/^SELINUX=.\*/SELINUX=permissive/g' /etc/selinux/config

[root@dmsptestawx03 ~]# cat /etc/selinux/config | grep SELINUX=



# login 4th server dmsptestawx04 server run below commands

[adityam@dmstestawx04 ~]$ sudo su –

[root@dmsptestawx04 ~]# MYSECRET=Zj87KlmpqUUgh1

[root@dmsptestawx04 ~]# curl -fL https://get.k3s.io | K3S\_TOKEN=${MYSECRET} \

sh -s - server --server https://10.1.214.131:6443

10.1.214.131 IP is dmspawxXX1 server IP

[root@dmsptestawx04 ~]# kubectl get nodes

A screen shot of a computer screen

AI-generated content may be incorrect.

[root@dmsptestawx04 ~]# setenforce 0

[root@dmsptestawx04 ~]# sed -i 's/^SELINUX=.\*/SELINUX=permissive/g' /etc/selinux/config

[root@dmsptestawx04 ~]# cat /etc/selinux/config | grep SELINUX=

A screen shot of a computer code

AI-generated content may be incorrect.

######################################################################

# Login 1st server again dmsptestawx01 servers run the below commands

[adityam@ dmsptestawx01 ~]$ sudo su –

[root@dmsptestawx01 ~]# curl -sfL https://get.k3s.io | sudo bash –

A computer screen shot of a black screen

AI-generated content may be incorrect.

[root@dmsptestawx01 ~]# chmod 644 /etc/rancher/k3s/k3s.yaml

[root@dmsptestawx01 ~]# systemctl status k3s.service

A computer screen with many colorful text

AI-generated content may be incorrect.

[root@dmspawxXX1 ~]$ kubectl version –short

A black screen with white text

AI-generated content may be incorrect.

# Configure AWX Operator

Login 1st server dmsptestawx01 servers run below commands

[root@dmsptestawx01 ~]#

[root@dmsptestawx01 ~]# yum -y install git make

A computer screen shot of a black screen

AI-generated content may be incorrect.

[root@dmsptestawx01 ~]# git clone <https://github.com/ansible/awx-operator.git>

A computer screen with white text

AI-generated content may be incorrect.

[root@dmsptestawx01 ~]# export NAMESPACE=awx

[root@dmsptestawx01 ~]# kubectl create ns ${NAMESPACE}

[root@dmsptestawx01 ~]# kubectl config set-context --current --namespace=$NAMESPACE

A black screen with white text

AI-generated content may be incorrect.

[root@dmsptestawx01 ~]# cd awx-operator/

[root@dmsptestawx01 ~]# dnf -y install jq

A computer screen with text

AI-generated content may be incorrect.

[root@dmsptestawx01 ~]# RELEASE\_TAG=`curl -s https://api.github.com/repos/ansible/awx-operator/releases/latest | grep tag\_name | cut -d '"' -f 4`

[root@dmsptestawx01 ~]# echo $RELEASE\_TAG

OUR ENVIRONEMNT SUITABLE OPERTOR Version is 0.28.0

[root@dmsptestawx01 ~]# git checkout 0.28.0

[root@dmsptestawx01 ~]# export NAMESPACE=awx

A screen shot of a computer

AI-generated content may be incorrect.

[root@dmsptestawx01 ~]# make deploy

A screenshot of a computer screen

AI-generated content may be incorrect.

[root@dmsptestawx01 ~]# kubectl get pods -n awx

A screenshot of a computer

AI-generated content may be incorrect.

Create a file name public-static-pvc.yml and update below code and then save it

[root@dmsptestawx01 ~]# vi public-static-pvc.yml

---

apiVersion: v1

kind: PersistentVolumeClaim

metadata:

name: public-static-data-pvc

spec:

accessModes:

- ReadWriteOnce

storageClassName: local-path

resources:

requests:

storage: 50Gi

A screen shot of a computer

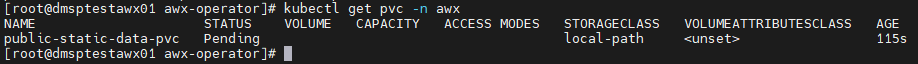
AI-generated content may be incorrect.

[root@dmsptestawx01 ~]# kubectl apply -f public-static-pvc.yml -n awx

A black background with white text

AI-generated content may be incorrect.

[root@dmsptestawx01 ~]# kubectl get pvc -n awx



Create a file name awx-postgres-secret-config.yml and update below code and then save it

host: "10.1.214.57" is postgres DB IP address

database: awx is postgres db name

username: awx is postgres db username

password: Ansible is postgres db pwd

[root@dmsptestawx01 awx-operator]# vi awx-postgres-secret-config.yml

---

apiVersion: v1

kind: Secret

metadata:

name: postgres-configuration

namespace: awx

stringData:

host: "10.1.214. 57"

port: "5432"

database: awx

username: awx

password: Ansible

sslmode: prefer

type: unmanaged

type: Opaque

# CERTS CONFIGURATION

Install Ansible AWX on RHEL 9 /

1) Getting Certs from Ashish/CORP security team for connecting to git ,satellite, ldap and ingress(https)

2) Cert names: tls.key, tls.cert and bundle\_awx\_mso\_g3\_priv\_satellite.pem

3) And copy above certs to awx-operator directory

4) And then run below commands

kubectl create secret generic awx-custom-satellite-certs --from-file=bundle-ca.crt=./bundle\_awx\_mso\_g3\_priv\_satellite.pem -n awx

kubectl create secret tls awx-ingress-tls-secret --cert=./tls.crt --key=./tls.key -n awx

###############################################################################################################################################

Create a file name awx-instance-deployment.yml and update below code and then save it

hostname: itawx.idm.dev.entrust.com is the host name of the awx console

if we wont create and excute certs commands comment out this line

#bundle\_cacert\_secret: awx-custom-satellite-certs

[root@dmsptestawx01 awx-operator]# vi awx-instance-deployment.yml

---

apiVersion: awx.ansible.com/v1beta1

kind: AWX

metadata:

name: awx

spec:

replicas: 1

service\_type: nodeport

ingress\_type: ingress

ingress\_tls\_secret: awx-ingress-tls-secret

hostname: itawx.idm.dev.entrust.com

projects\_persistence: true

projects\_storage\_access\_mode: ReadWriteOnce

web\_extra\_volume\_mounts: |

- name: static-data

mountPath: /var/lib/projects

extra\_volumes: |

- name: static-data

persistentVolumeClaim:

claimName: public-static-data-pvc

postgres\_configuration\_secret: postgres-configuration

bundle\_cacert\_secret: awx-custom-satellite-certs

[root@dmsptestawx01 awx-operator]#

# RUN BELOW KUBRNETES COMMANDS

[root@dmsptestawx01 awx-operator]# kubectl apply -f awx-postgres-secret-config.yml -n awx

[root@dmsptestawx01 awx-operator]# kubectl apply -f awx-instance-deployment.yml -n awx

[root@dmsptestawx01 awx-operator]# watch kubectl get pods -l "app.kubernetes.io/managed-by=awx-operator" -n awx

[root@dmsptestawx01 awx-operator]# kubectl logs -f deployments/awx-operator-controller-manager -c awx-manager

[root@dmsptestawx01 awx-operator]# chmod -R 777 /var/lib/rancher/k3s/storage/\*

[root@dmsptestawx01 awx-operator]# kubectl get deploy

OPTIONAL STEPS , ITS NOT MANDATORY TO RUN BELOW COMMANDS

######################################################################################

[root@dmsptestawx01 ~ ]# kubectl -n awx logs deploy/awx

[root@dmsptestawx01 ~ ]# kubectl -n awx logs deploy/awx -c redis

[root@dmsptestawx01 ~ ]# kubectl -n awx logs deploy/awx -c awx-web

[root@dmsptestawx01 ~ ]# kubectl -n awx logs deploy/awx -c awx-task

[root@dmsptestawx01 ~ ]# kubectl -n awx logs deploy/awx -c awx-ee

Access AWX Container’s Shell

Here is how to access each container’s shell:

[root@dmsptestawx01 ~ ]# kubectl exec -ti deploy/awx -c awx-task -- /bin/bash

[root@dmsptestawx01 ~ ]# kubectl exec -ti deploy/awx -c awx-web -- /bin/bash

[root@dmsptestawx01 ~ ]# kubectl exec -ti deploy/awx -c awx-ee -- /bin/bash

[root@dmsptestawx01 ~ ]# kubectl exec -ti deploy/awx -c redis -- /bin/bash

######################################################################################

[root@dmsptestawx01 ~ ]# kubectl get service -n awx

GETTING AWX CONSOLE PASSWORD to run below command

[root@dmsptestawx01 ~ ]# kubectl -n awx get secret awx-admin-password -o go-template='{{range $k,$v := .data}}{{printf "%s: " $k}}{{if not $v}}{{$v}}{{else}}{{$v | base64decode}}{{end}}{{"\n"}}{{end}}'

USERNAME IS : admin

login to itawx.idm.dev.entrust.com console

username is : admn

pasword is : xxxxxxxxxx

After login to the console configure LDAP servers for AD Users login awx console

GO to Settings-->and the got LDAP Settings ----> update below code

LDAP Server URI:- ldaps://pmspad01.corporate.datacard.com:636

LDAP Bind DN :- CN=svc\_awxpatch,OU=Service Accounts,OU=Admin,DC=corporate,DC=datacard,DC=com

LDAP Bind Password : Enter svc\_awxpatch

LDAP User DN Template:- Not configured

LDAP Group Type:- ActiveDirectoryGroupType

LDAP Require Group:- CN=DL- Security Engineering,OU=DisGrp,DC=corporate,DC=datacard,DC=com

LDAP Deny Group:-Not configured

LDAP Start TLS:- Off

LDAP User Search:-

[

"DC=corporate,DC=datacard,DC=com",

"SCOPE\_SUBTREE",

"(sAMAccountName=%(user)s)"

]

LDAP Group Search:-

[

"DC=corporate,DC=datacard,DC=com",

"SCOPE\_SUBTREE",

"(objectClass=group)"

]

LDAP User Attribute Map:-

{

"email": "mail",

"first\_name": "givenName",

"last\_name": "sn"

}

LDAP Group Type Parameters:- {}

LDAP User Flags By Group:-

{

"is\_superuser": [

"CN=DL- Security Engineering,OU=DisGrp,DC=corporate,DC=datacard,DC=com"

]

}

LDAP Organization Map:-{}

LDAP Team Map-{}