**PED Preparation**

Prepare Tier0\_users.csv file

Graphical user interface, text, application

Description automatically generated

Run *GetUserInfoDN.ps1* against *Tier0\_users.csv*

A picture containing text

Description automatically generated

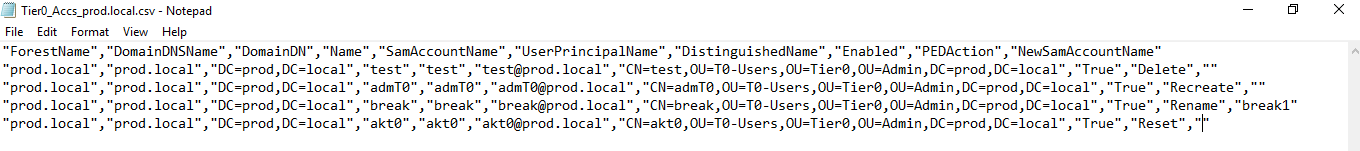
Result: 2 CSV files are created. For the accounts, the PED Action must be defined

* users: Tier0\_Accs\_prod.local.csvA picture containing diagram

  Description automatically generated
* group membership: Tier0\_Grp\_Membership\_prod.local.csv
* Text, letter

  Description automatically generated
* Possible PED actions are:
  + Delete
  + Disable
  + Recreate
  + Reset
  + Rename + new name
  + Deprivilege

Actions defined:



Run Prepare\_CSVs.ps1 pointing to the folder with PED actions files

Graphical user interface, text

Description automatically generated

Result: PED folder with all scripts and CSVs for running PED gets created

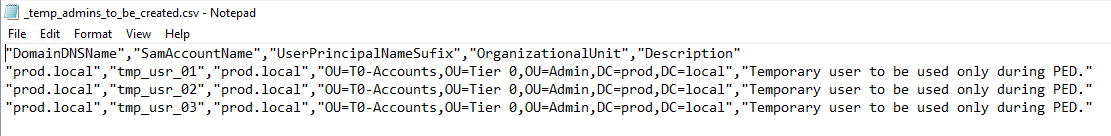
Define Reboot Plan in the CSV folder: “\_reboot\_plan.csv”

Copy PED folder to the root of PDC emulator

**PED Execution**

Create temporary PED EA/DA Admin accounts

Accounts to be created must be defined in the “\_temp\_admins\_to\_be\_created.csv” file in the CSV folder. Make sure the Organizational Unit exists! Otherwise, change it to the correct one!



By default, the temporary accounts are member of the following groups. This is defined in “\_temp\_admins\_grp\_membership.csv”

Text

Description automatically generated

Run CreateUsers.ps1 to create the temporary admin accounts

Text

Description automatically generated

Copy the file “New\_accounts.csv” in the Scripts folder to the CSV folder and name it as follows: “alphaorchid\_temp\_admins.csv”



Grant EA/DA admin rights to temporary admin accounts

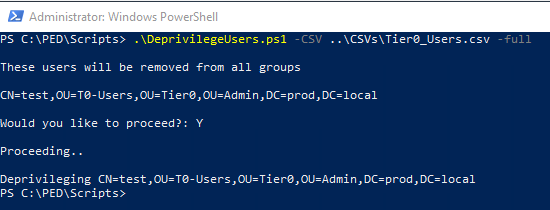


Enforce replication:

repadmin /syncall /AdeP

**Deprivilege old Tier 0 Admin accounts**

File used for that is “All\_Tier0\_Users.csv”. Copied to CSVs folder and modified for testing purposes so that only test user gets group memberships removed



**Disable old Tier 0 Admin accounts** (to be replaced with new Tier 0 accounts later)

.\DisableUsers.ps1 -csv ..\CSVs\\_will\_be\_permanently\_disabled.csv

.\DisableUsers.ps1 -csv ..\CSVs\\_will\_be\_renamed.csv

.\DisableUsers.ps1 -csv ..\CSVs\\_will\_be\_recreated.csv -rename

Text

Description automatically generated

A new OU gets created under domain root level: “OU=CR Disabled Accounts”.

The accounts defined in the csv files specified above, get disabled and moved to the newly created OU. However, they still have high-privilege group memberships.

Graphical user interface, text, application

Description automatically generated

**Disable old Admin accounts** (which have password reset later)

import-csv ..\CSVs\\_reset\_pass.csv | %{set-aduser $\_.name -Enabled:$false}



**KRBTGT Reset**

Execute First Kerberos Reset:

.\New-CtmADKrbtgtKeys.ps1

### WAIT! ###

Execute Second Kerberos Reset:

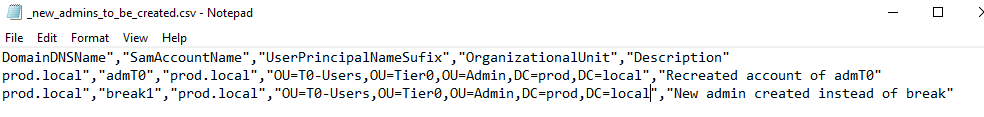
.\New-CtmADKrbtgtKeys.ps1

Restart All DCs according to Reboot plan.

.\RemoteServerRestart.ps1 -CSV ..\CSVs\\_reboot\_plan.csv

**Create and new Tier 0 admin accounts and issue passwords to administrators**

Make sure the OrganizationUnit is correct.



.\CreateUsers.ps1 -csv ..\CSVs\\_new\_admins\_to\_be\_created.csv



Both accounts are created in the Tier 0 OU.

Copy the file “New\_accounts.csv” in the Scripts folder to the CSV folder and name it as follows: “alphaorchid\_new\_admins.csv”

Grant Tier 0 Admin rights to new accounts

If the OUs, or groups are named differently, make sure to modify it in the “MakeItTier0.ps1” Script:

Graphical user interface, text, application

Description automatically generated



New accounts get added to the Tier0Admins group

Configure new Tier 0 accounts as "account sensitive and can't be delegated"

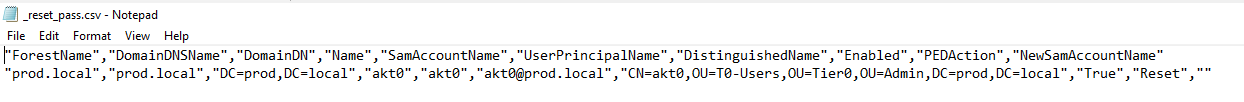
import-csv ..\CSVs\alphaorchid\_new\_admins.csv | %{set-aduser $\_.samaccountname -AccountNotDelegated:$true}



Grant admin rights to temporary admin accounts



**Reset passwords on defined accounts:**



.\ResetPasswordsFromFile.ps1 -csv ..\CSVs\\_reset\_pass.csv -showpass



Copy the file “New\_Passwords.csv” in the Scripts folder to the CSV folder and name it as follows: “alphaorchid\_pwdreset\_new\_admins.csv”

Copy-Item .\New\_Passwords.csv ..\CSVs\alphaorchid\_pwdreset\_new\_admins.csv



Enable accounts which had their password reset

import-csv ..\CSVs\\_reset\_pass.csv | %{set-aduser $\_.name -Enabled:$true}



**Cleanup / Remove all temporary accounts and permissions**

.\DeleteUsers.ps1 -csv ..\CSVs\alphaorchid\_temp\_admins.csv Graphical user interface, text, application

Description automatically generated