***Enabling public key authentication for Delphix Engine is strongly recommended for the processes documented in this document***.

Without public key authentication, Delphix will prompt for a password for each CLI command:

[CLI Cookbook: Configuring Key-Based SSH Authentication for Automation](https://docs.delphix.com/docs/developer-s-guide/command-line-interface-guide/cli-cookbook-common-workflows-tasks-and-examples/cli-cookbook-authentication-and-users/cli-cookbook-configuring-key-based-ssh-authentication-for-automation)

**DISABLING VDBs**

# Generate list with commands to disable VDBs

# DELETE VDBs that should not be disabled by adding the strings in egrep

echo "/source list display=name,virtual,config" | ssh admin@10.160.1.5 | grep true |egrep -v "dx|staging|AppFS|Postgres|ORA|MySQL" | awk '{print "/source; select "$1";disable; set type=SourceDisableParameters; commit"}' > /tmp/disable\_VDBs.out

# Edit the list of VDBs to confirm that the list is correct

vi /tmp/disable\_VDBs.out



#Run the command below to disable the VDBs

while read in; do echo "$in" | ssh admin@10.160.1.5; done < /tmp/disable\_VDBs.out

**CHANGING THE SQL SERVER OF THE VDBs**

# After disabling the VDBs, generate the commands to change the MSSQL server

# Change the value of the repository for the target server/instance

# DELETE the VDBs that should not be migrated by adding the strings in egrep

echo "/source list display=name,virtual,config" | ssh admin@10.160.1.5 | grep true |egrep -v "dx|staging|AppFS|Postgres|ORA|MySQL" | awk '{print "/sourceconfig; select "$1";update; set environmentUser=NEW\_HOST/user\_name;set repository=NEW\_HOST/NEW\_INSTANCE; commit"}' > /tmp/VDBs\_to\_migrate.out

# DELETE the VDBs that should not be migrated by adding the strings in egrepvi /tmp/ VDBs\_to\_migrate.out



**RE-enabling VDBs**

# Generate the commands to reactivate the VDBs:

echo "/source list display=name,virtual,config" | ssh admin@10.160.1.5 | grep true |egrep -v "dx|staging|AppFS|Postgres|ORA|MySQL" | awk '{print "/source; select "$1";enable; set type=SourceEnableParameters; commit"}' > /tmp/enable\_VDBs.out

# Edit the list to confirm that the commands are correct:

vi /tmp/enable\_VDBs.out

# Edit the list to confirm that the commands are correct:

while read in; do echo "$in" | ssh admin@10.160.1.5; done < /tmp/enable\_VDBs.out