**DB2 + DB3 User management scripts**

(WIP – Posting on BIT platform bible gitlab when complete)

db3docker- 10.25.1.150

db2security - 10.25.1.120

Itintern - password on keepass

adduser.sh to add users locally

runquerytxt. to create mysql logins

deleteuser.sh to delete

Run scripts with the appropriate csv

**Setup**

1. Check to see if there are any updates needed on the server. It’s good practice to patch a server before doing anything with it.
2. Firstly open the spreadsheet of students in excel & ensure there is a dummy line of data/headers in the first row and save as a csv (the script will **skip the first line** of the csv)
3. You will be logging in using the itintern account, the credentials of which can be found on Keepass (You can download the keepass file from Teams/files/ITPOperations.kdbx – double click this and enter our usual password x2 – please don’t hesitate to ask someone if you don’t know what this is)
4. WINSCP any relevant files over to the scripts folder on the server. (See Teams Files/Scripts for Load User/Scripts to load User/ )

**To load DB2 Users**

1. cd into the /Scripts folder and Gain root access with

*sudo su*

1. Any files that you are planning on executing will need their permissions modified.

*chmod +x addusers.sh*

1. To update users, run the DB2UserImport.sh script with the users list csv. If the file is a txt file, you can easily convert it using mv.

*./addusers.sh DB2ServerUsers.csv*

1. To verify success, check users have been created by examining /etc/passwd

*cat /etc/passwd*

1. Load mysql users with

*./runquerytxt.sh db2MySqlUsers.csv*

1. Try logging in with a selection of users locally, ideally the first, the last & one at random.
2. Repeat this step to test MariaDB access
3. Run the deletion script (as you will have changed the default password on the accounts you tested) & rerun the addusers.sh script.

*./deleteuser.sh db2ServerUsers.csv*

*./addusers.sh db2ServerUsers.csv*

**DB3 users**

You will need 2 files to perform this task. The list of users for DB3 and the script file to add the users.

You can find the DB3 user list @ Operations and Security > Files > Scripts to Load User > Scripts to Load User > 2020Sem2 > DB3users.txt

You can find the script file @ Operations and Security > Files > Scripts to Load User > Scripts to Load User > BashScripts > DB3load.sh

* Perform an update and upgrade on the system and reboot

*Sudo apt-get update*

*Sudo apt-get upgrade*

*Sudo reboot*

* WinSCP these files to 10.25.1.150
* Ensure script is executable

*Chmod 777 DB3load.sh*

* Change DB3users from .txt to .csv

*Mv DB3users.txt DB3users.csv*

* Add the users to the server

*./DB3load.sh DB3users.csv*

**Verify success**

* Verify user creation

*Cat /etc/passswd*

* Login to first, last and random user account to test that they’re working
* Choose an account, set a new password and log in
* Logged in as a user, run the following commands to check that the user can use Docker without escalated privileges – **IF YOU NEED TO RUN THESE COMMANDS USING SUDO, THEN THE USERLOAD SCRIPT HASN’T RUN SUCCESSFULLY. CHECK THAT THE DOCKER GROUP EXISTS AND USERS ARE ADDED TO THAT GROUP**
* Firstly, run the docker pull command to pull the latest server image

*docker pull mcr.microsoft.com/mssql/server:2019-latest*

* Secondly, run the docker run command to create an sql container. The command is below but it is easiest to copy and paste from the following link due to formatting errors that may occur - <https://docs.microsoft.com/en-us/sql/linux/quickstart-install-connect-docker?view=sql-server-ver15&pivots=cs1-bash> **STEP 2 UNDER “PULL AND RUN THE 2019 CONTAINER IMAGE”**

*docker run -e "ACCEPT\_EULA=Y" -e "SA\_PASSWORD=<YourStrong@Passw0rd>" \ -p 1433:1433 --name sql1 -h sql1 \ -d mcr.microsoft.com/mssql/server:2019-latest*

* View containers

*docker ps –a*

* Connect to container

*docker exec -it sql1 "bash"*

* Connect to sqlcmd

*/opt/mssql-tools/bin/sqlcmd -S localhost -U SA -P "<YourStrong@Passw0rd>"*

* Test functionality

*CREATE DATABASE TestDB*

*SELECT Name from sys.Databases*

*GO*