

Create User Linux and DB

Dit script wordt gebruikt om een of meer gebruikers aan te maken. Gebruiker wordt aangemaakt op Linux systeem en op de database. De gebruiker krijgt een database met de naam hetzelfde als de gebruikersnaam.

adduser.sh

```
#!/bin/bash
# Script to add a user to Linux system

DB_FILE=createDBUsers.tmp

addThisUser() {
if [ $(id -u) -eq 0 ]; then

    echo "" >> $DB_FILE

    if [ "$1" == "" ]; then
        read -p "Enter username : " username
    else
        username=$1
    fi
    if [ "$2" == "" ]; then
        read -s -p "Enter password : " password
    else
        password=$2
    fi

    egrep "^$username" /etc/passwd >/dev/null
    if [ $? -eq 0 ]; then
        echo "$username exists!"
        exit 1
    else
        U_HOME=/var/www/softwaredeveloper/$username
        U_GROUP=sftp
        U_SHELL=/bin/bash
```

```
pass=$(perl -e 'print crypt($ARGV[0], "password")' $password)
```

```
useradd -d $U_HOME -g $U_GROUP -m -s $U_SHELL -p $pass $username
```

```
[ $? -eq 0 ] && echo "User $username has been added to system!" || echo "Failed to add a user!"
```

```
chown $username $U_HOME
```

```
echo "<?php echo \"welkom $username\"; ?>" > $U_HOME/index.php
```

```
chown $username $U_HOME/index.php
```

```
echo "" > $DB_FILE
```

```
echo "CREATE USER \"${username}\"@\"%\" IDENTIFIED BY '$password';" >> $DB_FILE
```

```
echo "ALTER USER \"${username}\"@\"%\" REQUIRE NONE;" >> $DB_FILE
```

```
echo "CREATE DATABASE IF NOT EXISTS \"${username}\";" >> $DB_FILE
```

```
echo "GRANT ALL PRIVILEGES ON \"${username\".* TO \"${username}\"@\"%\";" >> $DB_FILE
```

```
if [ "$2" == "" ]; then
```

```
    echo "Database not created (will only be done automatically if password is provided via  
command line"
```

```
    echo "Create database with mysql < $DB_FILE"
```

```
    echo "for batch creation uncomment line #mysql < $DB_FILE"
```

```
else
```

```
    mysql < $DB_FILE
```

```
    echo "Database $username created"
```

```
fi
```

```
fi
```

```
else
```

```
    echo "Only root may add a user to the system"
```

```
    echo
```

```
    echo "Usage:"
```

```
    echo "sudo $0 u12345 password"
```

```
    exit 2
```

```
fi
```

```
}
```

```
# if parameter is file then read input from file <user> <password>
```

```
# else read two command line params
```

```
if test -f "$1"; then
```

```
    while IFS= read -r line
```

```
do
    addThisUser $line
done < $1
else
    addThisUser $1 $2
fi
```

Beschrijving `sudo adduser.sh gebruiker password`

In regel 11 en 16 wordt er gecontroleerd of de gebruikersnaam en wachtwoord als parameters zijn meegegeven. Als dat niet het geval dan vraagt het script om een gebruikersnaam en/of wachtwoord.

Regel 22 controleert of de opgegeven gebruiker al bestaat.

Op regel 33 wordt de gebruiker aan het Linux systeem toegevoegd, de waarden die daar boven staan worden gebruikt bij het aanmaken van een gebruiker.

Op regel 37 wordt een standaard index.php voor de gebruiker aangemaakt.

Op regel 40 en verder wordt een file gemaakt waarin SQL commando's worden geplaatst. De inhoud van de ze file moet na dit script nog zelf worden uitgevoerd.

Op regel 47 wordt gekeken of er een password is ingegeven als parameter als dat het geval is dan wordt de db user ook gemaakt.

Hiervoor moet mysql zo zijn geconfigureerd dat er een file vanuit de command line kan worden uitgevoerd.

Op regel 67 begint het script eigenlijk (alles daarvoor is een grote functie). Als parameter 1 een file is dan wordt er een file met gebruikersnamen en wachtwoorden verwerkt en anders wordt de functie 'addThisUser' gewoon aangeroepen met dezelfde parameters als dat dit script is aangeroepen.

--

Revision #6

Created 17 December 2021 18:11:01 by Max

Updated 17 December 2021 18:35:00 by Max