

# KT3 Upload website

Maak lijst met kandidaten in Excel, met alle studenten nummers en verzin/generate wachtwoorden.

Maak een text input file met userid's en passwords (gescheiden door spatie).

Scripts ran on Ubuntu Server VPS

## SFTP restriction

in /etc/ssh/sshd\_config (aan eind toevoegen).

```
Match Group sftp
ForceCommand internal-sftp
ChrootDirectory %h
AllowTCPForwarding no
PermitTunnel no
X11Forwarding no
```

Werkt niet goed - uiteindelijk opgelost met standaard rechten:

```
drwxr-xr-x 14 root root    4096 Jul 26 2020 var/

drwxrwxr-x  9 root  www-data 4096 Nov 30 10:28 www/

drwxr-x--x 78 max   root    4096 Feb  4 21:12 softwaredeveloper/

drwxr-x--x  4 u515  sftp    4096 Feb  5 16:15 u515/
(group root werkt ook)

-rw-rw-r-- 1 u515 sftp      38 Feb  3 19:42 index.php
(group root werkt ook)

Note group sftp only needed if ssh restriction are applied
```

## Readme

input file consists out of line with  
<userid> <password>

```
add users
run addUser.sh <file>
run creatDBUsers.sql file; sudo mysql < createDB Users.sql

remove
run removeUser.sf <file>
go to php myadmin and delete dabases and users (no script but easy with gui)
```

## Script add users

```
#!/bin/bash
# Script to add a user to Linux system
# Params <user> <passw> or <file>

DB_FILE=createDBUsers.sql

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
```

```
fi
```

```
else
```

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

```
exit 2
```

```
fi
```

```
}
```

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

```
# 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
```

## script del users

```
#!/bin/bash
```

```
# remove user, params <user> or <file>
```

```

removeThisUser() {
    if [ $(id -u) -ne 0 ]; then
        echo "Only root may add a user to the system"
        exit 2
    fi

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

    if [ $? -eq 0 ]; then
        userdel -r $username
        getent group $username || groupadd $username

        U_HOME=/var/www/softwaredeveloper/$username

        echo "Remove $U_HOME"

    fi
}

# is parameter is file then read input from file <user>
# else read command line params
if test -f "$1"; then
    while IFS= read -r line
    do
        removeThisUser $line
    done < $1
else
    removeThisUser $1 $2
fi

```