User administration script

GNU/Linux is a multiuser operating system allowing many users to log in and perform several activities at the same time. There are several administration tasks that are handled with user management, which include setting the default shell for the user, disabling a user account, disabling a shell account, adding new users, removing users, setting a password, setting an expiry date for a user account, and so on. This recipe aims at writing a user management tool that can handle all of these tasks.

How to do it...

Let's go through the user administration script:

```
#!/bin/bash
#Filename: user_adm.sh
#Description: A user administration tool

function usage()
{
  echo Usage:
  echo Add a new user
  echo $0 -adduser username password
  echo
  echo Remove an existing user
  echo $0 -deluser username
  echo
  echo Set the default shell for the user
  echo $0 -shell username SHELL_PATH
  echo
  echo Suspend a user account
  echo $0 -disable username
  echo
  echo Enable a suspended user account
  echo $0 -enable username
  echo
  echo Set expiry date for user account
  echo $0 -expiry DATE
  echo
  echo Change password for user account
  echo $0 -passwd username
  echo
  echo Create a new user group
  echo $0 -newgroup groupname
  echo
  echo Remove an existing user group
  echo $0 -delgroup groupname
  echo
  echo Add a user to a group
  echo $0 -addgroup username groupname
  echo
  echo Show details about a user
  echo $0 -details username
  echo
  echo Show usage
  echo $0 -usage
  echo

  exit
}

if [ $UID -ne 0 ];
then
  echo Run $0 as root.
  exit 2
fi

case $1 in

  -adduser) [ $# -ne 3 ] && usage ; useradd $2 -p $3 -m ;;
  -deluser) [ $# -ne 2 ] && usage ; deluser $2 --remove-all-files;;
  -shell)    [ $# -ne 3 ] && usage ; chsh $2 -s $3 ;;
  -disable) [ $# -ne 2 ] && usage ; usermod -L $2 ;;
  -enable) [ $# -ne 2 ] && usage ; usermod -U $2  ;;
  -expiry) [ $# -ne 3 ] && usage ; chage $2 -E $3 ;;
  -passwd) [ $# -ne 2 ] && usage ; passwd $2 ;;
  -newgroup) [ $# -ne 2 ] && usage ; addgroup $2 ;;
  -delgroup) [ $# -ne 2 ] && usage ; delgroup $2 ;;
  -addgroup) [ $# -ne 3 ] && usage ; addgroup $2 $3 ;;
  -details) [ $# -ne 2 ] && usage ; finger $2 ; chage -l $2 ;;
  -usage) usage ;;
  *) usage ;;
esac
```