

```
1 #!/bin/bash
2 # shellcheck source=kzcommon.sh
3 #-----#
4 # Netwerkschijf aankoppelen. #
5 # #
6 # Geschreven door Karel Zimmer <info@karelzimmer.nl>. #
7 # #
8 # Auteursrecht (c) 2014-2020 Karel Zimmer. #
9 # GNU Algemene Publieke Licentie <https://www.gnu.org/licenses/gpl.html>. #
10 # #
11 # RelNum=12.02.07 #
12 # RelDat=2020-11-16 #
13 #-----#
14
15 #-----#
16 # Global constants #
17 #-----#
18 source "${dirname "$0"}/kzcommon.sh
19 readonly NAS_LOGIN=/root/.${PROGRAMNAME}login
20 readonly NAS_MOUNT_POINT=/mnt/${PROGRAMNAME}
21 readonly NAS_FIXED_IP_ADDRESS="Zorg dat de NAS een vast IP-adres heeft.
22 Hiervoor is nodig:
23
24 - het MAC-adres van de NAS (zoals $(
25   ip -oneline link | awk '/ UP /{print $17}'
26 ) )
27 - toegang tot de NAS via een webbrowser
28 - toegang tot de router via een webbrowser
29
30 1. Kies in de router met het MAC-adres van de NAS een
31   IP-adres buiten het bereik van DHCP die automatisch
32   de IP-adressen uitdeelt (vaak 192.168.0.200).
33 2. Kies in de NAS het vaste IP-adres en zet DHCP uit."
34 readonly RUN_AS_SUPERUSER=true
35 readonly SYSTEMD_MOUNT_DIR=/etc/systemd/system
36 readonly SYSTEMD_MOUNT_FILE=mnt-nas.mount
37 readonly SYSTEMD_MOUNT=${SYSTEMD_MOUNT_DIR}/${SYSTEMD_MOUNT_FILE}
38
39 # Bij aanpassingen ook .completion aanpassen!
40 readonly OPTIONS_SHORT=${OPTIONS_SHORT_COMMON}
41 readonly OPTIONS_LONG=${OPTIONS_LONG_COMMON}
42 readonly USAGE="Gebruik: $PROGRAMNAME $OPTIONS_USAGE_COMMON
43
44 $OPTIONS_LONG_SHORT"
45 readonly HELP="Gebruik: $PROGRAMNAME [OPTIE...]"
46
47 Netwerkschijf aankoppelen.
48
49 Opties:
50   $OPTIONS_LONG_SHORT
51
52 $OPTIONS_HELP_COMMON"
53
54 #-----#
55 # Global variables #
56 #-----#
57 declare NAS_GEBRUIKERSNAAM=${SUDO_USER:-$USER}
58 declare NAS_IP_ADRES=192.168.0.200
59 declare NAS_SHARENAAM=${SUDO_USER:-$USER}
60 declare NAS_WACHTWOORD='geheim'
61
62 #-----#
63 # Functions #
64 #-----#
65 check_input() {
66     local -i getopt_rc=0
```

```

67     local parsed=''
68
69     parsed=$(
70         getopt --alternative          \
71                --options             "$OPTIONS_SHORT" \
72                --longoptions         "$OPTIONS_LONG"  \
73                --name                 "$PROGNAME"     \
74                -- "$@"
75     ) || getopt_rc=$?
76     if [[ $getopt_rc -ne 0 ]]; then
77         printf '%s\n' "$USAGELINE" >&2
78         QUIET=true exit $ERROR
79     fi
80     eval set -- "$parsed"
81     process_general_options "$@"
82
83     while true; do
84         case $1 in
85             --)
86                 shift
87                 break
88                 ;;
89             *)
90                 shift
91                 ;;
92         esac
93     done
94
95     if [[ "$*" ]]; then
96         printf "$PROGNAME: %s\n%s\n" 'geen argumenten opgeven' "$USAGELINE" >&2
97         QUIET=true exit $ERROR
98     fi
99
100    check_user
101    # Na check_user voor toegang tot media.
102    remove_old_configuration
103    request_input
104 }
105
106 remove_old_configuration() {
107     rm --force /root/.naslogin
108 }
109
110 request_input() {
111     local fstab=''
112     local quest1='Deze regels verwijderen?'
113     local quest2="Heeft de NAS een vast IP-adres zoals $NAS_IP_ADRES?"
114     local quest3='IP-adres van de NAS'
115     local quest4='Gebruikersnaam op de NAS'
116     local quest5='Wachtwoord van deze gebruiker op de NAS'
117     local quest6='Share-naam van deze gebruiker op de NAS'
118     local -i zenity_rc=0
119     local text='Beantwoord alstublieft de volgende vragen'
120     local warning=''
121
122     fstab=$(
123         if ! grep --regexp="$PROGNAME" /etc/fstab; then
124             printf '%s' ''
125         fi
126     )
127     if [[ $fstab ]]; then
128         warning="Onderstaande regels zijn al aanwezig in de koppeltabel:
129 $fstab"
130         if $OPTION_GUI; then
131             # Constructie '2> >($LOGCMD)' om stderr naar de log te krijgen.
132             # Voorbeeld: Unable to init server: Kon niet verbinden:

```

```
133         #           Verbinding is geweigerd
134         #           en: (zenity:47712): Gtk-WARNING **: 10:35:49.339:
135         #           cannot open display:
136         zenity --question                \
137             --no-markup                  \
138             --width 400                   \
139             --height 100                  \
140             --title 'Waarschuwing'       \
141             --text "$warning\n\n$quest1" \
142             2> >($LOGCMD) || zenity_rc=?
143         if [[ $zenity_rc -eq 0 ]]; then
144             verwijder_regels
145         fi
146     else
147         printf '%s\n' "$warning"
148         read -rp "$quest1 [j/N]: " < /dev/tty
149         while true; do
150             case $REPLY in
151                 j*|J*)
152                     verwijder_regels
153                     break
154                     ;;
155                 n*|N*|'')
156                     info 'Ok, we gaan verder.'
157                     break
158                     ;;
159                 *)
160                     printf '%b' "${UP_ONE_LINE}${ERASE_LINE}"
161                     continue
162                     ;;
163             esac
164         done
165     fi
166 fi
167
168 if $OPTION_GUI; then
169     if ! zenity --question                \
170             --no-markup                  \
171             --width 400                   \
172             --height 100                  \
173             --title 'Vraag'              \
174             --text "$quest2"             \
175             2> >($LOGCMD); then
176         zenity --info                      \
177             --no-markup                  \
178             --width 500                   \
179             --height 100                  \
180             --title 'Melding'            \
181             --text "$NAS_FIXED_IP_ADDRESS" \
182             2> >($LOGCMD)                || true
183         QUIET=true exit $SUCCESS
184     fi
185 else
186     while true; do
187         read -rp "$quest2 [j/N]: " < /dev/tty
188         case $REPLY in
189             j*|J*)
190                 info 'Ok, we gaan verder.'
191                 break
192                 ;;
193             n*|N*|'')
194                 printf '%s\n' "$NAS_FIXED_IP_ADDRESS"
195                 info 'Gestopt.'
196                 QUIET=true exit $SUCCESS
197                 ;;
198             *)
```

```

199             printf '%b' "${UP_ONE_LINE}${ERASE_LINE}"
200             continue
201             ;;
202         esac
203     done
204 fi
205
206 if $OPTION_GUI; then
207     REPLY=$(
208         zenity --forms                \
209               --title                 'Vragen' \
210               --text                  "$text" \
211               --separator             ", " \
212               --add-entry             "$quest3" \
213               --add-entry             "$quest4" \
214               --add-password          "$quest5" \
215               --add-entry             "$quest6" \
216               2> >($LOGCMD)
217     ) || zenity_rc=$?
218     if [[ $zenity_rc -ne 0 ]]; then
219         QUIET=true exit $SUCCESS
220     fi
221     NAS_IP_ADRES=$(printf '%s' "$REPLY" | awk -F, '{print $1}')
222     NAS_GEBRUIKERSNAAM=$(printf '%s' "$REPLY" | awk -F, '{print $2}')
223     NAS_WACHTWOORD=$(printf '%s' "$REPLY" | awk -F, '{print $3}')
224     NAS_SHARENAAM=$(printf '%s' "$REPLY" | awk -F, '{print $4}')
225 else
226     printf '%s\n%s\n' "$text." \
227 'Als u niets opgeeft, wordt de waarde tussen [] gebruikt als antwoord.'
228     read -rp "
229 $quest3? [$NAS_IP_ADRES]: " < /dev/tty
230     if [[ $REPLY ]]; then
231         NAS_IP_ADRES=$REPLY
232     fi
233     read -rp "
234 $quest4? [$NAS_GEBRUIKERSNAAM]: " < /dev/tty
235     if [[ $REPLY ]]; then
236         NAS_GEBRUIKERSNAAM=$REPLY
237     fi
238     read -rsp "
239 $quest5? [$NAS_WACHTWOORD]: " < /dev/tty
240     if [[ $REPLY ]]; then
241         NAS_WACHTWOORD=$REPLY
242     fi
243     printf '\n'
244     read -rp "
245 $quest6? [$NAS_SHARENAAM]: " < /dev/tty
246     if [[ $REPLY ]]; then
247         NAS_SHARENAAM=$REPLY
248     fi
249 fi
250 }
251
252 process_input() {
253     local text0='Opdrachten worden verwerkt'
254     local text1='Maak koppelpunt'
255     local text2='Maak NAS login'
256     local text3='Maak systemd'
257     local text4='Koppel NAS aan'
258     local title='Netwerkschijf aankoppelen'
259
260     if $OPTION_GUI; then
261         (
262             printf '%s\n' "#$text1"
263             create_mountpoint
264             printf '%s\n' "#$text2"

```

```

265     create_nas_login
266     printf '%s\n' "#$text3"
267     create_systemd_unit_file
268     printf '%s\n' "#$text4"
269     mount_nas
270     ) |&
271     zenity --progress          \
272           --pulsate           \
273           --auto-close       \
274           --no-cancel        \
275           --width             600 \
276           --height           50  \
277           --title             "$title" \
278           --text              "$text0" \
279           2> >($LOGCMD)
280     else
281         create_mountpoint
282         create_nas_login
283         create_systemd_unit_file
284         mount_nas
285     fi
286 }
287
288 create_mountpoint() {
289     if [[ -d $NAS_MOUNT_POINT ]]; then
290         return 0
291     fi
292
293     printf '%s\n' 'Maak NAS koppelpunt ...'
294
295     mkdir "$NAS_MOUNT_POINT" |& $LOGCMD
296     chmod '777' "$NAS_MOUNT_POINT" |& $LOGCMD
297 }
298
299 create_nas_login() {
300     printf '%s\n' 'Maak NAS login ...'
301
302     printf '%b\n' "username=$NAS_GEBRUIKERSNAAM\npassword=$NAS_WACHTWOORD" \
303             > "$NAS_LOGIN"
304     chmod '600' "$NAS_LOGIN" |& $LOGCMD
305 }
306
307 create_systemd_unit_file() {
308     printf '%s\n' 'Maak systemd mount-bestand ...'
309
310     printf '%s\n' "[Unit]
311 Description=Mount NAS Directory
312
313 [Mount]
314 What=//$NAS_IP_ADRES/$NAS_SHARENAAM
315 Where=/mnt/nas
316 Type=cifs
317 Options=credentials=$NAS_LOGIN,sec=ntlm,vers=1.0,rw,noperm,iocharset=utf8,\
318 dir_mode=0777,file_mode=0777
319
320 [Install]
321 WantedBy=multi-user.target" > "$SYSTEMD_MOUNT"
322 }
323
324 mount_nas() {
325     printf '%s\n' 'Koppel NAS aan ...'
326
327     systemctl enable "$SYSTEMD_MOUNT_FILE" |& $LOGCMD
328     if ! systemctl start "$SYSTEMD_MOUNT_FILE" |& $LOGCMD; then
329         true
330     fi

```

```
331     systemctl daemon-reload |& $LOGCMD
332 }
333
334 term_script() {
335     if ! systemctl status "$SYSTEMD_MOUNT_FILE"; then
336         true
337     fi
338     info "Het systemd mount-bestand is geactiveerd.
339 De NAS zou beschikbaar moeten zijn op '$NAS_MOUNT_POINT'.
340
341 De-actieveer het systemd mount-besta2nd met de opdracht:
342     ${BLUE}sudo systemctl disable mnt-nas.mount${NORMAL}"
343     QUIET=true exit $SUCCESS
344 }
345
346 #-----#
347 # Main line                                     #
348 #-----#
349 main() {
350     init_script
351     check_input "$@"
352     process_input
353     term_script
354 }
355
356
357 main "$@"
358
359 # EOF
```