

```
1 #!/bin/bash
2 #####
3 # Bestand: mkusb #
4 # Doel: Maak opstartbare USB-sticks. #
5 # Gebruik: In het terminalvenster: #
6 # mkusb [OPTIES] [ARGUMENTEN] #
7 # Auteur: Karel Zimmer (http://karelzimmer.nl, info@karelzimmer.nl) #
8 # ----- #
9 # Auteursrecht © 2014-2017 Karel Zimmer. #
10 # #
11 # Dit programma is vrije software: u mag het herdistribueren en/of wijzigen #
12 # onder de voorwaarden van de GNU Algemene Publieke Licentie zoals #
13 # gepubliceerd door de Free Software Foundation, onder versie 3 van de #
14 # Licentie of (naar uw keuze) elke latere versie. #
15 # #
16 # Dit programma is gedistribueerd in de hoop dat het nuttig zal zijn maar #
17 # ZONDER ENIGE GARANTIE; zelfs zonder de impliciete garanties die #
18 # GEBRUIKELIJK ZIJN IN DE HANDEL of voor BRUIKBAARHEID VOOR EEN SPECIFIEK #
19 # DOEL. Zie de GNU Algemene Publieke Licentie voor meer details. #
20 # #
21 # U hoort een kopie van de GNU Algemene Publieke Licentie te hebben #
22 # ontvangen samen met dit programma. Als dat niet het geval is, zie #
23 # http://www.gnu.org/licenses/. #
24 # ----- #
25 # Versies: 1.0.0 2014-06-01 Eerste versie. #
26 # 2.0.0 2014-09-17 Optie dev toegevoegd. #
27 # 3.0.0 2014-12-03 USB-stick schonen met sgdisk en argument #
28 # DEVICE toegevoegd. #
29 # 4.0.0 2015-06-05 Optie bronmap toegevoegd. #
30 # 5.0.0 2015-06-05 Voortaan uitvoeren als beheerder. #
31 # 6.0.0 2015-07-08 Hernoemd (-/- .sh). #
32 # 7.0.0 2015-08-08 Optie bronmap en argument device aangepast. #
33 # 8.0.0 2016-05-20 Van eigen log naar system log en systemd #
34 # journal. #
35 # 9.0.0 2016-07-23 Controle gemaakte USB-stick toegevoegd. #
36 #####
37 readonly VERSION_NUMBER=9.10.6
38 readonly RELEASE_DATE=2017-08-15
39
40 #####
41 # Instellingen #
42 #####
43
44 #-----#
45 # Algemene instellingen #
46 # -----#
47 # Lees de algemene variabelen en functies in. #
48 #-----#
49 if ! source "$(dirname "$0")/script-common.sh &> /dev/null; then
50 echo "Het algemeen scriptbestand (script-common.sh) is niet gevonden of
51 bevat fouten. Is 'wget karelzimmer.nl/s;. s' wel uitgevoerd?
52 Zie voor informatie http://karelzimmer.nl en klik op LEESMIJ."
53 exit 1
54 fi
55
56 #-----#
57 # Globale constanten #
58 #-----#
59
60 # Algemeen =====
61 readonly SCRIPT_NEEDS_SUDO=true # Beheerdersrechten nodig
62 readonly FIRST_COPYRIGHTYEAR=2014 # Eerste auteursrechtjaar
63
64 # Specifiek =====
65 readonly SOURCE_DFLT=$XDG_DOWNLOAD_DIR # Standaard bronmap m/beelddbestanden
66 readonly DEVICE_DFLT=/dev/sdb # standaard device-naam
```



```
133 #-Functie-----#
134 # Naam: controleer_invoer #
135 # Doel: Controleer en verwerk de invoer van het script. #
136 # Arg.: Script invoer. #
137 #-----#
138 controleer_invoer() {
139
140     local dir=''
141     local one_opt_source=true
142     local parm_error=false
143
144     # Test op aanwezigheid nieuwe versie getopt.
145     set +o errexit
146     getopt --test > /dev/null
147     getopt_rc=$?
148     set -o errexit
149     if [[ $getopt_rc -eq 4 ]]; then
150
151         # Ontleed (parse) opties en argumenten met getopt.
152         PARSED=$(getopt --alternative \
153             --options $OPTS_SHORT \
154             --longoptions $OPTS_LONG \
155             --name "$PROGNAME" \
156             -- "$@")
157
158         # Controleer op fouten.
159         if [[ $? -gt 0 ]]; then
160             info $HELPLINE
161             exit 2
162         fi
163
164         # Gebruik eval om quoting correct te verwerken.
165         eval set -- "$PARSED"
166     else
167         # Voeg '--' toe aan de invoer i.v.m. eindeloze loop (break op '--').
168         eval set -- "$@ --"
169         log "Oude versie van getopt in gebruik; opties exact opgeven \
170 volgens optie usage (-u)"
171     fi
172
173     # Verwerk de opties en optie-argumenten.
174     while true; do
175         case $1 in
176             -s|--source)
177                 if $OPT_SOURCE && $one_opt_source; then
178                     info "$PROGNAME: maximaal één optie source (-s) \
179 toegestaan"
180                     parm_error=true
181                     one_opt_source=false
182                 else
183                     OPT_SOURCE=true
184                     SOURCE_ARG=$2
185                 fi
186                 shift 2
187                 ;;
188             -d|--debug)
189                 OPT_DEBUG=true
190                 VERBOSE='--verbose'
191                 shift
192                 ;;
193             -h|--help)
194                 OPT_HELP=true
195                 toon_gebruik $OPT_USAGE
196                 exit 0
197                 ;;
198             -u|--usage)
```

```
199         OPT_USAGE=true
200         toon_gebruik $OPT_USAGE
201         exit 0
202         ;;
203     -v|--version)
204         OPT_VERSION=true
205         toon_versie
206         exit 0
207         ;;
208     --)
209         shift
210         break
211         ;;
212     *)
213         error "$PROGNAME: interne fout bij afhandelen optie '$1'"
214         parm_error=true
215         shift
216         ;;
217     esac
218 done
219
220 # Verwerk de non-optie argumenten.
221 while [[ @$@ ]]; do
222     # Argument USB-stick-device.
223     if $ARG_DEVICE; then
224         info "$PROGNAME: maximaal één argument toegestaan"
225         parm_error=true
226         break
227     else
228         ARG_DEVICE=true
229         DEVICE_ARG=$1
230     fi
231     shift
232 done
233
234 # Geef optie help en gebruik aan als er fouten zijn geconstateerd.
235 if $parm_error; then
236     info $HELPLINE
237     exit 3
238 fi
239
240 # Verwerk en controleer optie en argument source.
241 if $OPT_SOURCE; then
242     SOURCEDIR=$SOURCE_ARG
243 else
244     SOURCEDIR=$SOURCE_DFLT
245 fi
246 if [[ ! -d $SOURCEDIR ]]; then
247     info "$PROGNAME: map '$SOURCEDIR' bestaat niet"
248     exit $E_DIR_NOT_FOUND
249 fi
250
251 # Verwerk argument device.
252 if $ARG_DEVICE; then
253     DEVICE=$DEVICE_ARG
254 else
255     DEVICE=$DEVICE_DFLT
256 fi
257
258 # Vraag voor de zekerheid; opdracht 'dd' is zeer krachtig.
259 warning "
260 De volgende schijven en partities zijn aanwezig:"
261 # findmnt wordt door mount aangeraden, speciaal voor in scripts.
262 findmnt --list \
263         --noheadings \
264         --options=rw \
```

```
265         --output=SOURCE,TARGET
266     warning "
267     ****
268     *** '$DEVICE' wordt gebruikt als doel en OVERSCHREVEN! ***
269     ****
270     read -p 'Is dit OK? (j/N) '
271     case $REPLY in
272     j*|J*)
273         log 'Gebruiker kiest doorgaan met USB-stick maken.'
274         ;;
275     n*|N*|'')
276         info 'Stoppen.'
277         exit 0
278         ;;
279     *)
280         info 'Foutieve keuze - afbreken.'
281         exit 0
282         ;;
283     esac
284
285     return 0
286 }
287
288 #-Functie-----#
289 # Naam: toon_invoer #
290 # Doel: Toon wat het script gaat doen. #
291 # Arg.: Geen argumenten. #
292 #-----#
293 toon_invoer() {
294
295     info " Script: $PROGNAME
296 ISO-map: $SOURCEDIR
297 USB-map: $DEVICE
298 "
299
300     return 0
301 }
302
303 #-Functie-----#
304 # Naam: verwerk_invoer #
305 # Doel: Lees de beeldbestanden en kopieer deze met opdracht 'dd' naar de #
306 # USB-stick. #
307 # Arg.: Geen argumenten. #
308 #-----#
309 verwerk_invoer() {
310
311     local usbname=''
312     local file=''
313
314     info "
315 Lees beeldbestanden (.iso)..."
316     NO_FILES_FOUND=true
317     for file in "$SOURCEDIR"/*.iso; do
318         [[ -f "$file" ]] || continue
319         NO_FILES_FOUND=false
320         usbname=$(basename "$file" .iso | tr '-' ' ')
321
322         info --bold "$usbname"
323
324         read -p 'Maken/Overslaan/Stoppen? (m/0/s) ' </dev/tty
325         case $REPLY in
326         m*|M*)
327             maak_usb_stick "$file"
328             ;;
329         o*|0*|'')
330             continue
```

```

331         ;;
332         s*|S*)
333             info 'Stoppen.'
334             exit 0
335         ;;
336     *)
337         info 'Foutieve keuze - afbreken.'
338         exit 0
339     ;;
340 done
341 done
342     $NO_FILES_FOUND && warning "Geen beeldbestanden gevonden in map \
343 '$SOURCE_DIR'."
344
345     return 0
346 }
347
348 #-Functie-----#
349 # Naam: maak_usb_stick #
350 # Doel: Kopieer beeldbestand naar USB-stick met opdracht 'dd'. #
351 # Arg.: Eén verplicht argument: #
352 #     1. beeldbestand     string, invoer #
353 #-----#
354 maak_usb_stick() {
355
356     local dd_from=${1:-/tmp}
357     local -i unmount_rc=0
358     # local usbname is gedefinieerd in aanroepende functie
359
360     read -p "Plaats USB-stick '$usbname', gevolgd door Enter "
361     if [[ -e $DEVICE ]]; then
362         info "USB-stick '$usbname' wordt gemaakt (dit kan even duren)..."
363
364         # Kopieer het hybride beeldbestand (.iso). De opdracht dd "convert
365         # and copy a file".
366         dd          if=$dd_from \
367                   of=$DEVICE \
368                   bs=4M      |& $LOGCMD
369
370         sync |& $LOGCMD
371
372         # Breng de gebruiker op de hoogte dat de USB-stick klaar is.
373         command_exists canberra-gtk-play &&
374             canberra-gtk-play --id=complete |& $LOGCMD &
375         command_exists notify-send &&
376             notify-send --urgency=normal \
377                 --icon=utilities-terminal \
378                 "USB-stick '$usbname' is gemaakt" \
379                 "Zie terminalvenster voor meer informatie" \
380                 |& $LOGCMD
381         success "USB-stick '$usbname' is gemaakt."
382
383         read -p 'Controleren? (J/n) '
384         case $REPLY in
385             j*|J*|'')
386                 if [[ $dd_from == *amd64* ]]; then
387                     sudo qemu-system-x86_64 \
388                         -enable-kvm \
389                         -m 512 -hda \
390                         $DEVICE |& $LOGCMD &
391                 elif [[ $dd_from == *i386* ]]; then
392                     sudo qemu-system-i386 \
393                         -enable-kvm \
394                         -m 512 -hda \
395                         $DEVICE |& $LOGCMD &
396                 fi

```

```
397         else
398             info "Sorry, alleen voor 32-bit en 64-bit ISO's."
399         fi
400     ;;
401 *)
402     :
403     ;;
404 esac
405
406 log "Koppel af '$DEVICE'..."
407 # umount/unmount via udiskctl (udisks command line tool).
408 udiskctl unmount \
409     --block-device $DEVICE |& $LOGCMD
410 umount_rc=$?
411 udiskctl power-off \
412     --block-device $DEVICE |& $LOGCMD
413 if [[ $umount_rc -eq 0 ]]; then
414     read -p "Verwijder USB-stick '$usbname', gevolgd door Enter "
415 else
416     read -p "Onkoppel en verwijder zelf USB-stick '$usbname', \
417 gevolgd door Enter "
418 fi
419
420 else
421     warning "USB-stick '$usbname' is niet aanwezig op $DEVICE - \
422 overslaan."
423 fi
424
425 return 0
426 }
427
428 #-Functie-----#
429 # Naam: toon_afsluiten #
430 # Doel: Afsluitende meldingen en/of acties. #
431 # Arg.: Geen argumenten. #
432 #-----#
433 toon_afsluiten() {
434     :
435
436     return 0
437 }
438
439 #####
440 # Hoofdlijn #
441 #####
442
443 # init_script
444 {
445     controleer_invoer "$@"
446     controleer_gebruiker $SCRIPT_NEEDS_SUDO "$@"
447     toon_invoer
448 }
449
450 # verwerk
451 {
452     toon_gestart
453     verwerk_invoer
454     toon_gestopt
455 }
456
457 # afsl_script
458 {
459     toon_afsluiten
460     exit 0
461 }
462 }
```

463

464 # Einde script.