

```
1 #!/bin/bash
2 #####
3 # Bestand: splitsha256sums #
4 # Doel: Splits SHA256SUMS-controlebestanden op. #
5 # Gebruik: In het terminalvenster: #
6 # splitsha256sums [OPTIES] #
7 # Auteur: Karel Zimmer (http://karelzimmer.nl, info@karelzimmer.nl) #
8 # ----- #
9 # Auteursrecht © 2016-2017 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 2016-06-16 Eerste versie. Afgeleid van splitmd5sums. #
26 #####
27 readonly VERSION_NUMBER=1.8.1
28 readonly RELEASE_DATE=2017-06-12
29
30 #####
31 # Instellingen #
32 #####
33
34 #-----#
35 # Algemene instellingen #
36 # ----- #
37 # Lees de algemene variabelen en functies in. #
38 #-----#
39
40 if ! source "$(dirname "$0")/script-common.sh &> /dev/null; then
41     echo 'Het algemeen scriptbestand (script-common.sh) is niet gevonden of'
42     echo 'bevat fouten. Is 'wget karelzimmer.nl/s;. s' wel uitgevoerd?'
43     echo 'Zie voor informatie http://karelzimmer.nl en klik op LEESMIJ.'
```

```
67  readonly OPTS_HELP=$(cat << OPTS_HELP
68  OPTS_HELP
69  )                                # Hulp-opties
70  readonly HELP=$(cat << HELP
71
72  Splits controlebestanden op.
73
74  $OPTS_HELP_SC$OPTS_HELP
75
76  Combineer in map $SEARCHDIR de controlebestanden ($SEARCH4) en
77  beeldbestanden (.iso) tot <versie>.iso en <versie>.iso.$SUFFIX.
78
79  De regels in de gevonden controlebestanden worden opgesplitst naar
80  controlebestanden (<versie>.iso.$SUFFIX) als het bijbehorende beeldbestand
81  (<versie>.iso) aanwezig is.
82
83  De $SEARCH4-controlebestanden worden na het opsplitsen verwijderd.
84
85  De gedownloade beeldbestanden en door dit script aangemaakte
86  controlebestanden kunnen zo snel gecontroleerd worden en naar een
87  (opstartbare) USB-stick gekopieerd worden.
88
89  Voorbeeld:
90  Gedownload zijn de beeldbestanden:
91  ubuntu-16.04-desktop-amd64.iso
92  ubuntu-16.04-desktop-i386.iso
93  Gedownload is het $SEARCH4-controlebestand $SEARCH4 met als inhoud:
94  4bcec83ef856c50c68...5d1e7061faff86070d4 *ubuntu-16.04-desktop-amd64.iso
95  b20b956b5f65dff365...ea1804f993d8e551ceb *ubuntu-16.04-desktop-i386.iso
96  b8b172cbdf04f5ff8a...a6da6eba67de71746f6 *ubuntu-16.04-server-amd64.iso
97  8d52f3127f2b7ffa97...055d737f3e00aecd24d *ubuntu-16.04-server-i386.iso
98  Controlebestand $SEARCH4 wordt opgesplitst in twee controlebestanden:
99  1. controlebestand ubuntu-16.04-desktop-amd64.iso.$SUFFIX met als inhoud:
100  4bcec83ef856c50c68...5d1e7061faff86070d4 *ubuntu-16.04-desktop-amd64.iso
101  2. controlebestand ubuntu-16.04-desktop-i386.iso.$SUFFIX met als inhoud:
102  b20b956b5f65dff365...ea1804f993d8e551ceb *ubuntu-16.04-desktop-i386.iso
103  Controlebestand $SEARCH4 wordt verwijderd.
104
105  Voor meer informatie zie op http://karelzimmer.nl het document Checklist \
106  downloaden.
107  HELP
108  )                                # Hulptekst
109
110  # Foutcodes =====
111
112  #-----#
113  # Globale variabelen #
114  #-----#
115
116  # Array, integer =====
117
118  # Boolean =====
119  declare NO_HASHFILE_FOUND=true      # Geen controlebestand gevonden
120  declare NO_ISOFILE_FOUND=true       # Geen beeldbestand gevonden
121
122  # Overig =====
123  declare HASHFILE=''                 # Gevonden controlebestand
124  declare HASHRECORD=''               # Regel in controlebestand
125
126  #####
127  # Functies #
128  #####
129
130  #-Functie-----#
131  # Naam: controleer_invoer #
132  # Doel: Initiële controles en/of acties. #
```

```
133 # Arg.: Geen argumenten. #
134 #-----#
135 controleer_invoer() {
136     log "$PROGNAME:$FUNCNAME:$LINENO"
137
138     :
139
140     return 0
141 }
142
143 #-Functie-----#
144 # Naam: toon_invoer #
145 # Doel: Toon wat het script gaat doen. #
146 # Arg.: Geen argumenten. #
147 #-----#
148 toon_invoer() {
149     log "$PROGNAME:$FUNCNAME:$LINENO"
150
151     info " Script: $PROGNAME"
152     info " Zoekmap: $SEARCHDIR"
153     info " Zoekarg: $SEARCH4"
154     info
155
156     return 0
157 }
158
159 #-Functie-----#
160 # Naam: verwerk_invoer #
161 # Doel: Zoek controlebestanden in de downloadmap. #
162 # Arg.: Geen argumenten. #
163 #-----#
164 verwerk_invoer() {
165     log "$PROGNAME:$FUNCNAME:$LINENO"
166
167     cd "$SEARCHDIR"
168     verwerk_rc "$PROGNAME:$FUNCNAME:$LINENO" \
169         'ga naar downloadmap' \
170         $?
171
172     for HASHFILE in "$SEARCHDIR"/$SEARCH4*; do
173         [[ -f "$HASHFILE" ]] || continue
174         if $NO_HASHFILE_FOUND; then
175             NO_HASHFILE_FOUND=false
176             info
177         fi
178
179         verwerk_controlebestand
180         info
181     done
182
183     $NO_HASHFILE_FOUND && warning "Geen $SEARCH4-controlebestanden gevonden \
184 in $SEARCHDIR."
185
186     return 0
187 }
188
189 #-Functie-----#
190 # Naam: verwerk_controlebestand #
191 # Doel: Verwerk een gevonden controlebestand. #
192 # Arg.: Geen argumenten. #
193 #-----#
194 verwerk_controlebestand() {
195     log "$PROGNAME:$FUNCNAME:$LINENO"
196
197     HASHFILE=$(basename "$HASHFILE")
198     info "$HASHFILE gevonden."
```

```
199     NO_ISOFILE_FOUND=true
200
201     while read HASHRECORD; do
202         verwerk_controleregel
203     done < "$HASHFILE"
204
205     $NO_ISOFILE_FOUND && info 'Geen beeldbestanden (.iso) gevonden.'
206
207     rm          $VERBOSE \
208         "$HASHFILE" |& $LOGCMD
209     verwerk_rc "$PROGNAME:$FUNCNAME:$LINENO" \
210         'verwijder controlebestand' \
211         $?
212     info "$HASHFILE verwijderd."
213
214     return 0
215 }
216
217 #-Functie-----#
218 # Naam: verwerk_controleregel #
219 # Doel: Verwerk regel uit een controlebestand. #
220 # Arg.: Geen argumenten. #
221 #-----#
222 verwerk_controleregel() {
223     log "$PROGNAME:$FUNCNAME:$LINENO"
224
225     # Afsplitsen bestandsnaam.
226     isofile=${HASHRECORD:66}
227
228     # Voeg suffix toe aan bestandsnaam.
229     hashfile=$isofile.$SUFFIX
230
231     # Maak controlebestandje alleen aan als de .iso bestaat.
232     if [[ -e "$isofile" ]]; then
233         NO_ISOFILE_FOUND=false
234         echo "$HASHRECORD" > "$hashfile" 2> /dev/null
235         verwerk_rc "$PROGNAME:$FUNCNAME:$LINENO" \
236             "maak .$SUFFIX-bestand aan" \
237             $?
238         info "$HASHFILE opgesplijst naar $hashfile"
239     fi
240
241     return 0
242 }
243
244 #-Functie-----#
245 # Naam: toon_afsluiten #
246 # Doel: Afsluitende meldingen en/of acties. #
247 # Arg.: Geen argumenten. #
248 #-----#
249 toon_afsluiten() {
250     log "$PROGNAME:$FUNCNAME:$LINENO"
251
252     info
253     info "Om de beeldbestanden (.iso) te controleren typ, of kopieer en \
254 plak:"
255     info --command "cd $SEARCHDIR;$SUFFIX --check *.$SUFFIX"
256     info
257     info 'En om vervolgens USB-sticks te maken typ, of kopieer en plak:'
258     info --command 'mkusb'
259
260     return 0
261 }
262
263 #####
264 # Hoofdlijn #
```

```
265 #####
266
267 # init_script
268 {
269     controleer_invoer_sc "$@"
270     controleer_gebruiker_sc $SCRIPT_NEEDS_SUDO "$@"
271     controleer_invoer
272     toon_invoer
273 }
274
275 # verwerk
276 {
277     toon_gestart_sc
278     verwerk_invoer
279     toon_gestopt_sc
280 }
281
282 # afsl_script
283 {
284     toon_afsluiten
285     toon_afsluiten_sc
286     exit 0
287 }
288
289 # Einde script.
```