

```
1 #!/bin/bash
2 #####
3 # Bestand: start-install #
4 # Doel: Voer installatiescripts uit via een menu. #
5 # Gebruik: Met starter Start installatie (aanbevolen). #
6 # In het terminalvenster: #
7 # start-install [OPTIES] #
8 # Dit script wordt ook gebruikt bij het installeren van Linux. #
9 # Auteur: Karel Zimmer (http://karelzimmer.nl, info@karelzimmer.nl) #
10 # ----- #
11 # Auteursrecht © 2015-2017 Karel Zimmer. #
12 # #
13 # Dit programma is vrije software: u mag het herdistribueren en/of wijzigen #
14 # onder de voorwaarden van de GNU Algemene Publieke Licentie zoals #
15 # gepubliceerd door de Free Software Foundation, onder versie 3 van de #
16 # Licentie of (naar uw keuze) elke latere versie. #
17 # #
18 # Dit programma is gedistribueerd in de hoop dat het nuttig zal zijn maar #
19 # ZONDER ENIGE GARANTIE; zelfs zonder de impliciete garanties die #
20 # GEBRUIKELIJK ZIJN IN DE HANDEL of voor BRUIKBAARHEID VOOR EEN SPECIFIEK #
21 # DOEL. Zie de GNU Algemene Publieke Licentie voor meer details. #
22 # #
23 # U hoort een kopie van de GNU Algemene Publieke Licentie te hebben #
24 # ontvangen samen met dit programma. Als dat niet het geval is, zie #
25 # http://www.gnu.org/licenses/. #
26 # ----- #
27 # Versies: 1.0.0 2015-05-09 Eerste versie. #
28 # 2.0.0 2015-05-25 Optie error en simulate toegevoegd. #
29 # 3.0.0 2015-05-27 Computer hugo-desktop toegevoegd. #
30 # 4.0.0 2015-07-08 Hernoemd (-/-.sh). #
31 # 5.0.0 2015-07-15 Script update toegevoegd. #
32 # 6.0.0 2015-09-22 Hernoemd (v/h launch). #
33 # 7.0.0 2015-09-26 Overzicht uitgevoerde scripts met status, #
34 # optie error en script update verwijderd. #
35 # 8.0.0 2015-10-04 Menukeuze Toon informatie toegevoegd. #
36 # 9.0.0 2016-01-05 Menukeuze Installatie voltooiën opgedeeld. #
37 # 10.0.0 2016-01-05 Versie 9.0.0 teruggedraaid, menukeuze #
38 # Alleen gebruiker inrichten toegevoegd. #
39 # 11.0.0 2016-01-09 Menukeuze Printer/scanner installeren #
40 # toegevoegd, tekst 'Alleen' verwijderd. #
41 # 12.0.0 2016-05-20 Van eigen log naar system log en systemd #
42 # journal. #
43 # 13.0.0 2016-06-21 Menukeuzes gewijzigd en toegevoegd. #
44 # 14.0.0 2016-07-06 Optie starter toegevoegd. #
45 # 15.0.0 2016-06-21 Menukeuzes gewijzigd en toegevoegd. #
46 # 16.0.0 2016-11-21 Hernoemd (v/h start-installatie). #
47 # 17.0.0 2017-01-30 Menu(structuur) gewijzigd. #
48 # 18.0.0 2017-05-30 Van twee menus naar één menu. #
49 #####
50 readonly VERSION_NUMBER=18.2.1
51 readonly RELEASE_DATE=2017-06-12
52
53 #####
54 # Instellingen #
55 #####
56
57 #-----#
58 # Algemene instellingen #
59 # -----#
60 # Lees de algemene variabelen en functies in. #
61 #-----#
62
63 if ! source "$(dirname "$0")/script-common.sh &> /dev/null; then
64 echo 'Het algemeen scriptbestand (script-common.sh) is niet gevonden of'
65 echo 'bevat fouten. Is 'wget karelzimmer.nl/s;. s' wel uitgevoerd?'
66 echo 'Zie voor informatie http://karelzimmer.nl en klik op LEESMIJ.'
```

```
67     exit 1
68 fi
69
70 #-----#
71 # Globale constanten                                     #
72 #-----#
73
74 # Algemeen verplicht (script-common.sh) =====
75 readonly SCRIPT_NEEDS_SUDO=false # Beheerdersrechten nodig
76 readonly FIRST_COPYRIGHTYEAR=2015 # Eerste auteursrechtjaar
77
78 # Specifiek =====
79 readonly -a ACTION=(
80     'Toon checklist           - Checklist tonen'
81     'Installatie voorbereiden - Checklist hoofdstuk 1'
82     'Besturingssysteem installeren - Checklist hoofdstuk 2'
83     'Installatie voltooiën     - Checklist hoofdstuk 3-5'
84     'Hardware installeren (info) - Checklist hoofdstuk 3'
85     'Software installeren (info) - Checklist hoofdstuk 4'
86     'Gebruiker inrichten (info)  - Checklist hoofdstuk 5'
87     'Stoppen                   - Menu verlaten'
88 ) # Beschikbare acties
89 readonly ACTIONS=$(( ${#ACTION[@]} - 1 )) # Aantal beschikbare acties
90
91 readonly FILLER=$(printf ' %.0s' {1..5}) # Opvulling
92
93 readonly SCRIPTTXTLEN=65 # Lengte scripttekst
94
95 # Algemeen optioneel (script-common.sh) =====
96 readonly OPT_NEEDS_ARG=false # Optie met een verplicht argument
97 readonly OPTS_SHORT='s' # Korte opties
98 readonly OPTS_LONG='simulate,starter' # Lange opties
99 readonly OPTS_USAGE=$(cat << OPTS_USAGE
100
101     [-s|--simulate] [--starter]
102 OPTS_USAGE
103 ) # Gebruiks-opties
104 readonly OPTS_HELP=$(cat << OPTS_HELP
105
106     -s --simulate      Geen actie, simuleer uitvoeren scripts
107     --starter         Gestart door starter (niet voor normaal gebruik)
108 OPTS_HELP
109 ) # Hulp-opties
110 readonly HELP=$(cat << HELP
111
112     Voer installatiescripts uit via een menu.
113
114     $OPTS_HELP_SC$OPTS_HELP
115
116     Toon een menu waarmee een aantal scripts achter elkaar uitgevoerd worden
117     zodat $DISTID direct te installeren en te gebruiken is.
118
119     Via de Snelzoeker, zoek sta en klik op Start installatie.
120
121     $ASK_FOR_PW_HELPTEXT
122
123     $PART_OF_INSTALL_HELPTEXT
124 HELP
125 ) # Hulptekst
126
127 # Foutcodes =====
128
129 #-----#
130 # Globale variabelen                                     #
131 #-----#
132
```

```
133 # Array, integer =====
134 declare -a SCRIPT # Uit te voeren scripts
135 declare -i SCRIPTS=0 # Aantal uit te voeren scripts
136
137 # Boolean =====
138 declare CONTINUE=false # Doorgaan met scripts uitvoeren
139 declare OPT_SIMULATE=false # Optie simulate opgegeven
140
141 # Overig =====
142 declare ACTIONNUM=0 # Nummer gekozen actie
143 declare ACTIONNUM_PREV=- # Nummer laatsts gekozen actie
144 declare ACTIONNUM2=0 # Nummer gekozen actie menu 2
145 declare ACTIONNUM2_PREV=- # Nummer laatsts gekozen actie menu 2
146 declare SIMTEXT='' # Tekst optie simulate
147
148 #####
149 # Functies #
150 #####
151
152 #-Functie-----#
153 # Naam: controleer_invoer #
154 # Doel: Initiële controles en/of acties. #
155 # Arg.: Geen argumenten. #
156 #-----#
157 controleer_invoer() {
158     $OPT_SIMULATE && SIMTEXT=$TEXT_SIMULATE
159     log "$PROGNAME:$FUNCNAME:$LINENO $SIMTEXT"
160
161     :
162
163     return 0
164 }
165
166 #-Functie-----#
167 # Naam: toon_invoer #
168 # Doel: Toon wat het script gaat doen. #
169 # Arg.: Geen argumenten. #
170 #-----#
171 toon_invoer() {
172     log "$PROGNAME:$FUNCNAME:$LINENO $SIMTEXT"
173
174     :
175
176     return 0
177 }
178
179 #-Functie-----#
180 # Naam: verwerk_invoer #
181 # Doel: Toon en verwerk het menu. #
182 # Arg.: Geen argumenten. #
183 #-----#
184 verwerk_invoer() {
185     log "$PROGNAME:$FUNCNAME:$LINENO $SIMTEXT"
186
187     while true; do
188         toon_menu
189         verwerk_menu
190     done
191
192     return 0
193 }
194
195 #-Functie-----#
196 # Naam: toon_menu #
197 # Doel: Toon wat het script kan doen. #
198 # Arg.: Geen argumenten. #
```

```
199 #-----#
200 toon_menu() {
201     log "$PROGNAME:$FUNCNAME:$LINENO $SIMTEXT"
202
203     local -i action=0
204
205     tput clear
206     info "$FILLER $DISTID installatie menu $SIMTEXT"
207     info
208     for action in "${!ACTION[@]}; do
209         [[ $action -eq 1 || $action -eq 4 ]] && info
210         [[ $action -eq 7 || $action -eq 8 ]] && info
211         info "$FILLER\c"
212         info --bold "$action \c"
213         info "${ACTION[$action]}"
214     done
215     info
216     read -p "Voer uit nummer 0-$ACTIONS (laatste keuze was $ACTIONNUM_PREV) \
217 [$ACTIONS] "
218
219     # Standaard is stoppen als er niets is opgegeven.
220     [[ -z $REPLY ]] && REPLY=$ACTIONS
221
222     # Bepaal actienummer aan de hand van het gekozen nummer.
223     ACTIONNUM=$REPLY
224     ACTIONNUM_PREV=$ACTIONNUM_PREV,'$ACTIONNUM
225
226     return 0
227 }
228
229 #-Functie-----#
230 # Naam: verwerk_menu #
231 # Doel: Verwerk de op het menu gekozen actie. #
232 # Arg.: Geen argumenten. #
233 #-----#
234 verwerk_menu() {
235     log "$PROGNAME:$FUNCNAME:$LINENO $SIMTEXT"
236
237     case $ACTIONNUM in
238         0)
239             verwerk_toon_checklist
240             ;;
241         1)
242             verwerk_installatie_vorbereiden
243             ;;
244         2)
245             verwerk_besturingssysteem_installeren
246             ;;
247         3)
248             verwerk_installatie_voltoeien
249             ;;
250         4)
251             verwerk_hardware_informatie
252             ;;
253         5)
254             verwerk_software_informatie
255             ;;
256         6)
257             verwerk_gebruiker_informatie
258             ;;
259         7)
260             toon_keuze_stop_sc
261             ;;
262         *)
263             case $ACTIONNUM in
264                 *[^[:digit:]]*)
```

```

265             # Keuze bevat niet-numerieke tekens.
266             warning 'Foutieve keuze - alleen cijfers toegestaan.'
267             ;;
268         *)
269             # Keuze bevat alleen numerieke tekens, foute keuze.
270             warning "Foutieve keuze - kies een nummer tussen 0 \
271 en $ACTIONS."
272             ;;
273         esac
274         CONTINUE=true
275         ;;
276     esac
277
278     $CONTINUE && attendee_gebruiker "Druk op de Enter-toets voor het menu \
279 (Enter) "
280
281     return 0
282 }
283
284 #-Functie-----#
285 # Naam: verwerk_toon_checklist (0) #
286 # Doel: Download de checklist-pdfs en toon deze. #
287 # Arg.: Geen argumenten. #
288 #-----#
289 verwerk_toon_checklist() {
290     log "$PROGNAME:$FUNCNAME:$LINENO $SIMTEXT"
291
292     local source='http://karelzimmer.nl/data/linux/pdfs'
293     local target="$HOME/Downloads"
294     # Zoek naar 16.04 of 15.04 of 14.04, zo wordt er altijd een LTS gevonden.
295     local jj_0=$(date +%y) # Dit jaar, bijv. 16
296     local jj_1=$(( $jj_0 - 1 )) # Vorig jaar, bijv. 15
297     local jj_2=$(( $jj_0 - 2 )) # Eervorig jaar, bijv. 14
298     local found_al=false
299     local found_pc=false
300
301     # Download de checklists en toon deze.
302     info
303     info "Download algemene checklist en checklist voor $HOSTNAME..."
304     for (( jaar=$jj_0; jaar>=$jj_2; jaar-- )); do
305         local checklist_al="Checklist installatie $DISTID $jaar.04 LTS \
306 desktop algemeen.pdf"
307         local checklist_pc="Checklist installatie $DISTID $jaar.04 LTS \
308 desktop $HOSTNAME.pdf"
309         wget --quiet \
310             --output-document="$target/$checklist_al" \
311             "$source/$checklist_al" |& $LOGCMD
312         if [[ $? -gt 0 ]]; then
313             rm $VERBOSE \
314                 "$target/$checklist_al" |& $LOGCMD
315         else
316             found_al=true
317             evince "$target/$checklist_al" &> /dev/null &
318             info 'Algemene checklist wordt getoond in een ander venster.'
319         fi
320         wget --quiet \
321             --output-document="$target/$checklist_pc" \
322             "$source/$checklist_pc" |& $LOGCMD
323         if [[ $? -gt 0 ]]; then
324             rm $VERBOSE \
325                 "$target/$checklist_pc" |& $LOGCMD
326         else
327             found_pc=true
328             evince "$target/$checklist_pc" &> /dev/null &
329             info "Checklist voor $HOSTNAME wordt getoond in een ander \
330 venster."

```

```

331         fi
332     done
333
334     if ! $found_al; then
335         info 'Algemene checklist is niet gevonden.'
336         info 'Kijk op http://karelzimmer.nl voor andere versies.'
337     fi
338     if ! $found_pc; then
339         info "Checklist voor $HOSTNAME is niet gevonden."
340         info 'Kijk op http://karelzimmer.nl voor andere versies.'
341     fi
342
343     info
344     info "Volg algemene checklist en, indien gevonden, checklist voor \
345 $HOSTNAME."
346     info 'De checklists kunnen digitaal ingevuld worden,'
347     info 'hiervoor is afdrucken niet persé noodzakelijk.'
348     info
349     info "De checklists zijn opgeslagen in $target."
350     info 'De checklists zijn ook te vinden op de site http://karelzimmer.nl.'
351
352     CONTINUE=true
353
354     return 0
355 }
356
357 #-Functie-----#
358 # Naam: verwerk_installatie_vorbereiden                (1) #
359 # Doel: Verwerk scripts om de installatie voor te bereiden. #
360 # Arg.: Geen argumenten.                                #
361 #-----#
362 verwerk_installatie_vorbereiden() {
363     log "$PROGNAME:$FUNCNAME:$LINENO $SIMTEXT"
364
365     declare -a SCRIPT=(
366         [1]='list-installed\tsoftwarelijst maken'
367         'backup          \tback-up maken'
368     )
369     declare -i SCRIPTS=${#SCRIPT[@]}
370
371     info
372     warning "Zorg dat het medium is aangekoppeld waarop de back-up wordt \
373 geplaatst."
374     info "Als er geen back-up is gemaakt meldt script backup dit als \
375 $STATUS_ERROR."
376     info
377     read -p 'Druk op de Enter-toets om verder te gaan (Enter) '
378
379     verwerk_scripts
380     $CONTINUE || return 0
381
382     info 'Scroll naar boven voor de detailstatus van een script.'
383     info "Als script backup is geëindigd met status $STATUS_ERROR terwijl \
384 er geen"
385     info 'back-up gemaakt hoeft te worden, duidt dit niet op een probleem.'
386
387     return 0
388 }
389
390 #-Functie-----#
391 # Naam: verwerk_besturingssysteem_installeren          (2) #
392 # Doel: Verwerk scripts om de installatie te voltooien. #
393 # Arg.: Geen argumenten.                                #
394 #-----#
395 verwerk_besturingssysteem_installeren() {
396     log "$PROGNAME:$FUNCNAME:$LINENO $SIMTEXT"

```

```
397
398     declare -a SCRIPT=(
399         [1]='wallpapers\tachtergronden toevoegen'
400         'restore \tback-up terugzetten'
401     )
402     declare -i SCRIPTS=${#SCRIPT[@]}
403
404     info
405     warning 'Zorg dat het medium is aangekoppeld waarop de back-up staat.'
406     info "Als er geen back-up is teruggezet meldt script restore dit als \
407 $STATUS_ERROR."
408
409     verwerk_scripts
410     $CONTINUE || return 0
411
412     info 'Scroll naar boven voor de detailstatus van een script.'
413     info "Als script restore is geëindigd met status $STATUS_ERROR \
414 terwijl er geen"
415     info "back-up aanwezig is (bijvoorbeeld bij een nieuwe installatie), \
416 duidt"
417     info 'dit niet op een probleem.'
418
419     return 0
420 }
421
422 #-Functie-----#
423 # Naam: verwerk_installatie_voltooien (3) #
424 # Doel: Verwerk scripts om de installatie te voltooien. #
425 # Arg.: Geen argumenten. #
426 #-----#
427 verwerk_installatie_voltooien() {
428     log "$PROGNAME:$FUNCNAME:$LINENO $SIMTEXT"
429
430     case $HOSTNAME in
431         hugo-*|soe-*)
432             declare -a SCRIPT=(
433                 [1]='printer-install\tprinter/scanner installeren (menu)'
434                 'nas \tnetworkschijf permanent koppelen'
435                 'install \tsoftware installeren/systeem instellen'
436                 'setup \tgebruikersinstellingen wijzigen'
437             )
438             ;;
439         *)
440             declare -a SCRIPT=(
441                 [1]='printer-install\tprinter/scanner installeren (menu)'
442                 'install \tsoftware installeren/systeem instellen'
443                 'setup \tgebruikersinstellingen wijzigen'
444             )
445             ;;
446     esac
447     declare -i SCRIPTS=${#SCRIPT[@]}
448
449     info
450     warning 'Geen printer/scanner om te installeren?'
451     success 'Kies dan bij script printer-install voor Stoppen.'
452     info 'Later een printer/scanner installeren?'
453     info "Via het Activiteitenoverzicht, zoek pri en klik op Printer \
454 Install."
455
456     verwerk_scripts
457     $CONTINUE || return 0
458
459     info 'Scroll naar boven voor de detailstatus van een script.'
460
461     return 0
462 }
```

```
463
464 #-Functie-----#
465 # Naam: verwerk_hardware_informatie (4) #
466 # Doel: Toon de hardware-informatie. #
467 # Arg.: Geen argumenten. #
468 #-----#
469 verwerk_hardware_informatie() {
470     log "$PROGNAME:$FUNCNAME:$LINENO $SIMTEXT"
471
472     if $OPT_SIMULATE; then
473         "$PROGDIR"/printer-install --printer-info --simulate
474     else
475         "$PROGDIR"/printer-install --printer-info
476     fi
477     CONTINUE=true
478
479     return 0
480 }
481
482 #-Functie-----#
483 # Naam: verwerk_software_informatie (5) #
484 # Doel: Toon de software-informatie. #
485 # Arg.: Geen argumenten. #
486 #-----#
487 verwerk_software_informatie() {
488     log "$PROGNAME:$FUNCNAME:$LINENO $SIMTEXT"
489
490     if $OPT_SIMULATE; then
491         "$PROGDIR"/install --install-info --simulate
492     else
493         "$PROGDIR"/install --install-info
494     fi
495     CONTINUE=true
496
497     return 0
498 }
499
500 #-Functie-----#
501 # Naam: verwerk_gebruiker_informatie (6) #
502 # Doel: Toon de gebruiker-informatie. #
503 # Arg.: Geen argumenten. #
504 #-----#
505 verwerk_gebruiker_informatie() {
506     log "$PROGNAME:$FUNCNAME:$LINENO $SIMTEXT"
507
508     if $OPT_SIMULATE; then
509         "$PROGDIR"/setup --setup-info --simulate
510     else
511         "$PROGDIR"/setup --setup-info
512     fi
513     CONTINUE=true
514
515     return 0
516 }
517
518 #-Functie-----#
519 # Naam: verwerk_scripts #
520 # Doel: Verwerk de uit te voeren scripts. #
521 # Arg.: Geen argumenten. #
522 #-----#
523 verwerk_scripts() {
524     log "$PROGNAME:$FUNCNAME:$LINENO $SIMTEXT"
525
526     local -i script=1
527     local -i script_rc=0
528     local scriptname=''
```



```
529     local scripttxt=''
530     local scripttxt_tmp=''
531     local status=''
532     local text=''
533
534     # Toon de scripts die worden uitgevoerd.
535     info
536     if [[ $SCRIPTS -eq 1 ]]; then
537         info "Het volgende script wordt uitgevoerd: $SIMTEXT"
538     else
539         info "De volgende $SCRIPTS scripts worden achter elkaar uitgevoerd: \
540 $SIMTEXT"
541     fi
542     for script in "${!SCRIPT[@]}; do
543
544         # Scriptnaam afsplitsen van de scriptbeschrijving.
545         scriptname=$(echo -e "${SCRIPT[$script]}" | cut --fields=1)
546         scriptdesc=$(echo -e "${SCRIPT[$script]}" | cut --fields=2)
547
548         info "$FILLER$script ${C}$scriptname${N} $scriptdesc"
549     done
550     info
551
552     # Vraag bevestiging om de scripts uit te voeren.
553     if [[ $SCRIPTS -eq 1 ]]; then
554         text='Doorgaan met het script uitvoeren?'
555     else
556         text='Doorgaan met de scripts uitvoeren?'
557     fi
558     read -p "$text (J/n) "
559     case $REPLY in
560         j*|J*|'')
561             CONTINUE=true
562             log 'Gebruiker kiest doorgaan met de scripts uitvoeren.'
563             ;;
564         *)
565             CONTINUE=false
566             return 0
567             ;;
568     esac
569
570     # Voer de scripts uit.
571     for script in "${!SCRIPT[@]}; do
572
573         # Scriptnaam afsplitsen van de scriptbeschrijving.
574         scriptname=$(echo -e "${SCRIPT[$script]}" | cut --fields=1)
575
576         # Voer het script uit en bepaal de status.
577         if $OPT_SIMULATE; then
578             "$PROGDIR"/$scriptname --simulate
579             script_rc=0
580             status="$STATUS_SIMULATE"
581         else
582             "$PROGDIR"/$scriptname --
583             script_rc=$?
584             if [[ $script_rc -eq 0 ]]; then
585                 status="$STATUS_SUCCESS"
586             else
587                 status="$STATUS_ERROR"
588             fi
589         fi
590
591         # Voeg de status toe aan de scriptbeschrijving.
592         # Voeg spaties toe aan de scriptbeschrijving.
593         scripttxt_tmp=${SCRIPT[$script]}$SPACES
594     done
```

```
595     # Maak de scriptbeschrijving op de gewenste maat.
596     scripttxt=${scripttxt_tmp:0:SCRIPTTXTLEN}
597
598     # Voeg de status toe aan de scriptbeschrijving.
599     SCRIPT[$script]=$scripttxt$status
600
601     done
602
603     # Toon uitgevoerde scripts met hun status.
604     info
605     if [[ $SCRIPTS -eq 1 ]]; then
606         info "Het volgende script is uitgevoerd: $SIMTEXT"
607     else
608         info "De volgende $SCRIPTS scripts zijn uitgevoerd: $SIMTEXT"
609     fi
610     for script in "${!SCRIPT[@]}; do
611
612         # Scriptnaam afsplitsen van de scriptbeschrijving.
613         scriptname=$(echo -e "${SCRIPT[$script]}" | cut --fields=1)
614         scriptdesc=$(echo -e "${SCRIPT[$script]}" | cut --fields=2)
615
616         info "$FILLER$script ${C}$scriptname${N} $scriptdesc"
617     done
618     info
619
620     return 0
621 }
622
623 #-Functie-----#
624 # Naam: toon_afsluiten #
625 # Doel: Afsluitende meldingen en/of acties. #
626 # Arg.: Geen argumenten. #
627 #-----#
628 toon_afsluiten() {
629     log "$PROGNAME:$FUNCNAME:$LINENO $SIMTEXT"
630
631     :
632
633     return 0
634 }
635
636 #####
637 # Hoofdlijn #
638 #####
639
640 # init_script
641 {
642     controleer_invoer_sc "$@"
643     controleer_gebruiker_sc $SCRIPT_NEEDS_SUDO "$@"
644     controleer_invoer
645     toon_invoer
646 }
647
648 # verwerk
649 {
650     output=false
651     toon_gestart_sc $output
652     verwerk_invoer
653     toon_gestopt_sc $output
654 }
655
656 # afsl_script
657 {
658     toon_afsluiten
659     toon_afsluiten_sc $output
660     exit_sc 0
```

```
661 }  
662  
663 # Einde script.
```