

```
1: #!/bin/bash
2: # shellcheck source=kz-common.sh
3: #####
4: # Pakket kz installeren.
5: #
6: # Geschreven door Karel Zimmer <info@karelzimmer.nl>.
7: #####
8: PROGRAM_PATH=$(realpath "$(dirname "$0")")
9: if [[ -e "$PROGRAM_PATH"/kz-common.sh ]]; then
10:     source "$PROGRAM_PATH"/kz-common.sh
11: else
12:     wget --output-document=/tmp/kz-common.sh \
13:         https://karelzimmer.nl/data/linux/scripts/kz-common.sh &> /dev/null
14:     source /tmp/kz-common.sh
15: fi
16: PROGRAM_NAME=kz-getdeb
17: DISPLAY_NAME=${PROGRAM_NAME}/kz-/kz }
18: RELEASE_YEAR=2016
19:
20: VERSION_NUMBER=29.04.03
21: VERSION_DATE=2021-10-12
22:
23:
24: #####
25: # Global constants
26: #####
27:
28: readonly RUN_AS_SUPERUSER=false
29: readonly OPTIONS_SHORT=$OPTIONS_SHORT_COMMON
30: readonly OPTIONS_LONG=$OPTIONS_LONG_COMMON
31: readonly USAGE="Gebruik: $DISPLAY_NAME $OPTIONS_USAGE_COMMON
32:     of: wget karelzimmer.nl/kz
33:     bash kz
34:     of: wget karelzimmer.nl/kz; bash kz
35:     of: wget -O- karelzimmer.nl/kz | bash
36:     of: curl -L karelzimmer.nl/kz | bash
37:     of: zoek naar get en klik op Scripts installeren"
38: readonly HELP="Gebruik: $DISPLAY_NAME [OPTIE...]"
39:     of: wget karelzimmer.nl/kz
40:     bash kz
41:     of: wget karelzimmer.nl/kz; bash kz
42:     of: wget -O- karelzimmer.nl/kz | bash
43:     of: curl -L karelzimmer.nl/kz | bash
44:     of: zoek naar get en klik op Scripts installeren
45:
46: Pakket kz installeren.
47:
48: Opties:
49: $OPTIONS_HELP_COMMON"
50:
51: readonly BIN_REPO=$HOME/bin
52: readonly WORK_REPO=$HOME/work
53: readonly DEB_REPO=$HOME/kz-deb
54: readonly DOCS_REPO=$HOME/kz-docs
55: readonly SCRIPTS_REPO=$HOME/kz-scripts
56: readonly UPLOADS_REPO=$HOME/kz-uploads
57: readonly UPLOADS_DISTDIR=$UPLOADS_REPO/dist
58:
59:
60: #####
61: # Global variables
62: #####
63:
64:
65: #####
66: # Functions
67: #####
68:
```

```
69: check_input() {
70:     local -i getopt_rc=0
71:     local parsed=''
72:
73:     parsed=$(
74:         getopt --alternative                \
75:                --options                   "$OPTIONS_SHORT" \
76:                --longoptions              "$OPTIONS_LONG"   \
77:                --name                     "$DISPLAY_NAME"   \
78:                --                          "$@"              \
79:     ) || getopt_rc=$?
80:     if [[ $getopt_rc -ne $SUCCESS ]]; then
81:         printf '%s\n' "$USAGELINE" >&2
82:         exit $ERROR
83:     fi
84:     eval set -- "$parsed"
85:     process_common_options "$@"
86:
87:     while true; do
88:         case $1 in
89:             --)
90:                 shift
91:                 break
92:                 ;;
93:             *)
94:                 shift
95:                 ;;
96:         esac
97:     done
98:
99:     if [[ "$*" ]]; then
100:         TEXT='geen argumenten opgeven'
101:         printf "$DISPLAY_NAME: %s\n%s\n" "$TEXT" "$USAGELINE" >&2
102:         exit $ERROR
103:     fi
104:
105:     # Een non-gui script gestart met optie gui.
106:     if $OPTION_GUI; then
107:         OPTION_GUI=false
108:         DESKTOP_TERMINAL=true
109:     fi
110:
111:     clear -x
112:     check_user
113:     check_on_ac_power
114: }
115:
116:
117: check_on_ac_power() {
118:     local -i on_battery=0
119:
120:     on_ac_power >/dev/null 2>&1 || on_battery=$?
121:     if [[ $on_battery -eq 1 ]]; then
122:         TEXT='De computer gebruikt nu alleen de accu voor de stroomvoorziening.
123:
124: Geadviseerd wordt om de computer aan te sluiten op het stopcontact.'
125:         whiptail --backtitle "$DISPLAY_NAME" \
126:                 --title      'Waarschuwing' \
127:                 --msgbox     "$TEXT" \
128:                 10 71
129:     fi
130: }
131:
132:
133: check_sudo_and_dpkg() {
134:     check_sudo
135:     check_dpkg
136: }
```

```
137:
138:
139: check_sudo() {
140:     if ! sudo -n true 2> /dev/null; then
141:         while true; do
142:             TEXT='
143: Authenticatie is vereist voor het
144: installeren van pakket kz.
145:
146: Voer uw wachtwoord in:'
147:             # Constructie '3>&1 1>&2 2>&3' om stdout en stderr te wisselen.
148:             if printf '%s\n' "$("
149:                 whiptail --backtitle "$DISPLAY_NAME" \
150:                 --title 'Authenticatie nodig' \
151:                 --ok-button 'Verifiëren' \
152:                 --cancel-button 'Annuleren' \
153:                 --passwordbox "$TEXT" \
154:                 12 37 \
155:                 3>&1 1>&2 2>&3
156:             )" | sudo --stdin true 2> /dev/null; then
157:                 break
158:             else
159:                 TEXT='Geen of verkeerd wachtwoord ingevoerd, of
160: Annuleren gekozen.'
161:                 if ! whiptail --backtitle "$DISPLAY_NAME" \
162:                 --title 'Fout' \
163:                 --yes-button 'Opnieuw' \
164:                 --no-button 'Stoppen' \
165:                 --yesno "$TEXT" \
166:                 10 45; then
167:                     TEXT='Programma is gestopt door de gebruiker'
168:                     TERM=ansi whiptail --backtitle "$DISPLAY_NAME" \
169:                     --title 'Gestopt' \
170:                     --infobox "$TEXT" \
171:                     7 42
172:                     exit $SUCCESS
173:                 fi
174:             fi
175:         done
176:     fi
177: }
178:
179:
180: check_dpkg() {
181:     local -i wait_for_apt=5
182:
183:     if ls /snap/core/*/var/cache/debconf/config.dat &> /dev/null; then
184:         while sudo fuser \
185:             /var/{lib/{dpkg,apt/lists},cache/apt/archives}/lock* \
186:             /var/cache/debconf/config.dat \
187:             /snap/core/*/var/cache/debconf/config.dat \
188:             &> /dev/null; do
189:             TEXT="Wacht tot andere pakketbeheerder klaar is (kan even \
190: duren)..."
191:             TERM=ansi whiptail --backtitle "$DISPLAY_NAME" \
192:             --title 'Voortgang' \
193:             --infobox "$TEXT" \
194:             7 65
195:             sleep $wait_for_apt
196:         done
197:     else
198:         while sudo fuser \
199:             /var/{lib/{dpkg,apt/lists},cache/apt/archives}/lock* \
200:             /var/cache/debconf/config.dat \
201:             &> /dev/null; do
202:             TEXT="Wacht tot andere pakketbeheerder klaar is (kan even \
203: duren)..."
204:             TERM=ansi whiptail --backtitle "$DISPLAY_NAME" \
```

```
205:             --title      'Voortgang'      \
206:             --infobox    "$TEXT"          \
207:             7 65
208:         sleep $wait_for_aptd
209:     done
210: fi
211: }
212:
213:
214: process_input() {
215:     local debian_full_name=''
216:     local karel_full_name=''
217:     local ubuntu_full_name=''
218:
219:     install_scriptspackage
220:
221:     # Ben ik een ontwikkelaar?
222:     debian_full_name=$(
223:         getent passwd user           |
224:         awk -F: '{print $5}'         |
225:         awk -F, '{print $1}'         || true
226:     )
227:     karel_full_name=$(
228:         getent passwd karel          |
229:         awk -F: '{print $5}'         |
230:         awk -F, '{print $1}'         || true
231:     )
232:     ubuntu_full_name=$(
233:         getent passwd ubuntu         |
234:         awk -F: '{print $5}'         |
235:         awk -F, '{print $1}'         || true
236:     )
237:     if [[ ($USER = user    && $debian_full_name = 'Debian Live user' ) ||
238:          ($USER = karel   && $karel_full_name   = 'Karel Zimmer'      ) ||
239:          ($USER = ubuntu  && $ubuntu_full_name  = 'Live session user' ) ]]
240:     then
241:         create_development_environment
242:     fi
243: }
244:
245:
246: install_scriptspackage() {
247:     local site_deb=https://karelzimmer.nl/apps/kz/kz_365_all.deb
248:     local temp_deb=''
249:
250:     # Bij Live-sessie de universe repo aanzetten.
251:     if [[ $HOSTNAME = 'ubuntu' && $USER = 'ubuntu' ]]; then
252:         check_sudo_and_dpkg
253:         TEXT='Voeg pakketbron toe...'
254:         TERM=ansi whiptail --backtitle "$DISPLAY_NAME" \
255:             --title      'Voortgang'      \
256:             --infobox    "$TEXT"          \
257:             7 26
258:         sudo add-apt-repository --yes universe |& $LOGCMD
259:
260:         check_sudo_and_dpkg
261:         TEXT='Ververs pakketlijst'
262:         TERM=ansi whiptail --backtitle "$DISPLAY_NAME" \
263:             --title      'Voortgang'      \
264:             --infobox    "$TEXT"          \
265:             7 23
266:         sudo apt-get update |& $LOGCMD
267:     fi
268:
269:     # Installeer pakket.
270:     check_sudo_and_dpkg
271:     TEXT='Download pakket...'
272:     TERM=ansi whiptail --backtitle "$DISPLAY_NAME" \
```

```
273:          --title      'Voortgang'      \
274:          --infobox    "$TEXT"          \
275:          7 22
276:
277: temp_deb=$(mktemp -t "$PROGRAM_NAME-XXXXXXXXXX.deb")
278: wget --output-document="$temp_deb" $site_deb |& $LOGCMD
279:
280: check_sudo_and_dpkg
281: TEXT='Installeer pakket (kan even duren)...'
282: TERM=ansi whiptail --backtitle "$DISPLAY_NAME" \
283:          --title      'Voortgang'      \
284:          --infobox    "$TEXT"          \
285:          7 41
286:
287: check_sudo_and_dpkg
288: sudo DEBIAN_FRONTEND=noninteractive \
289: apt-get \
290: install \
291: --yes \
292: --reinstall \
293: "$temp_deb" |& $LOGCMD
294: rm "$temp_deb" |& $LOGCMD
295: }
296:
297:
298: create_development_environment() {
299:     TEXT='Maak ontwikkelomgeving...'
300:     TERM=ansi whiptail --backtitle "$DISPLAY_NAME" \
301:           --title      'Voortgang'      \
302:           --infobox    "$TEXT"          \
303:           7 29
304:     check_dependencies
305:     process_repos
306:     download_website
307: }
308:
309:
310: check_dependencies() {
311:     if ! command -v git 1> /dev/null; then
312:         # Voor versiebeheersysteem.
313:         sudo apt-get install --yes git |& $LOGCMD
314:     fi
315:     if ! command -v lftp 1> /dev/null; then
316:         # Voor upload website.
317:         sudo apt-get install --yes lftp |& $LOGCMD
318:     fi
319: }
320:
321:
322: process_repos() {
323:     if ! git clone \
324:           https://github.com/karelzimmer/bin.git \
325:           "$BIN_REPO" |& $LOGCMD
326:     then
327:         cd "$BIN_REPO" || exit $ERROR
328:         git config credential.helper store |& $LOGCMD
329:         git pull |& $LOGCMD
330:     fi
331:     if ! git clone \
332:           https://github.com/karelzimmer/work.git \
333:           "$WORK_REPO" |& $LOGCMD
334:     then
335:         cd "$WORK_REPO" || exit $ERROR
336:         git config credential.helper store |& $LOGCMD
337:         git pull |& $LOGCMD
338:     fi
339:     if ! git clone \
340:           https://github.com/karelzimmer/kz-deb.git \
```

```

341:         "$DEB_REPO" |& $LOGCMD
342:     then
343:         cd "$DEB_REPO" || exit $ERROR
344:         git config credential.helper store |& $LOGCMD
345:         git pull |& $LOGCMD
346:     fi
347:     if ! git clone \
348:         https://github.com/karelzimmer/kz-docs.git \
349:         "$DOCS_REPO" |& $LOGCMD
350:     then
351:         cd "$DOCS_REPO" || exit $ERROR
352:         git config credential.helper store |& $LOGCMD
353:         git pull |& $LOGCMD
354:     fi
355:     if ! git clone \
356:         https://github.com/karelzimmer/kz-scripts.git \
357:         "$SCRIPTS_REPO" |& $LOGCMD
358:     then
359:         cd "$SCRIPTS_REPO" || exit $ERROR
360:         git pull |& $LOGCMD
361:     fi
362:     if ! git clone \
363:         https://github.com/karelzimmer/kz-uploads.git \
364:         "$UPLOADS_REPO" |& $LOGCMD
365:     then
366:         cd "$UPLOADS_REPO" || exit $ERROR
367:         git config credential.helper store |& $LOGCMD
368:         git pull |& $LOGCMD
369:     fi
370: }
371:
372:
373: download_website() {
374:     local ftp_set='set ssl:verify-certificate no'
375:     local ftp_from=/httpdocs
376:     local ftp_to=$UPLOADS_DISTDIR
377:     local ftp_opts='--delete --verbose'
378:     local ftp_cmd="mirror $ftp_opts $ftp_from $ftp_to; exit"
379:     local ftp_host=server106.hosting2go.nl
380:     local ftp_user=kzimmer
381:     local ftp_login=$HOME/.kz-$ftp_host
382:
383:     if ! [[ -f $ftp_login ]]; then
384:         # < /dev/tty want FD 1 al in gebruik bij 'wget -O- karelzim... | bash'.
385:         read -rsp "FTP-wachtwoord voor $ftp_host: " < /dev/tty
386:         printf '%s\n' "$REPLY" > "$ftp_login"
387:         printf '\n'
388:         chmod 'u=rw,g=,o=' "$ftp_login" |& $LOGCMD
389:     fi
390:     TEXT='Download website (kan even duren)...'
391:     TERM=ansi whiptail --backtitle "$DISPLAY_NAME" \
392:         --title 'Voortgang' \
393:         --infobox "$TEXT" \
394:         7 40
395:     if ! lftp --user "$ftp_user" \
396:         --password "$(cat "$ftp_login")" \
397:         -e "$ftp_set; $ftp_cmd" \
398:         "$ftp_host" |& $LOGCMD; then
399:         rm "$ftp_login"
400:         warning 'Downloaden website is mislukt.'
401:         warning 'Controleer de log...!'
402:         return $WARNING
403:     fi
404: }
405:
406:
407: term_script() {
408:     local search=''

```

```
409:
410: search='/usr/share/doc/kz/Checklist installatie'
411: cp --update "$search"*{desktop,server}.pdf "$HOME" |& $LOGCMD || true
412: cp --update                                     \
413: "$search"*{desktop,server}" $HOSTNAME".pdf     \
414: "$HOME"                                         |& $LOGCMD || true
415:
416: TEXT='Pakket kz is succesvol geïnstalleerd.
417:
418: Volg nu de stappen zoals beschreven in de
419: Checklist installatie Ubuntu, of
420: Checklist installatie Debian.
421:
422: Deze checklists zijn te vinden:
423: - in de Persoonlijke map
424: - in de map /usr/share/doc/kz
425: - op de site https://karelzimmer.nl,
426: onder Linux'
427: height=17
428:
429: if ! $DESKTOP_TERMINAL; then
430:     TEXT=$TEXT"
431:
432: Typ 'exit' om dit venster te sluiten."
433:     height=19
434: fi
435: TERM=ansi whiptail --backtitle "$DISPLAY_NAME" \
436:                   --title 'Gereed'           \
437:                   --infobox "$TEXT"         \
438:                   $height 45
439:     exit $SUCCESS
440: }
441:
442:
443: #####
444: # Main line
445: #####
446:
447: main() {
448:     init_script "$@"
449:     check_input "$@"
450:     process_input
451:     term_script
452: }
453:
454:
455: main "$@"
456:
457:
458: # EOF
```