

```
1: #!/bin/bash
2: # shellcheck source=kz-common.sh
3: #####
4: # Installatiemenu.
5: #
6: # Geschreven door Karel Zimmer <info@karelzimmer.nl>.
7: #####
8: PROGRAM_PATH=$(realpath "$(dirname "$0")")
9: source "$PROGRAM_PATH"/kz-common.sh
10: PROGRAM_NAME=kz-menu
11: DISPLAY_NAME=${PROGRAM_NAME/kz-/kz }
12: RELEASE_YEAR=2015
13:
14: VERSION_NUMBER=36.01.00
15: VERSION_DATE=2021-10-12
16:
17:
18: #####
19: # Global constants
20: #####
21:
22: readonly RUN_AS_SUPERUSER=false
23: readonly OPTIONS_SHORT=$OPTIONS_SHORT_COMMON
24: readonly OPTIONS_LONG=$OPTIONS_LONG_COMMON
25: readonly USAGE="Gebruik: $DISPLAY_NAME $OPTIONS_USAGE_COMMON"
26: readonly HELP="Gebruik: $DISPLAY_NAME [OPTIE...]"
27:
28: Installatiemenu.
29:
30: Opties:
31: $OPTIONS_HELP_COMMON"
32:
33: DASHES=$(printf '%.0s=' {1..79}); readonly DASHES
34: SPACES=$(printf '%.0s ' {1..79}); readonly SPACES
35: readonly MENU1_TITLE=Installatiemenu
36: readonly MENU1_HEAD="\
37: ${DASHES:0:57}
38: =${SPACES:0:20}$MENU1_TITLE${SPACES:0:20}=
39: ${DASHES:0:57}"
40: readonly -a ACTIONS1=(
41:     [1]='Installatie voorbereiden'
42:     'Installatie uitvoeren'
43:     'Installatie afronden'
44:     'Gebruiker inrichten'
45:     'Opdrachtenmenu'
46:     'Stoppen'
47: )
48: readonly -a COMMANDS1=(
49:     [1]='Checklist hoofdstuk 1'
50:     'Checklist hoofdstuk 2'
51:     'Checklist hoofdstuk 3'
52:     'Checklist hoofdstuk 4'
53:     'Opdrachten uitvoeren'
54:     'Menu verlaten'
55: )
56: readonly MENU1_LINES=${#ACTIONS1[@]}
57:
58:
59: readonly MENU2_TITLE=Opdrachtenmenu
60: readonly MENU2_HEAD="\
61: ${DASHES:0:57}
62: =${SPACES:0:20}$MENU2_TITLE${SPACES:0:21}=
63: ${DASHES:0:57}"
64: readonly -a ACTIONS2=(
65:     [1]='Back-up maken'
66:     'Pakket installeren'
67:     'Apps installeren'
68:     'Installatie-instructies tonen'
```

```
69:      'Verwijder-instructies tonen'
70:      'Applijst maken'
71:      'Netwerkschijf aankoppelen'
72:      'Back-up terugzetten'
73:      'Apps instellen'
74:      'Instell-instructies tonen'
75:      'Reset-instructies tonen'
76:      'Instellijst maken'
77:      'Terug'
78:  )
79:  readonly -a COMMANDS2=(
80:    [1]='kz backup'
81:    'kz getdeb'
82:    'kz install'
83:    'kz install --info'
84:    'kz install --remove'
85:    'kz install --list'
86:    'kz nas'
87:    'kz restore'
88:    'kz setup'
89:    'kz setup --info'
90:    'kz setup --reset'
91:    'kz setup --list'
92:    'Naar vorig menu'
93:  )
94:  readonly MENU2_LINES=${#ACTIONS2[@]}
95:
96:
97: #####
98: # Global variables
99: #####
100:
101: declare -a EXEC=''
102: declare    COMMANDS_EXECUTED=false
103: declare -i NUMBER_OF_COMMANDS=0
104:
105:
106: #####
107: # Functions
108: #####
109:
110: check_input() {
111:     local -i getopt_rc=0
112:     local    parsed=''
113:
114:     parsed=$(
115:         getopt  --alternative          \
116:                 --options             "$OPTIONS_SHORT" \
117:                 --longoptions         "$OPTIONS_LONG"  \
118:                 --name                "$DISPLAY_NAME"  \
119:                 --                    "$@"
120:     ) || getopt_rc=$?
121:     if [[ $getopt_rc -ne $SUCCESS ]]; then
122:         printf '%s\n' "$USAGELINE" >&2
123:         exit $ERROR
124:     fi
125:     eval set -- "$parsed"
126:     process_common_options "$@"
127:
128:     while true; do
129:         case $1 in
130:             --)
131:                 shift
132:                 break
133:             ;;
134:             *)
135:                 shift
136:                 ;;
```

```
137:     esac
138: done
139:
140: if [[ "$*" ]]; then
141:     TEXT='geen argumenten opgeven'
142:     printf "$DISPLAY_NAME: %s\n%s\n" "$TEXT" "$USAGELINE" >&2
143:     exit $ERROR
144: fi
145:
146: check_user
147: }
148:
149:
150: process_input() {
151:     if $OPTION_GUI; then
152:         process_menu1_gui
153:     else
154:         process_menu1_tui
155:     fi
156: }
157:
158:
159: process_menu1_gui() {
160:     local -i zenity_rc=0
161:
162:     while true; do
163:         TITLE=$MENU1_TITLE
164:         # Constructie '2> >($LOGCMD)' om stderr naar de log te krijgen.
165:         # Voorbeeld: Unable to init server: Kon niet verbinden:
166:         #             Verbinding is geweigerd
167:         #             en: (zenity:47712): Gtk-WARNING **: 10:35:49.339:
168:         #             cannot open display:
169:         REPLY=$(
170:             zenity --list \
171:                 --radiolist \
172:                 --width 480 \
173:                 --height 250 \
174:                 --title "$TITLE" \
175:                 --text 'Selecteer een keuze:' \
176:                 --ok-label 'Verder' \
177:                 --cancel-label 'Afsluiten' \
178:                 --column 'Keuze' \
179:                 --column 'Nr.' \
180:                 --column 'Actie' \
181:                 --column 'Toelichting' \
182:                 TRUE 1 "${ACTIONS1[1]}" "${COMMANDS1[1]}" \
183:                 FALSE 2 "${ACTIONS1[2]}" "${COMMANDS1[2]}" \
184:                 FALSE 3 "${ACTIONS1[3]}" "${COMMANDS1[3]}" \
185:                 FALSE 4 "${ACTIONS1[4]}" "${COMMANDS1[4]}" \
186:                 FALSE 5 "${ACTIONS1[5]}" "${COMMANDS1[5]}" 2> >($LOGCMD)
187:         ) || zenity_rc=$?
188:         if [[ $zenity_rc -ne 0 ]]; then
189:             exit $SUCCESS
190:         fi
191:         process_menu1_choice
192:     done
193: }
194:
195:
196: process_menu1_tui() {
197:     local action1=''
198:
199:     while true; do
200:
201:         # Toon menu.
202:         clear -x
203:         printf '%s\n\n' "$MENU1_HEAD"
204:         for action1 in "${!ACTIONS1[@]}"; do
```

```
205:         printf '%2s. %-29.29s - %s\n' "$action1" "${ACTIONS1[$action1]}" \
206: "${COMMANDS1[$action1]}"
207:         done
208:         printf '\n'
209:
210:         while true; do
211:             read -rp "Voer uit nummer [1-${MENU1_LINES-1}], $MENU1_LINES]: "
212:             case $REPLY in
213:                 *[^[:digit:]]*)
214:                     continue
215:                     ;;
216:                 *)
217:                     if [[ $REPLY -ge 1 && $REPLY -le $MENU1_LINES ]] \
218:                         &> /dev/null; then
219:                         break
220:                     else
221:                         continue
222:                     fi
223:                     ;;
224:             esac
225:         done
226:         process_menu1_choice
227:         if $COMMANDS_EXECUTED; then
228:             read -rp '
229: Druk op de Enter-toets voor het menu [Enter]: '
230:             fi
231:             COMMANDS_EXECUTED=false
232:         done
233:     }
234:
235:
236: process_menu1_choice() {
237:     case $REPLY in
238:         1)
239:             process_menu1_choice1
240:             ;;
241:         2)
242:             process_menu1_choice2
243:             ;;
244:         3)
245:             process_menu1_choice3
246:             ;;
247:         4)
248:             process_menu1_choice4
249:             ;;
250:         5)
251:             process_menu2
252:             ;;
253:         6)
254:             exit $SUCCESS
255:             ;;
256:         *)
257:             error 'Niet afgehandelde keuze.'
258:             exit $ERROR
259:             ;;
260:     esac
261: }
262:
263:
264: process_menu1_choice1() {
265:     warning '
266: Zorg dat het medium is aangekoppeld waarop de back-up wordt geplaatst.'
267:
268:     TITLE="${ACTIONS1[1]}"
269:     EXEC=(
270:         [1]="${ACTIONS2[5]}#${COMMANDS2[5]}"
271:         "${ACTIONS2[11]}#${COMMANDS2[11]}"
272:         "${ACTIONS2[1]}#${COMMANDS2[1]}"
```

```
273:     )
274:
275:     process_commands
276: }
277:
278:
279: process_menu1_choice2() {
280:     TITLE="${ACTIONS1[2]}"
281:     TEXT='
282: Volg de stappen zoals beschreven in Checklist installatie, hoofdstuk 2.
283: Deze Checklist is te vinden op de site karelzimmer.nl onder Linux,
284: en in map /usr/share/doc/kz.
285:
286: Komt globaal neer op:
287: 1. Download een beeldbestand (.iso) vanaf nl.releases.ubuntu.com.
288: 2. Maak een opstartbare Ubuntu Live USB-stick of DVD aan.
289: 3. Start de computer op vanaf deze USB-stick of DVD.
290: 4. Installeer Ubuntu.'
291:     info "$TEXT"
292:     if ! $OPTION_GUI; then
293:         read -rp '
294: Druk op de Enter-toets voor het menu [Enter]: '
295:         fi
296:     }
297:
298:
299: process_menu1_choice3() {
300:     warning '
301: Zorg dat het medium is aangekoppeld waarop de back-up staat.'
302:
303:     TITLE="${ACTIONS1[3]}"
304:     if [[ ($HOSTNAME = pc-van-soe && $USER = soe ) ||
305:         ($HOSTNAME = pc-van-hugo && $USER = hugo) ]]; then
306:         EXEC=(
307:             [1]="${ACTIONS2[6]}#${COMMANDS2[6]}"
308:             "${ACTIONS2[7]}#${COMMANDS2[7]}"
309:             "${ACTIONS2[2]}#${COMMANDS2[2]}"
310:             "${ACTIONS2[3]}#${COMMANDS2[3]}"
311:         )
312:     else
313:         EXEC=(
314:             [1]="${ACTIONS2[7]}#${COMMANDS2[7]}"
315:             "${ACTIONS2[2]}#${COMMANDS2[2]}"
316:             "${ACTIONS2[3]}#${COMMANDS2[3]}"
317:         )
318:     fi
319:     process_commands
320: }
321:
322:
323: process_menu1_choice4() {
324:     TITLE="${ACTIONS1[4]}"
325:     EXEC=(
326:         [1]="${ACTIONS2[8]}#${COMMANDS2[8]}"
327:         "${ACTIONS2[9]}#${COMMANDS2[9]}"
328:     )
329:     process_commands
330: }
331:
332:
333: process_menu2() {
334:     if $OPTION_GUI; then
335:         process_menu2_gui
336:     else
337:         process_menu2_tui
338:     fi
339: }
340:
```

```

341:
342: process_menu2_gui() {
343:     local -i zenity_rc=0
344:
345:     while true; do
346:         TITLE=$MENU2_TITLE
347:         REPLY=$(
348:             zenity  --list                               \
349:                 --radiolist                             \
350:                 --width          480                   \
351:                 --height         420                   \
352:                 --title          "$TITLE"              \
353:                 --text           'Selecteer een keuze:' \
354:                 --ok-label       'Verder'              \
355:                 --cancel-label  'Terug'               \
356:                 --column        'Keuze'               \
357:                 --column        'Nr.'                 \
358:                 --column        'Actie'               \
359:                 --column        'Opdracht'            \
360:                 TRUE   1  "${ACTIONS2[1]}"  "${COMMANDS2[1]}" \
361:                 FALSE  2  "${ACTIONS2[2]}"  "${COMMANDS2[2]}" \
362:                 FALSE  3  "${ACTIONS2[3]}"  "${COMMANDS2[3]}" \
363:                 FALSE  4  "${ACTIONS2[4]}"  "${COMMANDS2[4]}" \
364:                 FALSE  5  "${ACTIONS2[5]}"  "${COMMANDS2[5]}" \
365:                 FALSE  6  "${ACTIONS2[6]}"  "${COMMANDS2[6]}" \
366:                 FALSE  7  "${ACTIONS2[7]}"  "${COMMANDS2[7]}" \
367:                 FALSE  8  "${ACTIONS2[8]}"  "${COMMANDS2[8]}" \
368:                 FALSE  9  "${ACTIONS2[9]}"  "${COMMANDS2[9]}" \
369:                 FALSE 10 "${ACTIONS2[10]}" "${COMMANDS2[10]}" \
370:                 FALSE 11 "${ACTIONS2[11]}" "${COMMANDS2[11]}" \
371:                 FALSE 12 "${ACTIONS2[12]}" "${COMMANDS2[12]}" \
372:                 2> >($LOGCMD)
373:         ) || zenity_rc=$?
374:         if [[ $zenity_rc -ne 0 ]]; then
375:             break
376:         fi
377:         process_menu2_choice
378:     done
379: }
380:
381:
382: process_menu2_tui() {
383:     local action2=''
384:     local exit_menu2=false
385:
386:     while true; do
387:
388:         # Toon menu.
389:         clear -x
390:         printf '%s\n\n' "$MENU2_HEAD"
391:         for action2 in "${!ACTIONS2[@]}"; do
392:             printf '%2s. %-29.29s - %s\n' "$action2" "${ACTIONS2[$action2]}" \
393:                 "${COMMANDS2[$action2]}"
394:         done
395:         printf '\n'
396:
397:         while true; do
398:             read -rp "Voer uit nummer [1-${MENU2_LINES-1}], $MENU2_LINES]: "
399:             case $REPLY in
400:                 *[^[:digit:]]*)
401:                     continue
402:                 ;;
403:                 *)
404:                     if [[ $REPLY -ge 1 && $REPLY -le $MENU2_LINES ]] \
405:                         2> /dev/null; then
406:                         break
407:                     else
408:                         continue

```

```
409:             fi
410:             ;;
411:         esac
412:     done
413:     process_menu2_choice
414:     if $exit_menu2; then
415:         break
416:     fi
417:     if $COMMANDS_EXECUTED; then
418:         read -rp '
419: Druk op de Enter-toets voor het menu [Enter]: '
420:         fi
421:         COMMANDS_EXECUTED=false
422:     done
423: }
424:
425:
426:
427: process_menu2_choice() {
428:     case $REPLY in
429:         1)
430:             TEXT='
431: Zorg dat het medium is aangekoppeld waarop de back-up wordt geplaatst.'
432:             warning "$TEXT"
433:             TITLE="${ACTIONS2[1]}"
434:             EXEC=(
435:                 [1]="${ACTIONS2[1]}#${COMMANDS2[1]}"
436:             )
437:             process_commands
438:             ;;
439:         2)
440:             TITLE="${ACTIONS2[2]}"
441:             EXEC=(
442:                 [1]="${ACTIONS2[2]}#${COMMANDS2[2]}"
443:             )
444:             process_commands
445:             ;;
446:         3)
447:             TITLE="${ACTIONS2[3]}"
448:             EXEC=(
449:                 [1]="${ACTIONS2[3]}#${COMMANDS2[3]}"
450:             )
451:             process_commands
452:             ;;
453:         4)
454:             TITLE="${ACTIONS2[3]}"
455:             EXEC=(
456:                 [1]="${ACTIONS2[4]}#${COMMANDS2[4]}"
457:             )
458:             process_commands
459:             ;;
460:         5)
461:             TITLE="${ACTIONS2[5]}"
462:             EXEC=(
463:                 [1]="${ACTIONS2[5]}#${COMMANDS2[5]}"
464:             )
465:             process_commands
466:             ;;
467:         6)
468:             TITLE="${ACTIONS2[6]}"
469:             EXEC=(
470:                 [1]="${ACTIONS2[6]}#${COMMANDS2[6]}"
471:             )
472:             process_commands
473:             ;;
474:         7)
475:             TITLE="${ACTIONS2[7]}"
476:             EXEC=(
```

```

477:             [1]="${ACTIONS2[7]}#${COMMANDS2[7]}"
478:         )
479:     process_commands
480:     ;;
481: 8)
482:     TITLE="${ACTIONS2[8]}"
483:     TEXT='
484: Zorg dat het medium is aangekoppeld waarop de back-up staat.'
485:     warning "$TEXT"
486:     EXEC=(
487:         [1]="${ACTIONS2[8]}#${COMMANDS2[8]}"
488:     )
489:     process_commands
490:     ;;
491: 9)
492:     TITLE="${ACTIONS2[9]}"
493:     EXEC=(
494:         [1]="${ACTIONS2[9]}#${COMMANDS2[9]}"
495:     )
496:     process_commands
497:     ;;
498: 10)
499:     TITLE="${ACTIONS2[10]}"
500:     EXEC=(
501:         [1]="${ACTIONS2[10]}#${COMMANDS2[10]}"
502:     )
503:     process_commands
504:     ;;
505: 11)
506:     TITLE="${ACTIONS2[11]}"
507:     EXEC=(
508:         [1]="${ACTIONS2[11]}#${COMMANDS2[11]}"
509:     )
510:     process_commands
511:     ;;
512: 12)
513:     TITLE="${ACTIONS2[12]}"
514:     EXEC=(
515:         [1]="${ACTIONS2[12]}#${COMMANDS2[12]}"
516:     )
517:     process_commands
518:     ;;
519: 13)
520:     exit_menu2=true
521:     ;;
522: *)
523:     error 'Niet afgehandelde keuze.'
524:     exit $ERROR
525:     ;;
526: esac
527: }
528:
529:
530: process_commands() {
531:     local -i exec=0
532:     local action=''
533:     local command=''
534:
535:     NUMBER_OF_COMMANDS=${#EXEC[@]}
536:     if [[ $NUMBER_OF_COMMANDS -eq 1 ]]; then
537:         COMMANDS_EXECUTED=true
538:     else
539:         show_commands_before_execution
540:     fi
541:     if $COMMANDS_EXECUTED; then
542:         for exec in "${!EXEC[@]}"; do
543:             action=$( printf '%s' "${EXEC[$exec]}" | awk -F# '{print $1}' )
544:             command=$( printf '%s' "${EXEC[$exec]}" | awk -F# '{print $2}' )

```



```
545:         execute_command
546:     done
547: fi
548: if $COMMANDS_EXECUTED && [[ $NUMBER_OF_COMMANDS -gt 1 ]]; then
549:     show_commands_after_execution
550: fi
551: }
552:
553:
554: show_commands_before_execution() {
555:     local      prompt=''
556:     local      -i  exec=0
557:     local      action=''
558:     local      command=''
559:
560:     TEXT=""
561: De volgende $NUMBER_OF_COMMANDS acties worden achter elkaar uitgevoerd:
562: "
563:     for exec in "${!EXEC[@]}"; do
564:         action=$( printf '%s' "${EXEC[$exec]}" | awk -F# '{print $1}')
565:         command=$( printf '%s' "${EXEC[$exec]}" | awk -F# '{print $2}')
566:         TEXT=$TEXT"
567: $exec. $action"
568:     done
569:
570:     prompt='
571: Doorgaan en deze acties uitvoeren?'
572:     if $OPTION_GUI; then
573:         TEXT=$TEXT"
574:
575: $prompt"
576:         if zenity --question          \
577:                 --no-markup         \
578:                 --width             400 \
579:                 --height            100 \
580:                 --title             "$TITLE" \
581:                 --text              "$TEXT" \
582:                 --ok-label          'Ja' \
583:                 --cancel-label     'Nee' \
584:                 >> >($LOGCMD); then
585:             COMMANDS_EXECUTED=true
586:         else
587:             COMMANDS_EXECUTED=false
588:         fi
589:     else
590:         printf '%b\n' "$TEXT"
591:         while true; do
592:             read -rp "$prompt [J/n]: "
593:             case $REPLY in
594:                 j*|J*|'')
595:                 COMMANDS_EXECUTED=true
596:                 break
597:                 ;;
598:                 n*|N*)
599:                 COMMANDS_EXECUTED=false
600:                 break
601:                 ;;
602:                 *)
603:                 continue
604:                 ;;
605:             esac
606:         done
607:     fi
608: }
609:
610: execute_command() {
611:     local      -i  cmd_rc=0
612:     local      exec=''
```

```
613: local prompt=''
614:
615: exec=${command/kz /kz-}
616: cmd=$(printf '%s' "$exec" | cut --delimiter=' ' --fields=1)
617: arg=$(
618:     printf '%s' "$exec" |
619:     cut --delimiter=' ' \
620:     --fields=2- \
621:     --only-delimited
622: )
623: case $exec in
624:     kz-backup|kz-install|kz-install*--list|kz-nas|kz-restore)
625:         # Zogenaamde sudo-scripts.
626:         if $OPTION_GUI; then
627:             pkexec "$PROGRAM_PATH/$cmd" "$arg" --gui -- || cmd_rc=$?
628:             log_rc cmd_rc?
629:         else
630:             prompt="Authenticatie is vereist om $command uit te voeren.
631: [sudo] wachtwoord voor %p: "
632:             sudo --prompt="$prompt" \
633:                 "$PROGRAM_PATH/$cmd" "$arg" -- || cmd_rc=$?
634:             log_rc $cmd_rc
635:         fi
636:         ;;
637:     kz-getdeb|kz-setup|*--info|kz-setup*--list|*--remove|*--reset)
638:         # Zogenaamde non-sudo-scripts.
639:         if $OPTION_GUI; then
640:             "$PROGRAM_PATH/$cmd" "$arg" --gui -- || cmd_rc=$?
641:             log_rc $cmd_rc
642:         else
643:             "$PROGRAM_PATH/$cmd" "$arg" -- || cmd_rc=$?
644:             log_rc $cmd_rc
645:         fi
646:         ;;
647:     *)
648:         warning "Niet af te handelen opdracht: $command"
649:         log_rc $ERROR
650:         ;;
651: esac
652: }
653:
654:
655: log_rc() {
656:     local rc=${1:-rc?}
657:
658:     log "$command: rc=$rc"
659: }
660:
661:
662: show_commands_after_execution() {
663:     local action=''
664:     local command=''
665:     local -i exec=0
666:     local prompt=''
667:     local temp_text_file=''
668:
669:     temp_text_file=$(mktemp -t "$PROGRAM_NAME-XXXXXXXXXX.txt")
670:     printf '\n%s\n\n' "De volgende $NUMBER_OF_COMMANDS acties zijn \
671: uitgevoerd:" > "$temp_text_file"
672:     for exec in "${!EXEC[@]}; do
673:         action=$( printf '%s' "${EXEC[$exec]}" | awk -F# '{print $1}')
674:         command=$(printf '%s' "${EXEC[$exec]}" | awk -F# '{print $2}')
675:         printf '%2d. %-25s\n' "$exec" "$action" >> "$temp_text_file"
676:     done
677:     if $OPTION_GUI; then
678:         if ! zenity --text-info \
679:             --width 400 \
680:             --height 200 \
```

```
681:          --title          "$TITLE"          \
682:          --filename        "$temp_text_file"  \
683:          --font             'Ubuntu Mono 12'  \
684:          --ok-label        'OkÃ©'           \
685:          --cancel-label    'Annuleren'       2> >($LOGCMD); then
686:      true
687:      fi
688:  else
689:      cat "$temp_text_file"
690:  fi
691:  rm "$temp_text_file"
692: }
693:
694:
695: term_script() {
696:     exit $SUCCESS
697: }
698:
699:
700: #####
701: # Main line
702: #####
703:
704: main() {
705:     init_script "$@"
706:     check_input "$@"
707:     process_input
708:     term_script
709: }
710:
711:
712: main "$@"
713:
714:
715: # EOF
```