

```

1 #!/bin/bash
2
3 run=1
4
5 # Verifica che il numero di argomenti sia almeno 2
6 if (( $# < 2 )); then
7     echo "Uso: $0
8         <file_di_configurazione> <comando> [argomenti_comando...]"
9     run=0
10 fi
11
12 # Imposta le variabili dagli argomenti
13 config_file=$1
14 command=$2
15 shift 2
16
17 # Controlla se il file di configurazione esiste e se è leggibile
18 if [[ ! -e $config_file ]]; then
19     echo "Il file di configurazione $config_file non esiste"
20     run=0
21 fi
22
23 if [[ ! -f $config_file ]]; then
24     echo "$config_file non è un file"
25     run=0
26 fi
27
28 if [[ ! -r $config_file ]]; then
29     echo "Il file $config_file non è leggibile"
30     run=0
31 fi
32
33 # Inizia l'esecuzione se tutte le condizioni sono soddisfatte
34 if (( run == 1 )); then
35
36     # Fase 0: crea la directory temporanea di lavoro
37     container_filepath=$(mktemp -d)

```

```

38
39     # Fase 1: crea l'ambiente di lavoro per il container
40     IFS=$'\n'
41     for line in $(cat "$config_file"); do
42         origin=$(echo "$line" | cut -d " " -f 1)
43         destination=$(echo "$line" | cut -d " " -f 2)
44         dest_path="$container_filepath$destination"
45
46         # Caso directory
47         if [[ -d $origin ]]; then
48             mkdir -p "$dest_path"
49             bindfs --no-allow-other "$origin" "$dest_path"
50         fi
51
52         # Caso file
53         if [[ -f $origin ]]; then
54             # Crea tutte le directory necessarie ricorsivamente,
55             # e cancella l'ultima (che dovrebbe essere un file invece)
56             # es: /bin/ls -> devo creare /bin/
57             ls come cartella, poi cancellare /ls che diventerà un file
58             mkdir -p "$(dirname "$dest_path")"
59             rmdir "$dest_path"
60             cp -p "$origin" "$dest_path"
61         fi
62     done
63
64     # Fase 2: esegue il comando nell'ambiente isolato
65     if (( $# == 0 )); then
66         fakechroot chroot "$container_filepath" "$command"
67     else
68         fakechroot chroot "$container_filepath" "$command" "$@"
69     fi

```

→ ESTAR NOS UN PROGRAMA DE ON AMBIENTE ISOTILO
SENZI RICHIE DIRE PRACTICO DE ADMINISTRAÇÃO

- Punti di controllo su avvertenze input \rightarrow deve essere più di due

↳ CÓDIGO "USO: \$0 < FILE OR CON-TR > COMMANDS [ARG....]
 ↳ STARTEZ
 ↳ TUTORIAL
 ↳ RUN => first => NC
 ↳ norm script → ./PROVA.SH
 ↳ PROTO
 ↳ ARG
 ↳ (\$1)
 ↳ \$2
 ↳ \$3
 ↳ altri comandi
 ↳ parabolo
 ↳ f(x) = ax^2 + bx + c

base i centrale, se possiede corrente neta poss border audit
config-file \rightarrow $\{1, config=2\}$

Shift 2 → SPOSTH ORG → DFF but → DRG1, ARU2, DRG3, DRG4, DRG5
\$1 \$2 \$3 \$4 \$5

Dopo SHIFT \rightarrow DRT3, DRT4, DRT5 (5-2)
\$1 \$2 \$3 SMT

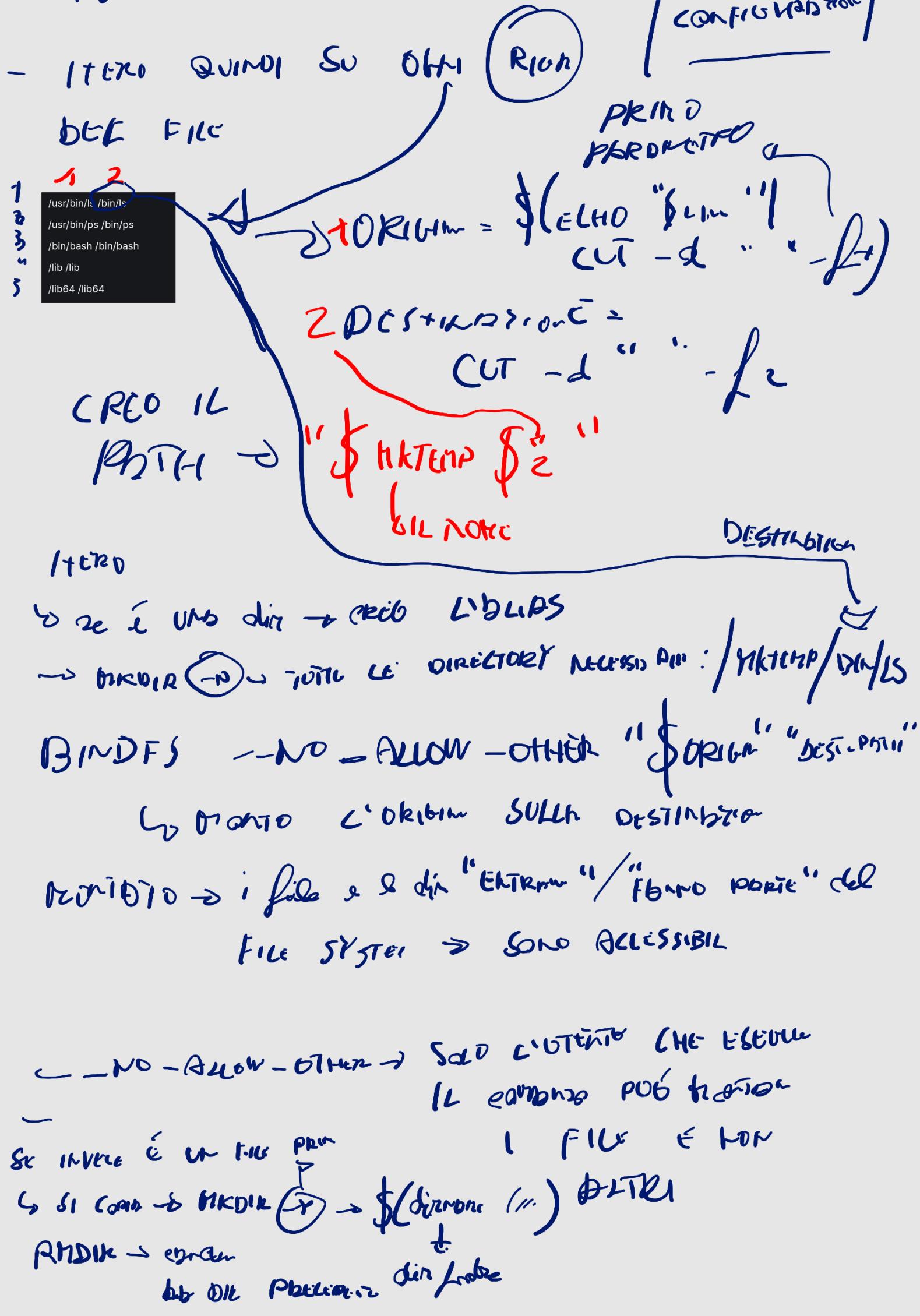
CONTROLLI SUL FILE DI CONFIGURAZIONE

→ exist → -e , size un file → -f , size LÉTHIB'LE → -R
→ quando esiste QUESTA costruzione → UTILIZZO LO IDR RUN → per "ALYDRE"
ECCESIONI

→ PREB OMB DIRECTOR TEMPORAL

MKTEMP (-D) → DIR<CREATE → NOT ON FILE

— Pten Charcot → PROD UN ALS }
Pten FLX → CEP10 } Colloco NEU
FLX DI



CP (-p) corso IL FILE DI ORIGIN ~ DEST_PDF

Costruzione i PERMESSI

perché ci servono SHIFT

↳ con SHIFT ci concentriamo SOLO sui COMBINATIVI
Costruttore FILE DI Config È COMBINATO PRINCIPALE

↳ \$@ → emette TUTTI → se ci sono

SE non ci sono > No

FOKECHROOT CHROOT " " " \$cat" "f@"
{ ANDROID
Punto

↓
AMBICHI
I SOGGETTI

