

# USB KEY BUILDING FOR CHIPSEC AND SECUREBOOT CHECKS v0.1 (04/2019)

A Help to build your own ChipSec and SecureBoot USB keys

## Final USB keys

- **USB KEY 1** : live Debian distribution to launch **ChipSec** from the computer to analyze
- **USB KEY 2** : contains **SecureBoot** keys to import, tool to import your own trust keys and to check importation

## Linux Tools to install before generating the USB keys

```
sudo apt-get install debootstrap
sudo apt-get install sbsigntool
sudo apt-get install efityools
```

## Tool to build the usb keys : create-keys.sh

### Before launching

- define “mount\_point” variable (path to mount point) into “create-shell.sh” file
- ensure that path “mount\_point” is empty and available

### Build USB KEY 1

Plug a new usb key (attached on /dev/sdc in this case).

```
./create-keys.sh /dev/sdc -
```

Unplug the usb key.

### Build USB KEY 2

Plug a new usb key (attached on /dev/sdc in this case).

```
./create-keys.sh - /dev/sdc
```

Unplug the usb key.

## Boot on keys

- Plug one of keys, start the computer.
- For USB KEY 1 :
  - 1/ boot on usb key, start linux live
  - 2/ from root terminal, launch ChipSec with “chipsec\_main.py”.
- For USB KEY 2 :
  - 1/ boot on usb key and launch EFI binaries from EFI shell (Shell.efi is automatically started).
  - OR interrupt the normal boot to select a shell EFI from Boot Configuration and launch EFI binaries from EFI shell.
  - 2/ launching of binaries from EFI shell :
    - \* Before to launch the binaries, it is imperative to identify the usb key letter storing the binaries with commmands “fs0” or “fs1” or fsX ... then “dir”
    - \* After disabling of SecureBoot and enabling of Setup Mode (with BIOS options) : launch “KeyTool.efi” to import trust keys.
    - \* After re-enabling of SecureBoot and disabling of Setupe Mode (with BIOS options) : launch “HelloWorld.efi” (signed with imported Trust Keys) to check the good importation of trust keys.