CLIP OS: Building a defense-in-depth OS with the Linux kernel and open source software

Timothée Ravier, Nicolas Godinho, Thibaut Sautereau

Agence nationale de la sécurité des systèmes d'information (ANSSI)

Paris Open Source Summit 2018

December 6, 2018

About the ANSSI

- *Agence nationale de la sécurité des systèmes d'information*
- French authority in the area of cyberdefence, network and information security
- We are **not** an intelligence agency

Overview

CLIP OS?

- Linux distribution developed by the ANSSI
- Initially only available internally
- Now open source, mostly under the LGPL v2.1+
- Code and issue tracker hosted on GitHub[1][2]:
  - Version 4: available as reference and for upstream patch contribution
  - Version 5: currently developed version, alpha status

[1] https://github.com/CLIPOS
[2] https://github.com/CLIPOS-Archive

CLIP OS?

Not yet another Linux distribution

- Not a generic/multi-purpose distribution

CLIP OS?

Not yet another Linux distribution

- Not a generic/multi-purpose distribution

Targets three main use cases

- Office workstation
- Administration workstation
- IPsec gateway

Hardened OS

- Based on Gentoo Hardened

Hardened OS

- Based on Gentoo Hardened
- Hardened Linux kernel and confined services

Hardened OS

- Based on Gentoo Hardened
- Hardened Linux kernel and confined services
- No interactive *root* account available:
  $\Rightarrow$ "Unprivileged" admin, audit and update roles

Hardened OS

- Based on Gentoo Hardened
- Hardened Linux kernel and confined services
- No interactive *root* account available:
  $\Rightarrow$ "Unprivileged" admin, audit and update roles
- Automatic updates using A/B partition model (similar to Android 7+)

Hardened OS

- Based on Gentoo Hardened
- Hardened Linux kernel and confined services
- No interactive *root* account available:
  $\Rightarrow$ "Unprivileged" admin, audit and update roles
- Automatic updates using A/B partition model (similar to Android 7+)
- Multilevel security:

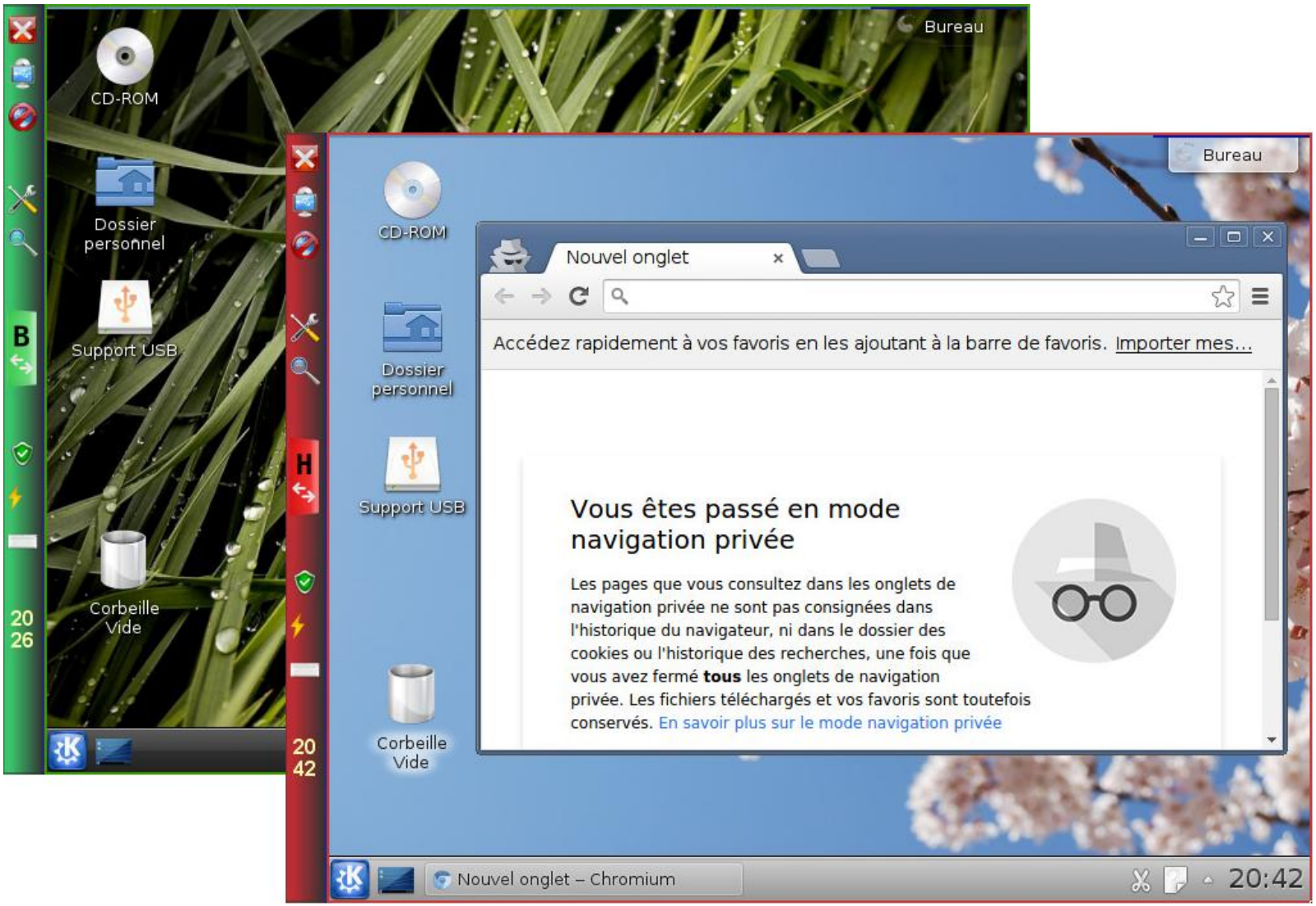
Hardened OS

- Based on Gentoo Hardened
- Hardened Linux kernel and confined services
- No interactive *root* account available:
  $\Rightarrow$ "Unprivileged" admin, audit and update roles
- Automatic updates using A/B partition model (similar to Android 7+)
- Multilevel security:
  - Provide two isolated user environments

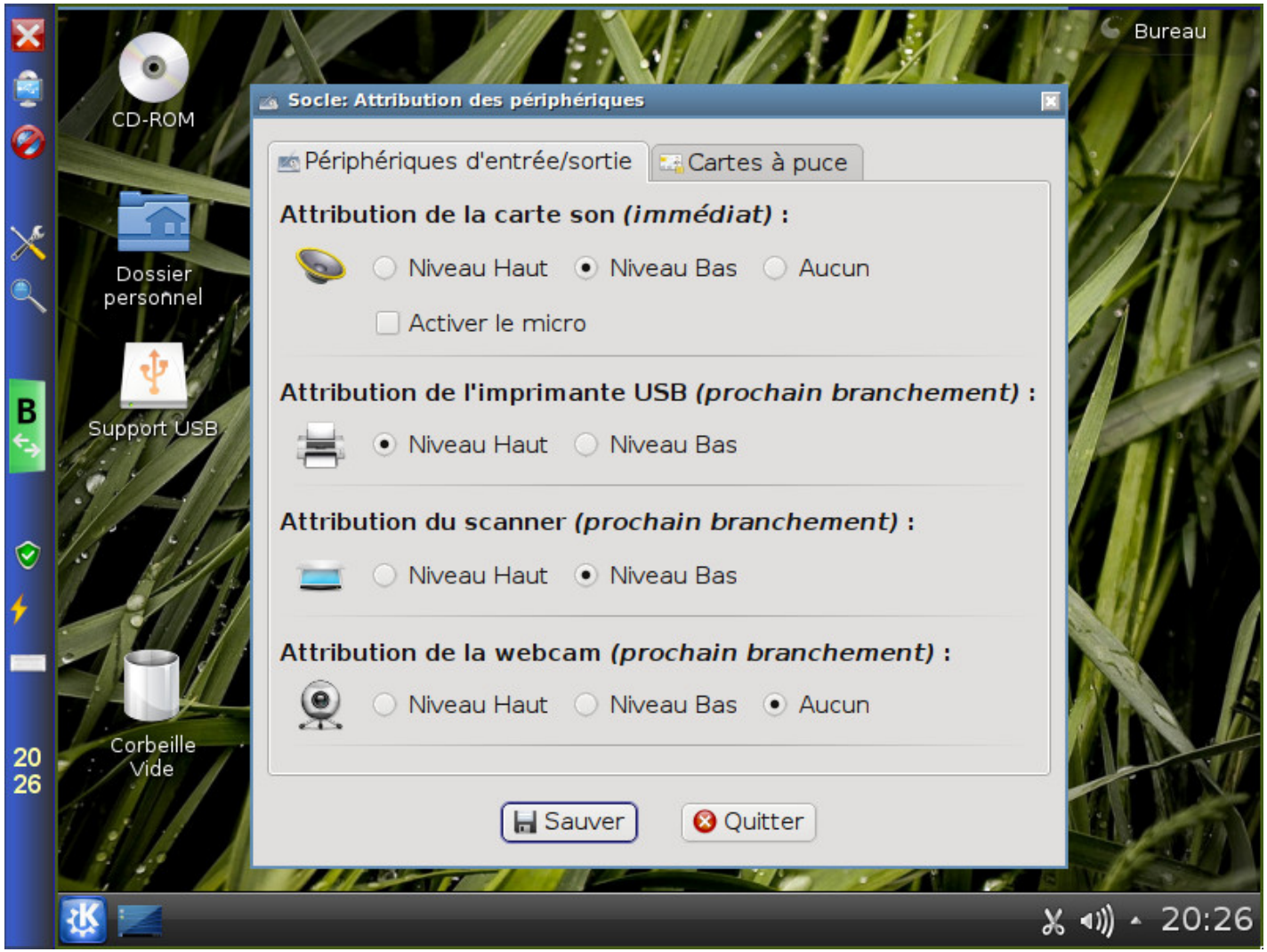
Hardened OS

- Based on Gentoo Hardened
- Hardened Linux kernel and confined services
- No interactive *root* account available:
  $\Rightarrow$ "Unprivileged" admin, audit and update roles
- Automatic updates using A/B partition model (similar to Android 7+)
- Multilevel security:
  - Provide two isolated user environments
  - Controlled interactions between isolated environments

Multilevel from the end user point of view



Admin panel: devices assignment per level



Differences with Qubes OS

CLIP OS development began 5 years earlier than Qubes OS

Differences with Qubes OS

CLIP OS development began 5 years earlier than Qubes OS

Main goals

- We target non-expert users
- Multilevel security model with two levels
- We favor a defense-in-depth approach

Differences with Qubes OS

CLIP OS development began 5 years earlier than Qubes OS

Main goals

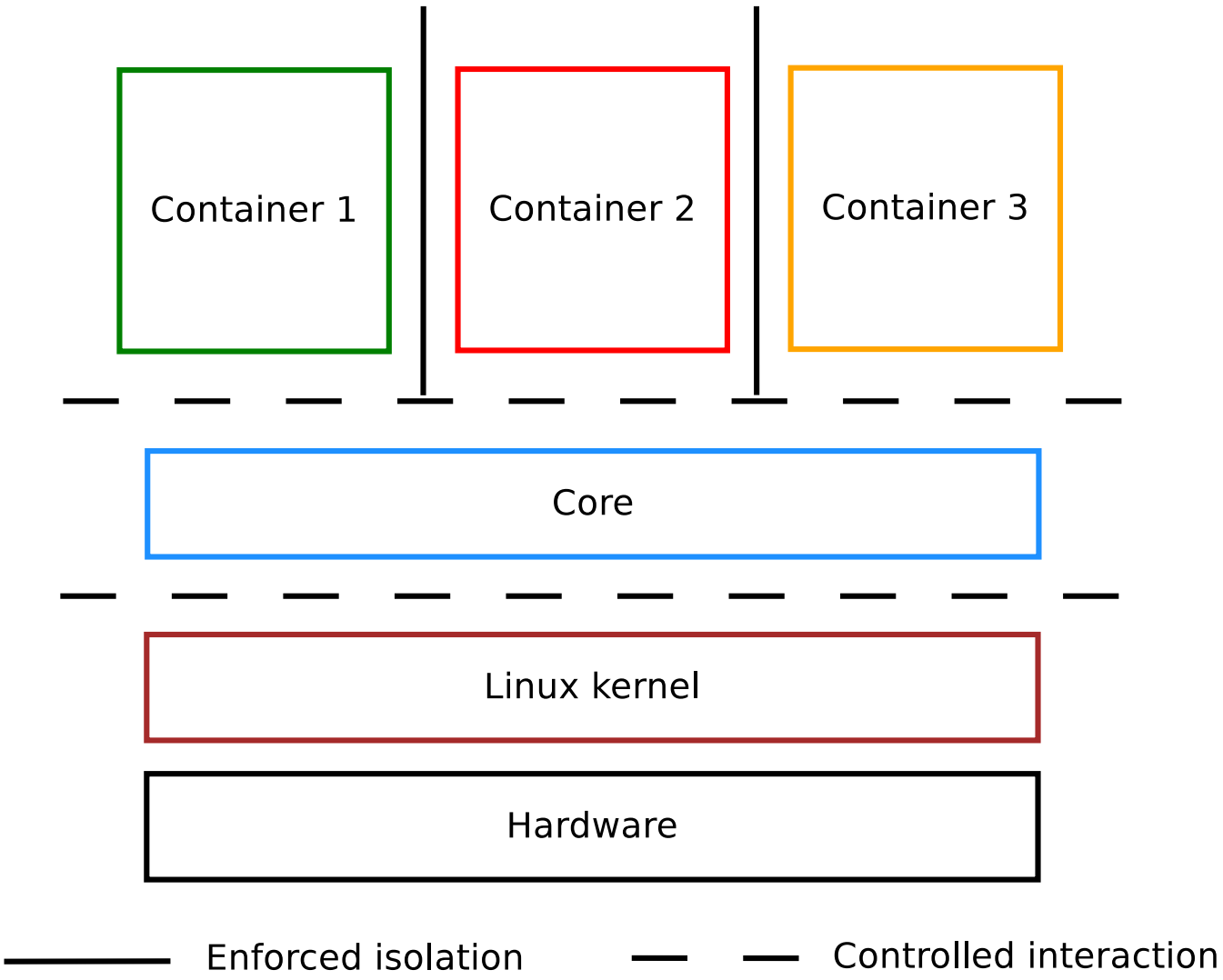
- We target non-expert users
- Multilevel security model with two levels
- We favor a defense-in-depth approach

Technical point of view

- Hypervisor (Qubes OS) vs. supervisor isolation (CLIP OS)
- CLIP OS: Limited access rights and capabilities, even for administrators

CLIP OS 5

Architecture



First public alpha release

- Functional core (boot to command line shell)

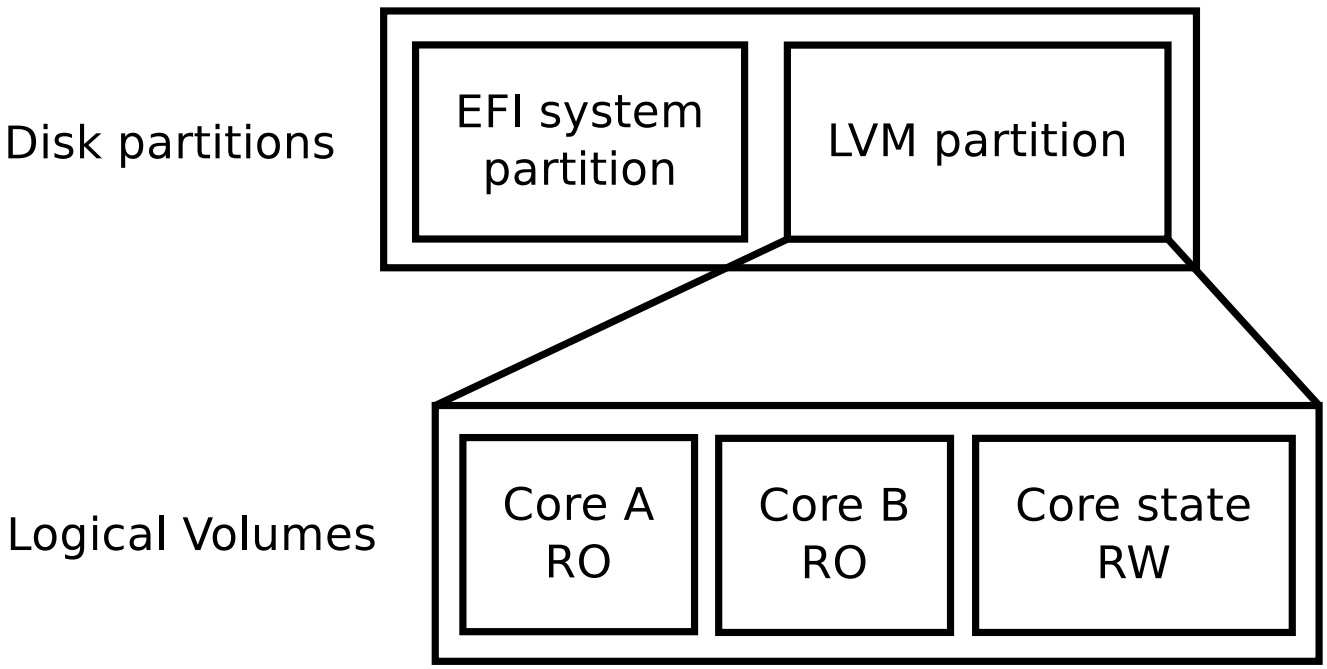
```
This is clipos-qemu.unknown_domain (Linux x86_64 4.18.18-r1-clipos) 13:45:15

Hint: Num Lock on

clipos-qemu login: root
Last login: Thu Nov 29 13:43:03 UTC 2018 on tty1
clipos-qemu / # lsblk
NAME                          MAJ:MIN RM   SIZE RO TYPE  MOUNTPOINT
vda                           254:0    0    20G  0 disk
|-vda1                        254:1    0   512M  0 part  /mnt/efiboot
`-vda2                        254:2    0  19.5G  0 part
  |-mainvg-core_5.0.0--alpha.1  252:0    0     4G  0 lvm
  | `-verity_core_5.0.0-alpha.1 252:3    0 158.5M  1 crypt /
  |-mainvg-core_state           252:1    0   512M  0 lvm
  | `-core_state_dif            252:4    0   474M  0 crypt
  |   `-core_state              252:5    0   474M  0 crypt /mnt/state
  `-mainvg-core_swap            252:2    0     1G  0 lvm
    `-swap                      252:6    0     1G  0 crypt [SWAP]
clipos-qemu / # uname -sr
Linux 4.18.18-r1-clipos
clipos-qemu / #
```

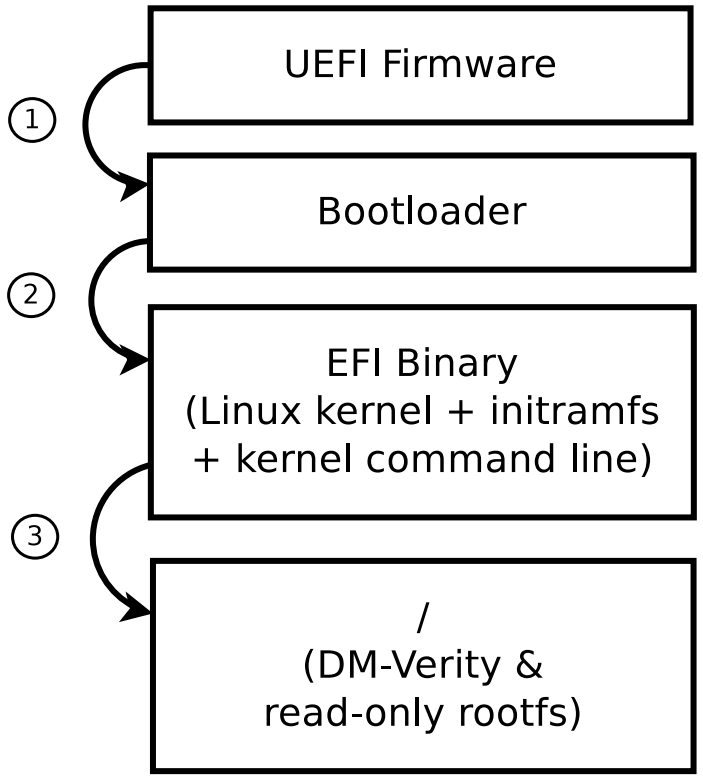
First public alpha release

- Strict split between:
  - Read Only: system executables, configuration and data
  - Read Write: runtime configuration, logs, user and application data



First public alpha release

- Initial boot chain integrity:
  - Secure Boot (bootloader, initramfs, Linux kernel and its command line)
  - Read-only system partition protected by DM-Verity



First public alpha release

- Initial hardware support: QEMU/KVM virtual machine

Features added or in progress since alpha

Added:

- Read-write system data stored in a DM-Crypt+Integrity volume
- Initial hardware profiles support

Features added or in progress since alpha

Added:

- Read-write system data stored in a DM-Crypt+Integrity volume
- Initial hardware profiles support

In progress:

- TPM support for unattended LUKS secret unsealing (system RW data)
- Public infrastructure setup:
  - Code review (Gerrit)
  - Buildbot (daily and on-demand builds)

Roadmap: Beta

- Client for automatic updates
- Confined IPsec client and SSH server
- Basic network (DHCP, static IP) and firewall (static rules) support
- "Unprivileged" admin, audit and update roles
- Initial physical hardware support

Roadmap: 5.0 stable

- Confined user environments (GUI)
- Multilevel support (Vserver-like LSM)
- Automated installation using PXE
- etc.

Working on CLIP OS?

See full documentation at `https://docs.clip-os.org`:

- Install dependencies
- Retrieve sources
- Automated build steps
- Test with QEMU

Working on CLIP OS?

See full documentation at `https://docs.clip-os.org`:

- Install dependencies
- Retrieve sources
- Automated build steps
- Test with QEMU

Full project build time estimates:

- From scratch: about 2-3 hours
- Incremental: about 5-10 minutes (and more depending on compilations)

Working on CLIP OS?

See full documentation at `https://docs.clip-os.org`:

- Install dependencies
- Retrieve sources
- Automated build steps
- Test with QEMU

Full project build time estimates:

- From scratch: about 2-3 hours
- Incremental: about 5-10 minutes (and more depending on compilations)

Join us at the Workshop this afternoon at 15h, room Projection, 2nd floor

Conclusion

Open source project:

- Sources: https://github.com/CLIPOS
- Bugs: https://github.com/CLIPOS/bugs
- Contribute with GitHub pull-requests

Code review (Gerrit) and Buildbot infrastructure setup in progress.

Planned contributions:

- Linux kernel (https://github.com/clipos/src_external_linux)
- CLIP OS version 4 patches will be submitted upstream
- Gentoo ebuilds, etc.

Thanks!

✉ clipos@ssi.gouv.fr

- Website: clip-os.org
- Docs: docs.clip-os.org
- Sources: github.com/CLIPOS
- Bugs: github.com/CLIPOS/bugs

We're hiring! (but not directly for CLIP OS)

Linux system security expert

https://www.ssi.gouv.fr/emploi/expert-en-securite-des-systemes-linux/