Journal : Sécuriser le boot dans un datacenter hostile : Heads, Trenchboot, ME... que choisir en 2025 ?

25 nov. 2025
12

Posté par **gnuromain** le 25 novembre 2025 à 13:17. **Licence CC By-SA.**
Étiquettes : **coreboot**, **heads**, **sécurité**

Cher Journal,

Je dois déployer des serveurs… mais dans des environnements où je n'ai pas totalement confiance. Pas au point d'imaginer un ninja intrus venir reflasher mes machines la nuit, mais suffisamment pour vouloir une chaîne de boot propre, vérifiable et contrôlée de bout en bout.

NB : Si vous avez du matériel à vendre qui respecte mon cahier des charges, je suis preneur :)
Je regarde en ce moment du côté des serveurs Purism, Nitrokey, etc.

J'ai évidemment pensé au combo classique : coreboot + Heads + clé Nitrokey.
C'est propre, c'est bien intégré, et pour un laptop c'est presque parfait.
Mais sur un serveur distant, il faut valider physiquement chaque boot, et là… ça devient franchement relou.
Je ne peux pas camper dans chaque datacenter juste pour appuyer sur Entrée.

Alors j'ai regardé la solution ✨ moderne ✨ :
③ Trenchboot + TPM2 + Keylime (remote attestation).
On gagne énormément : DRTM, attestation distante, automation, contrôle centralisé.
Mais… pour faire tout ça, on repose sur une base x86 moderne, et donc :
il faut faire confiance à Intel ME (ou AMD PSP).
Et ça, ça pue …

Je sais qu'il y a des moules qui bossent sur des plateformes open hardware, voire totalement libérées du ME/PSP (OpenPOWER, RISC-V…), mais ce n'est pas encore très pratique pour faire tourner tout ce dont j'ai besoin côté services.

Du coup je réfléchis à un compromis réaliste :
une chaîne de boot diskless, mesurable, et entièrement sous mon contrôle.

Un schéma du genre :

Hardware
↓
coreboot
↓
Heads (sécurité : TPM + GPG + clé physique)
↓
iPXE (avec mon propre CA intégré, pour n'accepter QUE mon serveur)
↓
Boot réseau / OS éphémère

L'idée :

garder l'intégrité du boot grâce à Heads,

éliminer tout stockage local sensible,

contrôler complètement la chaîne TLS entre iPXE et mon infra (via mon propre CA),

sans devoir aller physiquement valider chaque démarrage dans le datacenter.

Bref, je cherche à savoir si des moules ont déjà tenté un setup hybride du genre :

Heads pour garantir l'intégrité locale,

iPXE sécurisé avec un CA maison,

boot réseau minimaliste,

le tout sans devoir faire confiance aveuglément au ME.

Si vous avez déjà déployé ce genre d'architecture, ou trouvé un compromis acceptable entre Heads, boot réseau sécurisé et gestion à distance, vos retours d'expérience m'intéressent.

Merci les moules :)

(3 commentaires). Markdown EPUB

Threat model

Posté par **pasBill pasGates** le 25 novembre 2025 à 16:39.
Évalué à 7 (+6/-0).

> Tu peux clarifier quelles sont les menaces qui t'amènent à chercher ces solutions ?
>
> Si tu ne fais pas confiance aux gens qui bossent là-bas, tu penses vraiment que tu pourras te protèger ce des gens avec juste du soft sur du HW standard ?

Répondre

[^] # Re: Threat model

Posté par **gnuromain** le 25 novembre 2025 à 20:01.
Évalué à 1 (+2/-1).

> Merci pour ta question, elle est légitime : avant de parler d'outils, il faut clarifier le threat model.
>
> Dans mon cas, je ne cherche pas à me protéger d'un datacenter entièrement compromis façon "attaquant omnipotent". Je pars du principe que :
>
> L'infrastructure peut être honnête-mais-curieuse (accès physique limité mais possible, techniciens qui peuvent voir/booter la machine).
>
> Je veux réduire les attaques opportunistes ou non ciblées, pas me défendre contre un adversaire disposant d'un budget NSA.
>
> Le matériel est standard, donc pas de racine de confiance matérielle parfaite, mais je veux au moins détecter les modifications de firmware/boot.
>
> Je sais bien qu'on ne peut pas atteindre la sécurité absolue sur du hardware que je ne maîtrise pas. En revanche, ce que permettent Heads, TrenchBoot et les mécanismes de mesure d'intégrité, c'est :
>
> Augmenter le coût de l'attaque : un technicien ne pourra plus simplement booter une clé USB ou flasher un firmware sans que ce soit détecté.
>
> Détecter l'altération de la chaîne de démarrage via des mesures dans le TPM — ce qui est déjà un gain considérable.
>
> Valider l'intégrité du système avant de déployer des secrets (clé de déchiffrement, volumes, etc.).
>
> Limiter l'impact d'un accès physique court (evil-maid, reflash simple, boot sur matériel modifié).
>
> Et justement :
> c'est parce que ces menaces existent mais ne relèvent pas d'un attaquant tout-puissant que de tels outils sont développés et utilisés — pour rendre

l'attaque plus difficile, plus coûteuse et surtout détectable.

En résumé :
- Je ne cherche pas une sécurité absolue, mais un niveau de confiance raisonnable dans un environnement que je ne contrôle pas entièrement.
- Les outils du type Heads/TrenchBoot servent précisément à couvrir ce type de scénario.

Répondre

stboot

Posté par **gnuromain** le 25 novembre 2025 à 20:30.
Évalué à 0 (+1/-1).

Chers moules,

En fait, j'ai besoin de ce projet open source qui fait exactement ce que je veux :

https://git.glasklar.is/system-transparency/core/stboot
https://www.system-transparency.org/

Il ne me reste plus qu'à trouver des serveurs compatibles coreboot :) Une idée de bons revendeurs ? Du bon matos de chez pépé ? :)

Je ne me suis encore jamais amusé avec coreboot, donc tout retour d'expérience ou conseil est le bienvenu.

Répondre

Envoyer un commentaire

Suivre le flux des commentaires

Note : les commentaires appartiennent à celles et ceux qui les ont postés. Nous n'en sommes pas responsables.

Revenir en haut de page

Derniers commentaires

- Re: Stupéfiant...
- stboot
- Re: A noter que le s…
- Re: Threat model
- Re: La correction gr…
- La machine a bon dos
- Re: A noter que le s…
- Service libre ?
- Re: Et les appels d'u…
- Re: individu vs role
- A noter que le serve…
- La correction gram…

Étiquettes (tags) populaires

- intelligence_artificielle
- merdification
- grands_modèles_de…
- entretien
- administration_fran…
- souveraineté_numer…
- chatgpt
- jeu_vidéo
- tour_des_gull
- bulle
- bigtech
- adieu_windows

Sites amis

- April
- Agenda du Libre
- Framasoft
- Éditions D-BookeR
- Éditions Eyrolles
- Éditions Diamond
- Éditions ENI
- La Quadrature du Net
- Lea-Linux
- En Vente Libre
- Grafik Plus
- Open Source Initiative

À propos de LinuxFr.org

- Mentions légales
- Faire un don
- L'équipe de LinuxFr.…
- Informations sur le s…
- Aide / Foire aux que…
- Suivi des suggestion…
- Règles de modération
- Statistiques
- API pour le dévelop…
- Code source du site
- Plan du site