SSH Remot Unlock (LUKS)

SSH REMOTE UNLOCK (LUKS)

Outil Cible : *Dropbear (Initramfs)*

Lien Parent : *[MANUAL - SSH] (Gestion des clés)*

Cas d'usage : *Déverrouiller une partition chiffrée (LUKS) sur un serveur sans clavier/écran (Headless).*

Le Concept (Théorie Rapide)

Au démarrage, un serveur chiffré charge un mini-système en RAM (**Initramfs**) et attend la passphrase. L'OS complet (Debian) n'est pas encore chargé.

La solution : On injecte un serveur SSH léger (**Dropbear**) dans cet Initramfs pour permettre une connexion temporaire.

Le Piège : L'Initramfs et l'OS sont deux entités distinctes. Ils peuvent avoir des IPs différentes et des clés SSH différentes.

Installation (Deploy)

Sur le serveur cible (VM ou Bare Metal) :

1. Installer le paquet magique

```
sudo apt update && sudo apt install dropbear-initramfs
```

```
Installing:
  dropbear-initramfs

Installing dependencies:
  dropbear-bin  libtomcrypt1  libtommath1

Summary:
  Upgrading: 0, Installing: 4, Removing: 0, Not Upgrading: 0
  Download size: 466 kB / 702 kB
  Space needed: 1,952 kB / 22.4 GB available

Continue? [Y/n]
```

2. (Optionnel) Installer cryptsetup si ce n'est pas fait

```
sudo apt install cryptsetup
```

Configuration "Dr. Jekyll & Mr. Hyde" (Réseau)

Pour éviter que le serveur change d'IP entre le déverrouillage (Dropbear) et l'utilisation (OS), on force une **IP Statique** partout.

Figer l'OS (Debian)

```
sudo nano /etc/network/interfaces
```

```
# Exemple pour une VM sur 192.168.1.50
allow-hotplug enp1s0
iface enp1s0 inet static
    address 192.168.1.50/24
    gateway 192.168.1.1
    # DNS (Optionnel)
    dns-nameservers 9.9.9.9 192.168.1.1
```

Figer le Boot (Initramfs)

Ajouter ou modifier la variable IP à la fin du fichier.

```
sudo nano /etc/initramfs-tools/initramfs.conf
```

```
IP=192.168.1.50::192.168.1.1:255.255.255.0:vm-server:enp1s0:off
```

 # Syntaxe : IP=[Client]::[Gateway]:[Netmask]:[Hostname]:[Interface]:[Autoconf]

Configuration Essentielle

Les Clés (Le Piège)

Dropbear ne lit **PAS** le fichier /root/.ssh/authorized_keys de l'OS. Il a son propre trousseau.

Copier votre clé publique (PC Admin) dans le trousseau Dropbear

Attention : Le fichier doit être créé s'il n'existe pas.

```
sudo nano /etc/dropbear/initramfs/authorized_keys
```

Collez votre clé publique (ex: ssh-ed25519 AAAA...) sur une seule ligne.

L'Application (Build)

Toute modification ci-dessus est inutile sans cette commande :

```
sudo update-initramfs -u
```

Procédure de Connexion (Client Side)

C'est ici que ça diffère du SSH classique.

Le Problème : Dropbear génère une clé d'hôte (Host Key) différente de celle d'OpenSSH (l'OS final). Votre client SSH va hurler au "Man-in-the-Middle".

La Commande "Bélier" : On utilise une commande qui ignore volontairement la vérification de l'hôte connu pour cette session spécifique.

Syntaxe :

ssh -i [Clé_Privée] -o "UserKnownHostsFile=/dev/null" root@[IP_Serveur]

Exemple :

```
ssh -i ~/.ssh/key-server -o "UserKnownHostsFile=/dev/null" root@192.168.1.50
```

```
The authenticity of host '192.168.1.50 (192.168.1.50)' can't be established.
ED25519 key fingerprint is SHA256:P1w05TTmleoBVv/gA3HYLNnyW6Y9n8xfAJcTEuZEkr0.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '192.168.1.50' (ED25519) to the list of known hosts.
Enter passphrase for key '/home/main/.ssh/vm.rootstock':
To unlock root partition, and maybe others like swap, run `cryptroot-unlock`.


BusyBox v1.37.0 (Debian 1:1.37.0-6+b3) built-in shell (ash)
Enter 'help' for a list of built-in commands.
```

Une fois connecté (Prompt ~ #)

```
~ # cryptroot-unlock
Please unlock disk vda5_crypt:      <--  ENTER PASSPHRASE
cryptsetup: vda5_crypt set up successfully
~ # Connection to 192.168.1.50 closed by remote host.
Connection to 192.168.1.50 closed.
```

La session se coupe automatiquement (Connection closed). C'est normal : Dropbear se suicide pour laisser la place à l'OS réel.

> **Note :** "Ne confondez pas la clé de la porte d'entrée (OpenSSH) avec le pied-de-biche pour ouvrir la fenêtre (Dropbear). Ce sont deux entités distinctes. Si vous perdez l'accès à Dropbear sur un serveur distant chiffré, préparez la voiture, vous allez faire un tour au datacenter."

L'Alias "Double Tap" (Client Side)

Dropbear gère mal les terminaux. Pour que la commande cryptroot-unlock fonctionne via un alias, il faut forcer une double allocation TTY (-tt).

Ajoutez ceci à votre ~/.bash_aliases sur votre machine d'administration :

```
nano ~/.bash_aliases
```

Alias de déverrouillage (Adaptez l'IP et la clé -i)

```
alias unlock-vm-test='ssh -tt -i ~/.ssh/vm.server -o "UserKnownHostsFile=/dev/null" -o
"StrictHostKeyChecking=no" root@192.168.1.50 "cryptroot-unlock"'
```

-tt Force l'allocation d'un pseudo-tty (Permet de voir le prompt du mot de passe).

-o "UserKnownHostsFile=/dev/null" Ignore les erreurs de "Host Key" (car la clé de Dropbear est différente de celle de l'OS).

cryptroot-unlock La commande interne à l'initramfs pour ouvrir le volume LUKS.

```
source ~/.bashrc
```