Tutoriels Ordinateurs

SecureBlue avec des applis compartimentées

Difficulté ★★★☆☆ Temps requis 1h

Objectif

Compartimenter des applications dans des VM sur l'OS SecureBlue

Étapes

keyboard_arrow_right Installer le gestionnaire

keyboard_arrow_right Paramétrer ses compartiments

Prérequis

task Ordi secureblue

Matériel requis

sell Ordinateur sous système d'exploitation Secureblue

sell Connexion internet

Intérêts de la mesure

thumb_up Séparer de manière forte ses usages numériques

thumb_up Configuration simple pour chiffrer un disque, passer par Tor/VPN etc.

Inconvénients

thumb_down Demande à être à l'aise avec un terminal

thumb_down Outil très récent et peu testé encore

— Introduction et parti pris

SecureBlue est un très bon compromis comme système d'exploitation simple à prendre en main, robuste face à des attaques physiques et distantes, fonctionnel avec de nombreuses applications et fiable à l'usage.

Cependant, son système de virtualisation intégré est compliqué à prendre en main et ne permet pas de compartimenter fortement les usages numériques aussi facilement que sur QubesOS, un système bien plus poussé sur la compartimentation mais qui est très peu accessible pour un public non technique.

Cet outil s'intègre dans SecureBlue pour faciliter pour chaque besoin la création d'une VM dédiée à l'exécution d'une unique application isolée, avec de nombreux paramétrages de sécurité et en virtualisant complètement une autre machine peu lourde.

— Présentation de l'outil

Pour garantir une séparation forte avec le reste du système, cet outil fait tourner chaque application dans une **machine virtuelle**.

> **Intégration visuelle**
>
> Les applications compartimentées s'affichent de la même manière que n'importe quelle autre application, tu verras un petit badge bleu en bas à droite de l'icône officielle pour te rappeler qu'elle est compartimentée.
>
> Le démarrage prend nécessairement plus de temps qu'une application de base, mais tu ne verras pas de différence à l'usage.
>
> *Lors des tests,* `Firefox` *s'ouvre en 30s sur un ordinateur assez vieux avec son espace disque chiffré et à travers Tor.*

Cette pratique est pertinente lorsque tu as un modèle de menace fort qui demande à compartimenter tes usages, et que tu as plusieurs identités numériques ou certains usages spécifiques comme par exemple :

- consulter/écrire les mails d'une adresse publique/exposée
- plusieurs comptes Signal pour des identités numériques distinctes
- navigation sur Internet dédiée à un objectif sensible/exposé, comme gérer des réseaux sociaux

Cet outil reprend des concepts de QubesOS. Il installe l'application souhaitée dans une `base` *(équivalent d'un* ***TemplateVM****)*, ce qui crée un disque système minimal qui est partagé en lecture seule par plusieurs `compartiments` *(équivalent d'une* ***AppVM****)*.

Tu peux créer autant de compartiments que tu veux, et ils sont chacun configurables selon tes besoins :

- est-ce que tu as besoin d'une connexion Internet ? Qui garantit ton anonymat (Tor), avec un noeud de sortie Mullvad spécifique ou avec la même connexion que le reste des applications ?
- est-ce que le stockage des données doit en plus être chiffré ?
- est-ce que les données y sont persistantes ou jetables *(* ***DisposableVM****)*
- est-ce que tu dois partager un dossier avec SecureBlue pour faciliter le transfert de fichiers, lequel ?
- combien de RAM et d'espace disque ce compartiment peut-il utiliser ?

> ⚠ **Presse-papiers partagé**
>
> Ce que tu copies d'un compartiment sera disponible sur l'hôte SecureBlue et inversement. Ça ne peut pas être désactivé.

Trois bases sont fournies avec l'outil, avec pour chacune un compartiment par défaut :

Nom de la base	Application installée	Compartiment associé par défaut
`mail`	Thunderbird	`mail`
`browser`	Firefox	`social-network`
`signal`	Signal Desktop *(via Flatpak)*	`signal`

— Installation

Prérequis

Installe SecureBlue et suis le tuto Ordi secureblue avant de poursuivre.

Assure-toi d'avoir tous les outils nécessaires avec cette commande dans ton terminal (`^ Ctrl` + `⎇ Alt` + `T`) :

Bash

```
# Indispensables
rpm-ostree install libvirt qemu-kvm virtiofsd passt waypipe cloud-utils genisoimage
```

Installation de l'outil

Dans un terminal, pour installer cet utilitaire dans `$HOME/bin/secureblue-vms`, exécute les commandes suivantes :

Bash

```
# Change ce chemin par là où tu souhaites télécharger l'outil
DIR_TOOL="$HOME/bin/secureblue-vms"

# Téléchargement de l'outil
mkdir -p "$(dirname "$DIR_TOOL")"
git clone https://0xacab.org/TheCodesUprising/tools/secureblue-vms "$DIR_TOOL"
cd "$DIR_TOOL"

# Si tu veux Tor / Mullvad
./src/secureblue-vms --install-tools
# Installe des applications optionnelles
./src/secureblue-vms --install-tools=all

# Vérifie que tu n'as rien d'important ouvert avant de redémarrer
systemctl reboot
```

Après le redémarrage si tu veux passer par Tor :

Bash

```
systemctl enable --now tor
```

Tu peux vérifier que ton appareil est compatible et que tous les outils sont bien installés avec :

Bash

```
./src/secureblue-vms --host-check
```

Si une entrée apparaît en rouge, tu verras dans la dernière colonne la commande à exécuter pour installer le bon logiciel.

Pour avoir un aperçu de toutes les commandes disponibles :

Bash

```
./src/secureblue-vms --help

Usage: ./src/secureblue-vms [OPTION...] [COMMAND]
      OPTIONS SUMMARY
              -h | --help                     Print help information
              -v | --verbose                  Increase verbosity
              -q | --quiet                    Suppress non-error messages
      COMMANDS SUMMARY
              -l | --list                     List bases and compartments
and their state
              -s | --status                   Show which compartments are
running
              -S | --stop <compartment>       Stop a running compartment
              -c | --host-check               Report what this host can
and cannot do
              -g | --guest-build              Download and verify the
Alpine base image
              -b | --build <base>             Provision a reusable base
rootfs
              -L | --launch <compartment>     Boot a compartment and open
its app
              -r | --remove-base <base>       Delete an unused base
rootfs
                   --auto-clean               Delete every unreferenced
base
                   --install-launchers        Add a menu .desktop entry
per compartment
                   --remove-launchers         Remove those menu entries
                   --install-tools[=all]      Layer host deps: tor, socat
(=all adds the 3 optional ones)
```

— Mise en place et paramétrage de l'outil

Créer les `bases` des applications

> **Compilation automatique au premier lancement**
>
> Cette étape est optionnelle, au lancement de ton application la base sera automatiquement compilée si elle n'existe pas.
>
> Ça te permet cependant de tout préparer en avance pour économiser 15 à 45 minutes d'attente au premier lancement sans que tu saches où ça en est.

C'est l'étape la plus longue : l'outil va télécharger une version de Linux minimale puis va télécharger chaque application pour créer une image template partagée entre tes différents

compartiments.

Chaque ligne peut prendre entre 5 et 15 minutes, en fonction de ta connexion Internet. Il n'y en a besoin que d'une seule fois ; si tu n'as pas besoin d'un outil, épargne-toi la ligne.

Bash

```
# Firefox
./src/secureblue-vms -vv --build browser
# Thunderbird
./src/secureblue-vms -vv --build mail
# Signal Desktop (installé avec Flatpak)
./src/secureblue-vms -vv --build signal
```

Tu peux créer de nouvelles `bases` avec les fichiers `$HOME/.config/secureblue-vms/bases` grâce aux paramètres suivants :

Clef	Description
`packages`	Liste des paquets `apk`, séparés par des espaces
`flatpak`	Flathub app id : https://flathub.org/
`app_args`	Arguments supplémentaires donnés à l'application au lancement
`app_env`	Variables d'environnement supplémentaires données à l'application au lancement

Premiers lancements des `compartiments`

Une fois que tu as défini tous tes compartiments, tu peux créer des `.desktop` pour pouvoir les lancer facilement avec leur icône depuis ton lanceur d'applications :

Bash

```
./src/secureblue-vms --install-launchers
```

Pour facilement distinguer les applications compartimentées, tu verras un petit badge bleu en bas à droite des icônes officielles.

Tu peux ensuite ouvrir ton application comme tu lancerais n'importe quel autre programme.

> **Information**
>
> Le premier lancement sera plus lent que les suivants, ça ne sert à rien d'appuyer plusieurs fois sur l'icône 🙂

Si tu as décidé de chiffrer le disque (configuration par défaut), tu verras une fenêtre s'ouvrir deux fois pour te demander de taper le mot de passe de chiffrement, de préférence unique par compartiment. Chaque nouveau lancement te demandera son mot de passe.

Si tu as besoin d'avoir plus d'informations sur ce qu'il se passe :

Bash

```
# Savoir quels compartiments sont en train de tourner
./src/secureblue-vms --status

# Remplace 'mail' par le nom du compartiment que tu souhaites démarrer
./src/secureblue-vms -vv --launch mail

# Arrête le compartiment 'mail' proprement
./src/secureblue-vms --stop mail
```

Configurer les `compartiments` selon tes besoins

Plusieurs propositions de compartiments ont été générées dans `$HOME/.config/secureblue-vms/compartments`, tu peux y créer/supprimer/éditer les fichiers de configuration selon tes besoins.

Par exemple, tu peux vouloir plusieurs comptes Signal, augmenter l'allocation de la RAM pour un usage de Firefox en particulier ou consulter tes mails à travers Tor.

Le nom d'un fichier correspond au nom de ton compartiment (`social-network.conf` créera le compartiment `social-network` par exemple), et tu peux changer les paramètres suivants :

Clef	Défaut	Description
`base`	***obligatoire***	Base utilisée : `browser` / `mail` / `signal`
`app`	***obligatoire***	Commande lancée dans chaque compartiment au démarrage
`ram_mb`	`2048`	Quantité de RAM assignée, en Mo

Clef	Défaut	Description
`data_size`	`8G`	Taille du disque persistant monté sur `/home`, comprend les suffixes `M` / `G`. Plafond, l'espace utilisé se remplit au fur et à mesure.
`persist`	`yes`	`yes` pour garder le contenu de `/home` entre plusieurs démarrages, `no` pour être temporaire
`encrypt_data`	`no` (configuration des compartiments par défaut sur `yes`)	`yes` chiffre le contenu de `/home` avec LUKS, mot de passe demandé à chaque lancement
`network`	`nat`	`nat` / `none` / `tor` / `socks`
`shared_folder_path`	-	Partage d'un dossier avec SecureBlue
`icon`	-	Change l'icône de la fenêtre

> ⚠ **N'oublie pas ton mot de passe de déchiffrement !**
>
> Si tu as `encrypt_data=yes`, à chaque démarrage tu devras entrer ton mot de passe. **Si tu l'oublies, les données seront définitivement perdues** sans moyen de les récupérer.

Paramètre `network`

4 modes sont disponibles :

Mode	Isolé ?	Fonctionnement
`nat`	**Non**	Le trafic sortant utilise l'IP de ton ordinateur directement
`none`	Oui	Aucun accès Internet
`tor`	Oui	Tout le trafic (dont DNS) passe par Tor, avec un circuit différent par compartiment

Mode	Isolé ?	Fonctionnement
`socks[=server:port]`	Oui	Fait passer le trafic par un tunnel `SOCKS`, par Mullvad par défaut

En-dehors du mode par défaut `nat`, le paramétrage réseau est fait depuis SecureBlue, un compartiment n'a pas d'interface IP : cette isolation réseau garantit que même si un compartiment est compromis, l'attaquant ne peut pas récupérer ton adresse IP ni aucune information en-dehors du compartiment.

Si tu as besoin de plus que deux compartiments en même temps pour passer par Tor, il te faut augmenter la valeur de `SBVMS_TOR_POOL_N` et ouvrir de nouveaux ports :

Bash

```
# Change 'NN' par le nombre de compartiments dont tu as besoin + 9050
# Par exemple pour 15 compartiments : 9052-9065
run0 semanage port -a -t tor_port_t -p tcp 9052-90NN
```

Si tu veux utiliser le nœud de sortie Mullvad par Paris, tu peux utiliser l'option : `network=socks=fr-par-wg-socks5-001.relays.mullvad.net:1080` (tu peux trouver la liste ici : https://mullvad.net/en/servers?type=wireguard&status=true)

Paramètre `shared_folder`

Si tu as besoin, tu peux partager **un** dossier entre SecureBlue et le compartiment pour faciliter le transfert de fichiers.

> **Partage uniquement un dossier dédié**
>
> Plutôt que de monter un dossier très gros qui contient trop d'informations *(refus automatique de `$HOME`, `/`, dossiers cachés)*, restreins la taille du dossier aux seules informations nécessaires au compartiment.
>
> Si Thunderbird se fait compromettre par exemple, un attaquant n'aurait accès qu'aux quelques fichiers contenus dans ce dossier.

Le dossier pointé par le paramètre `shared_folder_path=<dir>` est monté dans le compartiment sur `/mnt/shared` et est aussi accessible sur `$HOME/Shared` pour un accès plus simple.

Le presse-papier est partagé entre SecureBlue et tous les compartiments.

Mises à jour

Lorsque des mises à jour existent, il te suffit de :

Bash

```bash
cd "$HOME/bin/secureblue-vms"
git pull
```

Lorsqu'une base peut être mise à jour elle le sera à la prochaine exécution.

Les configurations de tes compartiments ne seront jamais écrasées.

Un compartiment supprimé n'est pas récupérable.

Désinstallation

Bash

```bash
# Supprime les compartiments dont tu n'as plus besoin
rm ~/.config/secureblue-vms/compartments

cd "$HOME/bin/secureblue-vms"
./src/secureblue-vms --auto-clean
./src/secureblue-vms --remove-launchers

# Suppression des bases
./src/secureblue-vms --remove-base browser
./src/secureblue-vms --remove-base mail
./src/secureblue-vms --remove-base signal
```

Si tu veux supprimer les disques qui contiennent tes données. **Attention, c'est irréversible** !

Bash

```bash
ls ~/.local/state/secureblue-vms/
rm ~/.local/state/secureblue-vms/TON_COMPARTIMENT
# Suppression du cache
rm -rf ~/.cache/secureblue-vms/
# Suppression de cet outil
rm -rf "$HOME/bin/secureblue-vms"
```

— Troubleshooting

Mon application n'apparaît pas dans le menu ou son icône n'est pas bonne

Bash

```bash
cd "$HOME/bin/secureblue-vms"
./src/secureblue-vms --install-launchers
```

Si tu as une application différente de celles qui sont affichées par :

Bash

```
ls "$HOME/bin/secureblue-vms/share/icons"/badged-*.png
```

alors il te faut installer l'outil qui génère les icônes :

Bash

```
cd "$HOME/bin/secureblue-vms"
./src/secureblue-vms --build ta-base
./src/secureblue-vms --install-tools=all
# Redémarre pour rendre disponible
./src/secureblue-vms --install-launchers
```

`--host-check` a encore des entrées rouges après l'installation

Redémarre ton ordinateur pour rendre `--install-tools` effectif.

Un compartiment avec `tor` n'a pas de réseau

Bash

```
systemctl enable --now tor
```

Pour t'assurer que le réseau passe à travers Tor, rends-toi sur https://check.torproject.org depuis le Firefox compartimenté avec `network=tor`.

— Plus d'informations

Cet outil est encore en phase de tests, n'hésite pas à l'utiliser pour en faire des retours ! Notamment est-ce que tu as détecté des bugs, est-ce que tu as des besoins qui ne sont pas couverts par l'outil ou est-ce que tu vois des améliorations à y faire ? → barricade-numerique@riseup.net

Pour plus de détails techniques, tu peux aller lire le README.md du projet :
https://0xacab.org/TheCodesUprising/tools/secureblue-vms

2026-07-302026-08-08