Analyse du comportement du Malware Linux "perfctl" avec Kunai et Nushell

Jean-Marc Pouchoulon

Novembre 2025

L'objectif de ce TP est d'analyser le comportement d'un "malware" Linux dans un environnement isolé en maximisant le niveau de sécurité. Cet environnement est une "sandbox Kunai" lancée dans une machine virtuelle Debian 13 supportant la virtualisation imbriquée.

Kunai est un outil de monitoring de la sécurité pour Linux. Il utilise la technologie eBPF (Extended Berkeley Packet Filter) permettant une interaction sécurisée avec le kernel Linux. Cette technologie performante permet de capturer les appels systèmes (sycalls) réalisés par les processus en cours d'exécution sur le système hôte Linux.

Kunai permet ainsi de surveiller et d'analyser le comportement des processus Linux en temps réel sous un angle orienté cyber-sécurité.

Une "sandbox Kunai" est une solution qui permet d'exécuter un binaire malveillant en enregistrant son activité réseau et système. Elle est basée sur Kunai et la virtualisation KVM.

Incus est un gestionnaire de machines virtuelles utilisant KVM et de conteneurs (LXC ou docker) qui nous permettra de mettre en oeuvre de la virtualisation imbriquée utile pour ce TP.

Dans *la première partie* de ce TP on se familiarisera avec Kunai et Nushell qui pré-installés dans la VM Debian 13 créée avec Incus. L'installation de cette VM est automatisée.

Dans la *seconde partie* de ce TP vous essaierez d'analyser le comportement d'un malware Linux lancé dans une "Sandbox Kunai" en corrélant les traces produites par Kunai avec un article de blog décrivant ce malware nommé "perfctl". Vous utiliserez Suricata pour analyser l'activité réseau du malware.

Vous serez évalués sur:

- Individuellement sur trois questions notées durant la première partie du TP (la célérité et la justesse de vos réponses).
- En bînôme sur la mise en place de Yara-X, qui permet de reconnaître des malwares à partir de signatures. Vous l'appliquerez sur les fichiers binaires malveillants pré-chargés dans votre VM debian.
- En binôme sur la qualité de votre analyse des traces cyber et votre capacité à le présenter oralement à l'enseignant.

1 "Build" de l'Environnement du TP

1.1 Description de l'environnement de TP

Le TP doit être réalisé sur une machine virtuelle Debian 13 supportant la virtualisation imbriquée. [1]

On préservera ainsi l'intégrité de la machine Physique hôte en limitant les risques de contamination par le malware.

L'utilisation d'une "sandbox Kunai" directement sur la machine physique hôte est interdite sauf indication formelle de l'enseignant.

1. La vérification de l'activation de la virtualisation imbriquée est décrite sur `https://www.linux-kvm.org/page/Nested_Guests`

La "sandbox kunai" est une machine virtuelle imbriquée dans la machine virtuelle debian 13.

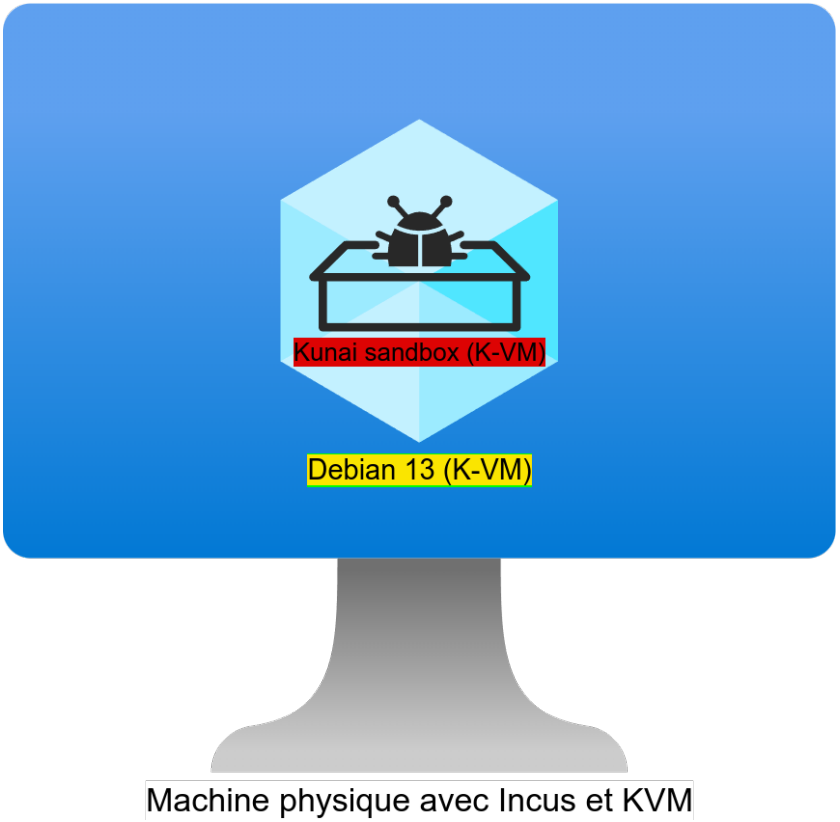

FIGURE 1 – Virtualisation imbriquée avec Incus et la "sandbox Kunai"

1.2 Installation d'Incus et création de la VM Debian 13

On va tout d'abord installer Incus sur votre hôte Linux à l'aide des scripts du projet incus-sandbox.

```
git clone https://gitlab.com/u8668481468-group/incus-sandbox.git
cd incus-sandbox
./install-incus.sh
```

Puis créer la VM Debian 13 supportant la virtualisation imbriquée à l'aide d'Incus.

```
# créer le profil pour la VM debian 13 supportant la virtualisation imbriquée
incus profile create kunai-sandbox < sandbox Kunai.profile

# créer la VM Debian 13 avec l'aide du profil sandbox Kunai précédent
incus launch images:debian/13/cloud sandbox-vm --vm --profile kunai-sandbox

# allez boire un café pendant l'initialisation de la VM (5 minutes pas plus)
...
# accéder à la VM.
incus shell sandbox-vm
# vérifiez que l'installation de la VM s'est bien passée
tail -f /var/log/cloud-init-output.log
```

On va d'abord utiliser cette VM afin d'expérimenter le fonctionnement de Kunai et l'analyse des traces produites par ce dernier à l'aide de Nushell.

2 Premiers pas avec Kunai et Nushell

Kunai va enregistrer les appels systèmes réalisés dans la machine debian démarrée à l'aide d'Incus.

Les fichiers de traces produits par Kunai sont au format "JSON multilines".

Ils peuvent être lus ou convertis dans d'autres formats comme "parquet" (un format "csv" moderne), puis analysés par des outils comme "Nushell" et son plugin "polars" bien adaptés à l'analyse de données.

Polars est un framework de manipulation de données , écrit en Rust, qui offre des performances élevées pour le traitement de grands ensembles de données.

Le plugin "polars" de Nushell permet d'intégrer les fonctionnalités de Polars dans Nushell (similaire à un binding Python).

La documentation de Kunai est disponible sur `https://why.kunai.rocks/`

1. Lancez Kunai sous NuShell durant 5 minutes. Effectuez des actions (installer des paquets, éditer des fichiers, ...)

```
cd
# basculez sous Nushell
sudo su -s  /bin/nu
# copiez kunai dans /usr/local/bin pour le mettre dans le PATH
chmod +x ./kunai
cp ./kunai /usr/local/bin/kunai
# Lancez kunai pour une durée de 5 minutes et enregistrez les traces dans /tmp/kunai.ndjson
kunai run --include=all | tee {save -f /tmp/kunai.ndjson}
```

2. Créez et exportez la variable d'environnement NUKUNAI qui pointera sur le répertoire "/root/nukunai".

```
$env.NUKUNAI = '/root/nukunai'
```

3. Transformez le fichier de traces au format parquet à l'aide de nukunai:

```
nu ($env.NUKUNAI)/kunai_to_parquet.nu  /tmp/kunai.ndjson
```

   Le fichier /tmp/kunai.ndjson.parquet est créé. Explorons-le avec Polars.

4. Affichez une "dataframe" Polars avec toutes les colonnes du fichier parquet et retrouvez vos actions réalisées durant les 5 minutes d'observation:

```
polars open /tmp/kunai.ndjson.parquet|polars into-nu|flatten --all|flatten --all |explore
```

5. Affichez le schéma des données du fichier parquet (dataframe polars), puis les colonnes du dataframe Nushell:

```
polars open /tmp/kunai.ndjson.parquet | polars schema
polars open /tmp/kunai.ndjson.parquet|polars into-nu|flatten --all|flatten --all |columns
```

6. Affichez la table de tous les évènements possibles que Kunai peut générer avec "print_events_table.nu"

```
nu ($env.NUKUNAI)/kunai_print_events_table.nu
```

7. Visualisez les évènements Kunai avec kunai_events_analysis.nu et regardez la documentation des évènements sur `https://why.kunai.rocks/docs/events/` pour comprendre les différents champs listés.

```
nu ($env.NUKUNAI)/kunai\_events\_analysis.nu /tmp/kunai.ndjson.parquet
```

Maintenant ces trois questions vont être évaluées par l'enseignant:

8. A l'aide de Polars sélectionnez la colonne "command_line" et affichez les 10 premières commandes passées les plus fréquentes.
9. A l'aide du scripts filter_events.nu affichez les évènements de type "execveat" et "execve":

   Vous pouvez repasser sous "bash" avec la commande exit.
10. Installez Kunai comme service systemd et bloquez sa désactivation "live". Pourquoi est-ce une configuration conseillée ?

```
sudo ./kunai install --systemd --enable-unit --harden
vim /etc/kunai/config.yaml # modifiez les options selon vos besoins rotation des logs..
systemctl start kunail
sudo systemctl start 00-kunai
```

Signalez-vous auprès de l'enseignant pour validation de cette partie.

3 Analyse du comportement d'un "malware" avec "Kunai Sandox"

3.1 Outils présents dans la "sandbox Kunai"

Kunai va enregistrer les appels systèmes réalisés par le malware lors de son exécution dans la sandbox.

Les fichiers de traces produits par Kunai sont au format "JSON multilines".

Ils peuvent être lus, convertis dans d'autres formats comme "parquet" (un format "csv" moderne), puis analysés par des outils comme "Nushell" et son plugin "polars" bien adaptés à l'analyse de données.

La documentation de Kunai est disponible sur `https://why.kunai.rocks/`

L'activité réseau de la sandbox sera aussi enregistrée par "tcpdump" dans un fichier "pcap" qui pourra être analysé avec "Suricata".

Sous /root/malware-dataset vous trouverez des fichiers binaires ELF malveillants.

Ne rendez pas exécutables les fichiers mis à disposition !!! Ne les exécutez pas directement sur votre machine virtuelle !!!

3.2 Préparation de la Sandbox Kunai avec ks-sandbox-init

1. Dans une "directory" "projetm", créez une sandbox nommée "sandbox1" avec la commande ks-sandbox-init.

```
cd /root/projetm
ks-sandbox-init /root/images/debian/debian-12-genericcloud-amd64.qcow2 \
        $SANDBOXES/sandbox1 --kunai-bin /root/kunai
```

3.3 Analyse d'un "Malware" Linux avec sandbox Kunai

On va analyser un des malwares Linux (perfctl) situés dans /root/malware-dataset/linux/22e4a57ac560ebe1eff8957906589f

```
cd /root/malware-dataset/linux/22e4a57ac560ebe1eff8957906589f4dd5934ee555ebcc0f7ba613b07fad2c13
```

1. Vérifiez que le malware est bien un binaire *statique* au format elf64 via les commandes file et readelf.

```
file ./22e4a57ac560ebe1eff8957906589f4dd5934ee555ebcc0f7ba613b07fad2c13
readelf -a ./22e4a57ac560ebe1eff8957906589f4dd5934ee555ebcc0f7ba613b07fad2c13
readelf -d  ./22e4a57ac560ebe1eff8957906589f4dd5934ee555ebcc0f7ba613b07fad2c13
```

2. Lancez le malware dans la sandbox sandbox Kunai et analysez les traces produites par kunai. Vous pouvez modifiez le temps de vie de la "sandbox" en modifiant le fichier de configuration.

```
export VIRUSDIR="/root/malware-dataset/linux"
cd $SANDBOXES/sandbox1
mkdir -p analyse_virus1
export ANALYSE1="$SANDBOXES/sandbox1/analyse_virus1"
export VIRUS1="$VIRUSDIR/22e4a57ac560ebe1eff8957906589f4dd5934ee555ebcc0f7ba613b07fad2c13"
sandbox Kunai -c $SANDBOXES/sandbox1/config.yaml -f \
            -o $ANALYSE1 -- $VIRUS1/22e4a57ac560ebe1eff8957906589f4dd5934ee555ebcc0f7ba613b07fad2c13
```

3. Expliquez le fonctionnement de la "sandbox" en faisant un schéma d'architecture que vous montrerez à votre enseignant..
4. Rédigez une analyse du "malware" au travers des traces produites dans $ANALYSE1 par Kunai et Suricata en vous aidant de la première partie du TP. [2]
5. Utilisez suricata pour analyser les traces réseaux du fichier dump.pcap.

```
apt install suricata
suricata-update
mkdir suricata_output
cd suricata_output
suricata -S /var/lib/suricata/rules/suricata.rules -v -r  ./dump.pcap
sudo su -s  /bin/nu
cat eve.json |from json -o |flatten --all|flatten --all|explore
```

6. Extrayez des alertes Suricata via Nushell.

   Faites valider cette question par l'enseignant.
7. Installez Yara-X dans la sandbox Kunai et utilisez-le afin de classifier les fichiers binaires malveillants situés dans /root/malware-dataset/linux.
8. Présentez en binôme oralement (trois minutes au maximum) ce rapport à votre enseignant au travers d'un petit support. Il sera aussi à rendre en binôme sur Moodle.
9. **Effacez impérativement la machine virtuelle VMware ayant servi au TP**. Faite valider cette action par l'enseignant.

```
exit # pour sortir de la VM
incus rm -f sandbox-vm
```

2. Vous pouvez vous aider de cet article de la société trendmicro