Master 2 SeCReTS

TP Surveillance et gestion d'incidents
Collecte d'évènements et audits

Première partie

Contexte

L'objectif de ce TP est de mettre en place des mécanismes de capture de traffic réseau et de collecte de logs, puis d'effectuer des manipulations d'audit pour avoir des données intéressantes à collecter. La mise en place de l'architecture, puis l'évaluation de sécurité, nous permettront de revoir plusieurs concepts et outils étudiés dans les cours précédents. En particulier, nous reverrons des notions abordées en cours de sécurité réseau, de cartographie réseau, de sécurité système, et enfin de forensics.

Nous allons donc commencer par mettre en place une architecture représentative d'un réseau à surveiller, avec des services potentiellement ciblés, une machine de collecte, et enfin une machine d'audit. Sur la machine de collecte, nous allons mettre en place la capture du traffic réseau et la collecte des logs à l'aide d'outils simples n'effectuant aucun traitement supplémentaire.

Ensuite, nous allons mettre en place une machine hébergeant des services vulnérables, puis une machine avec des outils d'audit et d'attaque. Nous procéderons enfin à une évaluation de sécurité de la machine vulnérable, tout cela sous la surveillance de la machine de collecte d'évènements de sécurité.

Nous allons manipuler plusieurs outils pendant ce TP, sur différentes machines. Il s'agit essentiellement de :

— `rsyslog`
— `tcpdump`
— `net2pcap`
— `argus`
— `nmap`
— `metasploit`
— `openvas`

Téléchargements :

— Image vmcollecte.ova : `http://moutane.net/uvsq/UE402-Surveillance-Gestion-dincidents/vmcollecte_2020.ova`
— Image Metasploitable2 : `http://moutane.net/uvsq/UE402-Surveillance-Gestion-dincidents/Metasploitable2-Linux.ova`

Deuxième partie

Mise en place de l'architecture de surveillance et d'audit

1 Machine de collecte

L'élément central de l'architecture du TP sera la machine virtuelle de collecte. Afin de simplifier la mise en place de cette architecture, et de ne pas trop augmenter le nombre de machines virtuelles, la machine de collecte assure également la fonction de routeur.

En tant que routeur, cette machine connecte 2 réseaux qui nous intéressent particulièrement :

— le réseau des services nommé `reseau_interne` dans la configuration VirtualBox ;
— le réseau des attaquants nommé `reseau_externe`.

Par commodité, la machine contient également une interface configurée en NAT pour l'accès Internet, et une interface en Host-only pour l'accès SSH depuis l'hôte.

➤ *Importez et démarrez la machine de collecte.*

La machine de collecte assure un autre service important, le DHCP pour l'attribution d'adresses IP à toutes les autres machines raccordées.

➤ *Vérifiez que le service dnsmasq est bien actif avant de démarrer d'autres machines.*

Si ce n'est pas le cas, un simple `systemctl start dnsmasq` devrait suffire.

Pour la suite des manipulations, nous allons donc raccorder les autres machines virtuelles aux 2 réseaux nommés interne ou externe suivant leurs rôles.

2 Configuration de la collecte d'évènements

Avant de démarrer les manipulations d'audit, nous allons mettre en place la collecte d'évènements de sécurité, soit principalement ici la capture réseau et la collecte de logs.

2.1 Capture réseau

Nous allons mettre en place une capture intégrale sur l'interface raccordée au réseau de services. Identifiez quelle est cette interface, vous aurez besoin de son nom.

➤ *Quelle est l'interface raccordée au réseau de services sur votre machine de collecte ? (Question facile pour voir si vous suivez)*

Vous allez écrire un fichier nommé par exemple `capture.sh` qui contiendra la ligne de commande effectuant la capture intégrale sur l'interface mentionnée précédemment, avec l'outil `tcpdump`. Vous devrez également prévoir un dossier sur le disque de la machine virtuelle où enregistrer les captures réseau (par exemple `/capture`). Cette ligne de commande devra donc comporter, entre autres :

— le nom de l'outil (`tcpdump`) ;
— le nom de l'interface ;
— la taille de capture des trames réseau, par défaut cet outil ne capture pas l'intégralité de chaque trame ;
— l'emplacement du fichier dans lequel écrire les trames capturées.

➤ *Quelle est la ligne de commande que vous avez écrite ?*

Pour aller plus loin, vous pouvez ajouter des actions ou tests supplémentaires dans le script, comme par exemple renommer le fichier de capture avant ou après le lancement de l'outil pour ne pas perdre les captures précédentes.

➤ *Vous pouvez tester si la capture fonctionne en envoyant des trames, par exemple avec* `ping`*, sur l'interface surveillée, puis en relisant le fichier de capture pour vérifier que vous voyez vos trames.*

2.2 Capture des flux réseau

Un autre service de capture est présent sur la machine virtuelle, il s'agit de l'outil `argus`. C'est un outil de capture réseau qui ne conserve que les métadonnées des flux réseau, c'est à dire les adresses et ports source et destination, les volumes et nombres de paquets échangés.

Vous pouvez vérifier si le service est bien en cours de fonctionnement, il s'agit de l'unité systemd qui s'appelle `argus-server.service`.

2.3 Collecte des logs

Nous allons mettre en place la collecte des logs sur notre machine virtuelle. Pour cela nous allons utiliser le logiciel `rsyslog`.

Le logiciel `rsyslog` est déjà présent et actif sur la machine, pour la collecte locale des logs, et l'enregistrement dans des fichiers dans `/var/log`. Nous allons donc créer une seconde instance de ce programme, qui prendre en charge les logs arrivant par le réseau, et le stockage dans des fichiers à part.

Pour créer une seconde instance de `rsyslog`, un second service rsyslog a été ajouté dans le gestionnaire `systemd`. Pour démarrer ce nouveau service, le fichier `/etc/systemd/system/rsyslog-collecteur.service` a été créé. Ce fichier crée un service `systemd` qui s'appelle `rsyslog-collecteur`.

Voici le contenu du fichier :

```
[Unit]
Description=Service de collecte de logs

[Service]
Type=notify
ExecStart=/usr/sbin/rsyslogd -n -f /etc/rsyslog-collecteur.conf \
                             -i /var/run/rsyslog-collecteur.pid
Restart=on-failure

[Install]
WantedBy=multi-user.target
```

➤ *Expliquez la ligne de commande pour démarrer ce nouveau service.*

Il faut également ajouter le nouveau fichier de configuration mentionné dans l'unité `systemd`. Examinez le contenu du fichier de configuration `/etc/rsyslog-collecteur.conf`.

Voici les lignes essentielles à retrouver dans le fichier de configuration :

— Ecoute sur les ports TCP et UDP 514 ;

— Enregistrement des logs dans des fichiers spécifiques par machine.

Troisième partie

Configuration de la machine de services

3 La machine virtuelle Metasploitable

Nous avons choisi de vous faire travailler sur la machine virtuelle créée par le projet Metasploitable. Il s'agit d'une machine faite pour apprendre à utiliser Metasploit, elle présente donc un certain nombre de vulnérabilité. Il n'est pas recommandé de la mettre sur Internet !

Vous allez donc pouvoir télécharger et importer cette machine virtuelle. Elle est fournie par le projet au format VmWare mais il est très simple de l'utiliser avec VirtualBox. Nous vous fournissons directement la version utilisable avec ce dernier.

4 Configuration du renvoi des logs

Une chose importante à configurer dans la machine virtuelle Metasploitable pour la suite du TP est le renvoi des logs vers la machine de collecte. Comme c'est une très ancienne version d'Ubuntu, le fichier de configuration est `/etc/syslog.conf`. Il faut ajouter la ligne suivante de préférence vers le début du fichier :

```
*.* @192.168.1.254
```

➤ *Expliquez précisément ce que fait cette configuration.*

Quatrième partie

Déroulement de l'évaluation de sécurité

Les manipulations décrites ici sont testées avec Kali Linux, soit en mode Live, soit avec une installation. Il est possible de faire les manipulations avec un autre système, il faut disposer des outils suivants :

— nmap ;
— metasploit ;
— postgresql-server.

5 Cartographie du réseau de services

Nous allons d'abord commencer par effectuer une cartographie progressive des services présents sur la plage d'adresses du réseau de services. Nous allons pour cela utiliser l'outil `nmap`.

➤ *Commencez par effectuer un scan de type Ping pour identifier les adresses IP qui répondent.*

Nmap possède un ensemble de tests pour identifier précisément les services et leurs versions, en effectuant seulement des requêtes par le réseau. Procédez en 2 temps, faites d'abord la liste des ports TCP qui sont ouverts, puis faites le scan d'identification des services.

➤ *Quelles sont les commandes que vous avez utilisées ?*

6 Utilisation de Metasploit dans une optique d'audit

Nous avons déjà abordé certains aspects de l'utilisation de Metasploit, ici nous allons voir comment l'utiliser pour effectuer un audit un peu plus exhaustif d'une machine. En particulier, dans ce mode, Metasploit utilise une base de données pour enregistrer des résultats de commandes nmap et peut automatiser un certain nombre d'actions.

Avant de lancer Metasploit, il faut créer une base de données. Il faut disposer d'un serveur PostgreSQL. Vous pouvez créer la base de données avec la commande `msfdb init`. Une fois cette commande effectuée avec succès, vous pouvez lancer `msfconsole`, ou d'autres frontend de Metasploit.

Maintenant vous pouvez utiliser, entre autres, la commande `db_nmap`. Elle utilise la vraie commande nmap pour scanner des adresses IP puis stocke les résultats dans la base de données de Metasploit. Ensuite vous avez accès à diverses commandes comme `workspace`, `hosts`, `services`, `creds` et `loot` pour naviguer dans les informations stockées dans la base de données.

Le tutoriel disponible ici est un bon descriptif des différentes commandes disponibles avec la base de données : `https://www.offensive-security.com/metasploit-unleashed/using-databases/`.

7 Pour aller plus loin

Vous pouvez trouver des manipulations avancées décrites dans les tutoriels sur Metasploitable 2 disponibles ici : `https://www.hackingtutorials.org/category/metasploit-tutorials/`

Cinquième partie
(Bonus) Mise en place d'un audit automatique

8 L'outil OpenVAS

Attention, pour cette partie, il est préférable d'avoir une installation de Kali plutôt qu'une version Live. En effet, l'outil OpenVAS va télécharger des fichiers très volumineux pendant sa configuration.

La mise en place d'audits automatisés peut se faire avec un outil Open Source comme OpenVAS. Afin de commencer, il est nécessaire d'installer l'outil. Sous Kali :

```
# apt-get install --no-install-recommends openvas
```

On désactive l'installation des dépendances recommandées afin de réduire la taille de l'installation.

Après, vous pouvez démarrer la configuration de l'outil en suivant le tutoriel disponible ici : `https://www.kali.org/penetration-testing/openvas-vulnerability-scanning/`

➤ *Configurez un scan pour l'adresse IP de la machine Metasploitable et observez les résultats. Comparez avec ce que vous avez pu observer avec Metasploit.*