Document Technique : ConTacGen - Plugin de Simulation d'Attaques Réseau pour Weka

Adan Raynaud, William Benedico, Rayan Rochdi, Frederic Dinahet

Supervised by: Nida Meddouri

EPITA, CYBER 2

1 Introduction

La gestion efficace des données est cruciale en apprentissage automatique. Le plugin ConTackGen a été initialement développé par des étudiants pour automatiser la création de grands ensembles de données destinés à l'apprentissage automatique, en particulier pour les attaques DDoS. Cette automatisation permet de gagner du temps et de réduire les erreurs. Cependant, le plugin était limité par des problèmes de génération de données et sa seule attaque disponible. Nous l'avons mis à jour pour qu'il fonctionne correctement et soit plus compréhensible pour le grand public. De plus, nous avons ajouté l'attaque Man In The Middle (MITM), mais nous n'avons pas eu le temps de la finaliser.

2/ Prérequis et Compatibilité

Dans un premier temps, il convient de noter que le plugin a été développé en utilisant IntelliJ IDEA et Visual Studio Code.

Pour assurer le bon fonctionnement de **ConTacGen 3.0**, les prérequis suivants doivent être respectés. Le projet a été testé et validé avec les versions de logiciels et systèmes d'exploitation ci-dessous :

Systèmes d'Exploitation Compatibles

- **Ubuntu** : 18.04, 20.04, 22.04
- **Windows** : 10, 11
- **Kali Linux** : 2022.4
- **Linux Mint** : 21.3

Logiciels Requis

- **Weka** : Version 3.9.6 et supérieure.
- **JDK (Java Development Kit)** : Version 11 et supérieure.
- **Docker Desktop** : Version 4.29.0 et supérieure (pour Windows).
- **Apache Ant** : Version 1.10.2 et supérieure.

3/ Organisation du Projet

L'architecture de **ConTacGen 3.0** est organisée comme suit :

```
ConTacGen/
 Logo/
    .gitkeep
    shared_image.jpg

 misc/
    docker/
        serverattack/
           Dockerfile
           lien.txt
        udattack/
            Dockerfile
            lien_image.txt
            payload.sh

 scientific_paper/
    ConTacGen.pdf
    ConTacGen.zip
    Papier_technique.pdf

 src/
    main/java/fr/
        contacgen/
           ConTacGenPacketHandler
           ConTacGenUtils
           DockerRunner
           MITMAttack
           PacketData
           SSHAttack
           UDPDos

        hyper.testpcap/
           TcpUdpPacketHandler
           TestPcap

        weka.datagenerators.classifiers.classification/
            ConTacGen

 .gitignore
 build.xml
 Description.props
 LICENSE
 pom.xml
 README.md
 resultExample.png
```

4/ Description des Fichiers Techniques

4.1 `misc/docker/serverattack/Dockerfile`

- **Chemin** : `misc/docker/serverattack/Dockerfile`
- **Rôle** : Configure un conteneur pour un serveur d'attaque basé sur Ubuntu.
- **Fonctionnalités principales** :

- Installe et configure Apache (`port 80`) et SSH (`port 22`).
- Démarre automatiquement les services lors de l'exécution.

4.2 `misc/docker/udattack/Dockerfile`

- **Chemin** : `misc/docker/udattack/Dockerfile`
- **Rôle** : Configure un conteneur pour une attaque UDP.
- **Fonctionnalités principales** :
  - Installe `tshark`, `tcpdump`, et `openjdk-11-jre`.
  - Exécute le script `payload.sh`.

4.3 `misc/docker/udattack/payload.sh`

- **Chemin** : `misc/docker/udattack/payload.sh`
- **Rôle** : Orchestre une attaque MITM et capture le trafic réseau.
- **Fonctionnalités principales** :
  - Démarre un serveur HTTP (`port 8080`).
  - Utilise `tshark` pour capturer et convertir les données en `.pcap`.

4.4 `ConTacGenPacketHandler`

- **Chemin** : `src/main/java/fr/contacgen/ConTacGenPacketHandler.java`
- **Rôle** : Gère les paquets réseau capturés.
- **Fonctionnalités principales** :
  - Analyse les paquets IPv4 et IPv6, et stocke leurs données dans une liste.
  - Permet d'appliquer des actions sur les paquets via des consommateurs.

4.5 `ConTacGenUtils`

- **Chemin** : `src/main/java/fr/contacgen/ConTacGenUtils.java`
- **Rôle** : Fournit des utilitaires pour gérer Docker et les fichiers de capture.
- **Fonctionnalités principales** :
  - Vérifie et exécute des conteneurs Docker.
  - Copie, parse et lit les fichiers PCAP.

4.6 `DockerRunner`

- **Chemin** : `src/main/java/fr/contacgen/DockerRunner.java`
- **Rôle** : Orchestration des attaques réseau via Docker.
- **Fonctionnalités principales** :
  - Configure et exécute des conteneurs pour les attaques UDPDos et MITM.
  - Traite les données capturées à l'aide de `ConTacGenPacketHandler`.

4.7 `PacketData`

- **Chemin** : `src/main/java/fr/contacgen/PacketData.java`
- **Rôle** : Représente un paquet réseau et ses métadonnées.
- **Fonctionnalités principales** :
  - Extrait des informations sur les paquets (IP, protocole, taille, etc.).
  - Vérifie la présence d'une attaque dans le contenu des paquets.

4.8 `SSHAttack`

- **Chemin** : `src/main/java/fr/contacgen/SSHAttack.java`
- **Rôle** : Simule une attaque brute-force SSH.
- **Fonctionnalités principales** :
  - Implémente des connexions SSH pour exécuter des commandes.
- **Note** : Classe partiellement fonctionnelle nécessitant des ajustements.

4.9 UDPDos

- **Chemin** : src/main/java/fr/contacgen/UDPDos.java
- **Rôle** : Simule une attaque par déni de service (DoS) via le protocole UDP.
- **Fonctionnalités principales** :
  - Envoie des paquets avec des charges utiles aléatoires.
  - Implémente l'interface Runnable pour des exécutions asynchrones.

4.10 TcpUdpPacketHandler

- **Chemin** : src/main/java/fr/hyper/testpcap/TcpUdpPacketHandler.java
- **Rôle** : Fichier de test pour analyser les paquets réseau capturés.
- **Fonctionnalités principales** :
  - Vérifie le protocole de chaque paquet (TCP, UDP, IPv4, IPv6).
  - Affiche les charges utiles des paquets pour validation.

4.11 TestPcap

- **Chemin** : src/main/java/fr/hyper/testpcap/TestPcap.java
- **Rôle** : Fichier de test pour lire et parcourir un fichier PCAP.
- **Fonctionnalités principales** :
  - Utilise TcpUdpPacketHandler pour analyser les paquets.

4.12 ConTackGen

- **Chemin** : src/main/java/weka/datagenerators/classifiers/classification/ConTackGen.java
- **Rôle** : Générer des ensembles de données simulant des attaques réseau (UDPDos, MITM) dans le framework Weka.
- **Fonctionnalités principales** :
  - Configuration des attaques via le type (AttackType) et la durée (duration_s).
  - Définit un format de dataset compatible Weka, incluant des attributs tels que les adresses IP, le protocole, et les indicateurs d'attaque.
  - Simule et capture des paquets réseau à l'aide de Docker.
- **Interactions avec d'autres classes** :
  - DockerRunner : Orchestration des attaques en utilisant Docker.
  - ConTacGenPacketHandler : Capture et gère les paquets réseau.
  - UDPDos et MITMAttack : Implémentent respectivement les attaques UDPDos et MITM.
  - PacketData : Fournit un modèle pour représenter les paquets capturés.
- **Attributs du dataset** :
  - srcIP, dstIP, protocol, attack, TTL, content, timestamp, etc.
- **Méthodes importantes** :
  - generateExamples : Orchestration des attaques et capture des données.
  - defineDataFormat : Définit le format du dataset.
  - handlePacket : Convertit les paquets capturés en instances Weka.
  - listOptions et setOptions : Configuration des paramètres via la ligne de commande.

5. Dépendances Java

Cette section détaille les bibliothèques et dépendances Java utilisées dans le projet **ConTacGen 3.0**. Ces dépendances sont essentielles au bon fonctionnement des différentes classes et à l'intégration avec Docker et Weka.

5.1 Bibliothèques Internes Java

Les bibliothèques Java standard utilisées dans les classes du projet incluent :
- java.net.* : Gestion des connexions réseau et des adresses IP (importée dans MITMAttack, UDPDos, SSHAttack).

— `java.io.*` : Lecture, écriture et manipulation des flux et fichiers (importée dans `ConTacGenUtils`, `TestPcap`).
— `java.util.*` : Utilitaires comme les listes, fonctions, vecteurs et générateurs aléatoires (importée dans toutes les classes principales).
— `java.time.*` : Gestion des durées et des dates pour les analyses temporelles (importée dans `ConTacGenPacketHandler`, `ConTackGen`).

5.2 Bibliothèques Externes

Le projet s'appuie sur plusieurs bibliothèques externes pour étendre ses fonctionnalités :
— **Docker Java API (`com.github.dockerjava.*`)** : Gestion des conteneurs Docker, utilisée pour configurer et orchestrer les attaques (`ConTacGenUtils`, `DockerRunner`).
— `io.pkts.*` : Bibliothèque pour analyser et manipuler les fichiers PCAP, utilisée dans les classes manipulant les paquets réseau (`ConTacGenPacketHandler`, `PacketData`, `TcpUdpPacketHandler`).
— **Apache Commons Compress (`org.apache.commons.compress.*`)** : Extraction et manipulation des fichiers TAR depuis les conteneurs Docker (`ConTacGenUtils`).
— **JSch (Java Secure Channel)** : API pour gérer les connexions SSH (`SSHAttack`).
— **Weka (`weka.core.*` et `weka.datagenerators.*`)** : Gestion des instances, des attributs et des options pour le framework Weka (`ConTackGen`).