Interface de test Exchange Multi-versions - JARM

PROHACKTIVE

Mathis Scaffidi Fonti

Sommaire

1. Description

Dans le but de tester et d'identifier automatiquement des failles dans un système, il faut au préalable identifier les versions logiciels qu' embarque un serveur donné.
Pour cela on utilisera l'outil JARM permettant d'obtenir un hash à partir des informations de connexion TLS. En comparant un hash connu à celui de serveur cible, on sera en mesure d'identifier les versions logiciels.

2. Pré-requis

Installation préalable de python.
Nous utiliserons poetry pour l'exécution de jarm.py

Installation de poetry (windows):

```
> (Invoke-WebRequest -Uri
https://raw.githubusercontent.com/python-poetry/poetry/master/get-poetry
.py -UseBasicParsing).Content | python

> pip install --user poetry          #Correction des dépendances
```

Création d'un projet poetry:

```
> poetry new test
```

Lancement de l'environnement virtuel:

```
> poetry install
> poetry shell
```

3. Exécution de jarm

téléchargement du script:
https://github.com/salesforce/jarm/raw/master/jarm.py

Lancement du script dans l'environnement virtuel:

```
> python jarm.py <IpCible>
```

exemple:

```
$ python jarm.py 172.17.0.176
Domain: 172.17.0.176
Resolved IP: 172.17.0.176
JARM: 2ad2ad0000000000002ad2ad2ad2ad0f0dcb2ae084f34cae790be1eab88c30
```

Ce hash obtenu permet d'identifier un hôte unique.
Si les 30 premiers caractères sont les mêmes sur différents hôtes, et que les 32 derniers sont différents, cela signifierait que les serveurs ont des configurations très similaires, acceptant les mêmes versions.

Signatures JARM récoltés:

Exchange 2012:

```
JARM: 26d26d00026d26d22c26d26d26d26d2bb1101b28b790bf5d9d4dcad463fdc2
```

Exchange 2016:

```
JARM: 29d29d00000000021c29d29d29d29d1f4989c319e75da83988253a39553038
```

Exchange 2019:

```
JARM: 2ad2ad0000000000002ad2ad2ad2ad0f0dcb2ae084f34cae790be1eab88c30
```

Afin de comparer les hash, un second serveur exchange 2019 est installé:

Exchange 2019_2:

```
JARM: 2ad2ad0000000000002ad2ad2ad2ad0f0dcb2ae084f34cae790be1eab88c30
```

Avec une installation similaire, on obtient strictement le même hash.