Attaques par injection de faute

Hélène Le Bouder, Ronan Lashermes et **Fabien Autrel**

2023

IMT Atlantique
Bretagne-Pays de la Loire
École Mines-Télécom

Inria

Objectifs pédagogiques

- 

  Comprendre la pertinence des injections de fautes.

- 

  Duplication de code.

Modèle de l'attaquant

- Injection des fautes lors de l'exécution.
- Saut d'une instruction.

Exploration des paramètres

Terminal

```
pin_verif/> python3 ../L3/F-faute-directe.py bin/
```

- faute toutes les instructions,
- une seule à la fois,
- de `Verify_PIN` et `Compare_Arrays`.

Exploration des paramètres

bin/pin.list (extrait)

```
80000d4 <compare_arrays>:
80000d4:    b172          cbz r2, 80000f4
80000d6:    b410          push {r4}
80000d8:    1e43          subs r3, r0, #1
80000da:    3901          subs r1, #1
80000dc:    4410          add r0, r2
80000de:    1e44          subs r4, r0, #1
80000e0:    f813 0f01     ldrb.w r0, [r3, #1]!
```

- Explorer les instructions.
- Quelles instructions sont pertinentes à fauter selon vous ?

À vous de jouer !

- Modifier votre implémentation pour résister à cette attaque.
- Une faute possible par exécution.
- Indice : dupliquer les instructions.

Attaque sur le compteur

Terminal

```
pin_verif/> python3 ../L3/G-faute-compteur-bruteforce.py bin/
```

L'attaquant :

1. teste sur toutes les adresses pour trouver celles qui empêchent le décrément ;
2. combine une faute sur le décrément avec une attaque par force brute.

À vous de jouer !

- Améliorer votre code pour durcir le compteur et résister à l'attaque.

Au revoir

IMT Atlantique
Bretagne-Pays de la Loire
École Mines-Télécom

Inria

illustrations : Le Mooncat, pixabay.