

Analyse automatisée d'une bibliothèque cryptographique



Détection de failles par canal auxiliaire par analyse statique et symbolique

Duzés Florian



Introduction - 1



Introduction - 1

1996 : Paul C. Kocher, *Timing Attacks on Implementations of Diffie-Hellman, RSA, DSS, and Other Systems*

Une mesure précise du temps requis par des opérations sur les clés secrètes permettrait à un attaquant de casser le cryptosystème.



Introduction - 1

1996 : Paul C. Kocher, *Timing Attacks on Implementations of Diffie-Hellman, RSA, DSS, and Other Systems*

Une mesure précise du temps requis par des opérations sur les clés secrètes permettrait à un attaquant de casser le cryptosystème.

2003 : BRUMLEY et BONEH *Remote Timing Attacks Are Practical*



Introduction - 1

1996 : Paul C. Kocher, *Timing Attacks on Implementations of Diffie-Hellman, RSA, DSS, and Other Systems*

Une mesure précise du temps requis par des opérations sur les clés secrètes permettrait à un attaquant de casser le cryptosystème.

2003 : BRUMLEY et BONEH *Remote Timing Attacks Are Practical*

2011 : BRUMLEY et TUVERI *Remote Timing Attacks are Still Practical*



Introduction - 2

HACL*

*"**H**igh **A**ssurance **C**ryptography **L**ibrary"*^a est une bibliothèque cryptographique, écrite en F* ("F star"), implémentant tous les algorithmes de cryptographie modernes et est prouvée mathématiquement sûre.

HACL* est notamment utilisé dans plusieurs systèmes de production tels que Mozilla Firefox, le noyau Linux, le VPN WireGuard...

a. <https://hacl-star.github.io/>



Introduction - 3

- QR1** Est-il possible de propager les garanties de sécurité pendant la compilation ?
- QR2** Est-il possible d'automatiser la détection de ces failles sur des fichiers compilés ?
- QR3** Est-il possible d'appliquer ces mécanismes pour assurer la vérification d'une bibliothèque cryptographique ?



Sommaire

1. Méthodes de protection et limitations

2. Outils de vérifications

3. Automatismes

4. Érysichthon

1. Conception générale
2. Andhrímnir

02

Méthodes de protection et limitations






Usages de compilateur



Bonnes pratiques de programmations

03

Outils de vérifications





Usages de compilateur

04

Automatismes






Usages de compilateur

05

Érysichthon





Usages de compilateur



Usages de compilateur