Plan de l'oral

- Introduction — contexte et objectifs
- Méthodes de protection et limitations
- Outils de vérifications
  - Étude sur cas simple
  - Contraintes et identification des limitations
- Érysichthon
  - Conception générale
  - Andhrímnir
- Résultats
- Annexes

Introduction

- Attaquer sur la sécurité et le besoin d'avoir des libs cryptographiques
- Présenter HACL*
- Historique timing attacks et mise à distance
- Exemple et instructions en temps constant

Axes de défenses

- programmation en temps constant

motivation du stage avec l'article

- compilateurs
  - CompCert => garanties formelles ..... retard sur les standards
  - Jasmine => annotations de codes, execute toutes les branches ..... pas employable sur un projet industriel, artefact de recherche
  - Raccoon => annotations de codes ..... pas le temps constants
  - Constantine => linéarisation ..... 16.36x taille binaire & 27.1x temps
- assembleur

Réalisation

vérification de binaire et continuité des Spécifications

Présentation Érysichthon et résultats

Sommaire

Outils de vérifications

- tableau
vérification de binaire
vérification correcte
    - **Binsec**

Binsec

- analyse au binaire
- couvrir de nombreuses architectures
- permet l'automatisation

Commande de choix de l'outil $\bigoplus$ script d'instruction

EXEMPLE

Automatismes

- Simplification
- Réduction de la taille des binaires
- Script binsec simple

tableaux de résultats
    - identifications des points d'attention
- Variables secrètes et test complet

Cahier des charges

- identification des points clés
- graphes de fonctionnements
- spécialisation x86_64

Érysichthon

- construction en modules - ensemble des scripts et de Makefile

Andhrímnir

- Indépendant : automatique et corrects
- Adapté : facilité l'usage et avancer dans la conception de recherche

Graphe de fonctionnement

Exemple et standardisation

Résultats

- graphes
- discuter des **unknown**

ORDRE des arrêts

- max-depth / timeout / killed
- killed
- syscall / KO / error
- KO / error

Conclusion

(afficher les références)

Annexes

- options de compilation
- construction en vue user
- pourquoi json
- fin de stage / ouvertures autres pb