Sécurité avec Python

ESGI

June 15, 2025

Présentation du module

Sécurité avec Python

Ce module vise à introduire les fondements de la cybersécurité à travers le langage Python.
Python est aujourd'hui un outil central en sécurité offensive (pentesting), défensive (détection, analyse de logs) et en cryptographie.

Pourquoi Python pour la cybersécurité ?

- Facile à lire et à écrire, rapide à prototyper.
- Large écosystème de bibliothèques pour :
  - Analyse réseau : `scapy`, `socket`, `nmap`
  - Cryptographie : `cryptography`, `hashlib`
  - Web & automation : `requests`, `selenium`
- Couramment utilisé par les pentesters et analystes sécurité.

Objectifs du module

Ce que vous apprendrez :

- Comprendre les bases pratiques de la cybersécurité.
- Utiliser Python pour automatiser des tâches de sécurité.
- Implémenter des techniques cryptographiques classiques.
- Réaliser des analyses de vulnérabilités avec Python.

Contenu du module

- **Partie 1 : Cryptographie appliquée**
  - XOR, AES, modes ECB/CBC
  - Attaques simples (padding oracle, pattern ECB)
  - TP : Challenges Cryptopals
- **Partie 2 : Scripting Python pour la sécurité**
  - Automatisation de scan, parsing de logs, détection brute-force
  - TP : Scripts IDS, alertes, logs SSH
- **Partie 3 : Pentesting avec Python**
  - Scan de ports, fingerprinting, bruteforce web
  - TP : Scanner de services + exploitation Web

Compétences visées

- Utiliser Python comme outil de sécurité offensive et défensive.
- Comprendre et implémenter des primitives cryptographiques.
- Identifier des vulnérabilités de services et d'applications.
- Créer des scripts de surveillance ou d'exploitation simples.

Environnement de travail

Outils utilisés dans le module

- **Python 3.8+** avec : `cryptography`, `socket`, `requests`, `nmap`, etc.
- Environnement local (VSCode, terminal) ou machines virtuelles (Kali, Metasploitable2)
- Plateformes de challenge : `cryptopals.com`, `TryHackMe` (optionnel)

Feuille de route du module