

Hacking Cryptography

Abstract

Crypto related bugs are super common. OWASP even ranks “Cryptographic Failure” as the second most common security vulnerability class in software. Yet, very often these vulnerabilities are overlooked by developers, code auditors, blue teamers and penetration testers alike. Because, let’s face it: Nobody knows how cryptography works.

During the course you will:

- understand how modern cryptography works.
- find common crypto vulnerabilities in real software.
- write crypto exploits for real software (and an IoT device).

Using case studies from our own pentesting and red teaming engagements, we’ll introduce core concepts of applied cryptography and how they fail in practice.

No prior knowledge required!

Prerequisite Knowledge

This is a beginner to intermediate course. Students should be familiar with at least one scripting language and have a basic understanding of computer networks.

The contents are compressed, but no prior knowledge of cryptography is needed. Every subject is introduced before attacks are presented.

Equipment Requirements

Participants should bring a laptop with a modern browser to join the virtual learning environment.

Outline

- Introduction to Cryptography
 - Basic Terminology
 - Security Guarantees
 - Composition of Primitives
- Attack Categorization
 - Security Objectives and their Relation to Cryptography
 - Attack Categorization
- Working with Crypto Tools
 - Introduction to Cyber Chef
 - Crypto tools: CryCry Toolkit and OpenSSL
 - *Challenge Lab: CryCry, OpenSSL and Cyber Chef*
- Hacking Encryption

- Stream Ciphers
 - * Introduction to Stream Ciphers
 - * Real World Examples of Vulnerabilities
 - * Attacks on Stream Cipher Uses
 - * *Challenge Lab: (Ab)using Stream Ciphers*
- Block Ciphers
 - * Introduction to Block Ciphers
 - * Modes of Operation
 - * Real World Examples of Vulnerabilities
 - * Attacks on Block Cipher Uses
 - * *Challenge Lab: (Ab)using Block Ciphers*
- Hash Functions
 - Introduction to Hash Functions
 - Real World Examples of Vulnerabilities
 - Password Storage & Cracking
 - *Challenge Lab: (Ab)using Hash Functions and PW Cracking*
- Message Authentication Codes and Authenticated Encryption
 - Introduction to Message Authentication Codes
 - Pitfalls on Trivial Constructions
 - Real World Examples of Vulnerabilities
 - *Challenge Lab: (Ab)using MACs and AuthEnc*
- Attacks on Entropy and Randomness
 - Generating Secure Keys with OS Entropy Pools
 - Misuse of Pseudo Random Number Generators
 - Backdoors and Cleptography
 - Real World Examples of Vulnerabilities
 - *Challenge Lab: Keys and Randomness*
- Asymmetric Crypto with RSA and ECC
 - Introduction to RSA and ECC
 - Key Formats
 - Key Sizes and Brute Force
 - Real World Examples of Vulnerabilities
 - *Challenge Lab: RSA and ECC*
- Public Key Infrastructure and Certificates
 - Introduction to Certificates
 - x509 Certificate Structure and Features
 - Common Certificate Pitfalls
 - Chain of Trust and PKI services
 - TOFU Principle and Man-In-The-Middle Threats
 - *Challenge Lab: Certificates and PubKeys*
- TLS and Friends
 - Introduction to TLS and Similar Protocols
 - TLS Security parameters
 - Exploiting a Man-In-The-Middle position for TLS and VPN
 - Intercepting and Decrypting TLS Traffic for Application Testing
 - Defeat Public Key Pinning with Dynamic Instrumentation

- *Challenge Lab: Intercepting TLS*
- JWTs and JOSE
 - Introduction to JSON Web Tokens and Javascript Object Signing and Encryption
 - Real World Examples of Vulnerabilities
 - *Challenge Lab: Exploiting JWT*
- Passkeys, WebAuthn, FIDO and 2nd Factor Solutions
 - Introduction to Password-Less Authentication
 - TOTP Algorithms and Seeds
 - Passkeys, FIDO2 and WebAuthn
 - Footguns and Examples of Vulnerabilities
- Post-Quantum Cryptography
 - Introduction to Post-Quantum Algorithms
 - Post-Quantum Signatures and KEMs
 - Upcoming Post-Quantum Standards
 - *Challenge Lab: Using OpenSSL with Post-Quantum*
- Farewell
 - Outlook on Future Developments
 - Presentation of Take Home Challenges
 - Recap