CRYPTOGRAPHY

Intro

Created by Alkalem

23.11.2023

OVERVIEW

- Security goals
- Cryptographic Ciphers/Protocols
  - Classical Cryptography
  - Randomness
  - Symmetric Cryptography
  - Asymmetric Cryptography
- Typical Vulnerabilities
- Tools
- Further Topics
- Tasks

SECURITY GOALS

- CIA (Confidentiality, Integrity, Availability)
- Authenticity
- Which protection is necessary for my Application?
- Wich security does a cipher/protocol provide? What are the conditions?

CLASSICAL CRYPTOGRAPHY

- Caesar cipher
  - Each letter is shifted by a fixed value k
  - Rarely used today as well (Spiegel Paywall)
  - Attacks: "try each key" (Bruteforce), frequency analysis

CLASSICAL CRYPTOGRAPHY

- Vigenère cipher
  - choose a key of multiple letters, shift each letter according to the key letter
  - Attacks: determine key length, break Caesar cipher for each position

CLASSICAL CRYPTOGRAPHY

- XOR cipher
  - Encrypt plaintext by XOR with key
  - Attacks: analogous to Vigenère (Bruteforce, Frequency analysis per key byte)
- One Time Pad (OTP)
  - use key with same length as plaintext
  - Information-theoretic security, if key is chosen equally distributed at random and used only once

RANDOMNESS

- Randomness is needed as input for many ciphers
- in most programming languages default pseudo-random number generators (PRNGs) are not cryptographically secure
- cryptographically secure RNGs:
  - /dev/urandom
  - hardware RNGs
  - RNGs of the cryptographic libraries (e.g., secrets or Crypto.Random in python)
- Attacks: recover state, side channels

SYMMETRIC CRYPTOGRAPHY

- Require secure channel or pre-shared secret
- Stream ciphers
  - pseudo-random key stream generated from key
  - key stream is XORed with plaintext stream
- Block ciphers
  - encrypt blocks of fixed length
  - mode of operation used to encrypt multiple blocks

STREAM CIPHERS

- Examples: RC4, SEAL, Salsa, CryptMT
- Linear Feedback Shift Registers (LFSRs)
- Attacks:
  - known plaintext: calculate parts of key stream when parts of plaintext are known
  - key reuse: two messages are encrypted with same key stream, difference (XOR) between plaintexts is observable

BLOCK CIPHERS

- Examples: DES, IDEA, RC5, AES, Blowfish, ...
- block length and key length not necessarily the same
- padding: extend messages to full block length
- Modes of operation:
  - Electronic Code Book (ECB)
  - Cipher Block Chaining (CBC)
  - Counter Mode (CTR)
  - Galois Counter Mode (GCM)
- Attacks:
  - against cipher: differential or linear cryptanalysis
  - against mode of operation

ELECTRONIC CODE BOOK (ECB)

- encrypts each block seperately, independent of each other
- Problems:
  - insertion of blocks possible
  - deterministic

CIPHER BLOCK CHAINING (CBC)

- for plaintext blocks P_i, ciphertext blocks $C_i, i \in 1, \ldots, n$, $C_0 = IV$
- Encrypt: $C_i = Enc(P_i \oplus C_{i-1})$
- Decrypt: $P_i = Dec(C_i) \oplus C_{i-1}$ (can be done in parallel)
- Initialisation vector (IV) chosen at random
- Problems: loss of one ciphertext block results in loss of two plaintext blocks
- Attacks: POODLE

(GALOIS) COUNTER MODE

- for plaintext blocks P_i, $i \in 0, \ldots, n-1$, $C_0 = IV$
- Encrypt / Decrypt: $C_i = F(IV + i) \oplus P_i$ (can be pre-computed)
- Initialisation vector (IV) chosen at random
- Problems: XOR malleability, counter non-random or reused
- Galois Counter Mode: add signature to detect modification

ASYMMETRIC CRYPTOGRAPHY

- Establish secure channel with remote party
- Key-Exchange
- Encryption
- Signing

RSA - KEY GENERATION

- Choose large primes p and q
- Calculate modulus $N = pq$
- Calculate $\Phi(N) = (p-1) * (q-1)$
- Choose e with $gcd(e, \Phi(N)) = 1 \land 1 < e < \Phi(N)$
- Calculate d as inverse of e ($e * d \equiv 1 \mod \Phi(N)$)
- Public key: N, e
- Private key: d

RSA

- Encryption: $c = m^e \mod N$
- Decryption: $c^d = m^{ed} = m^{ed \mod \Phi(N)} = m^1 \mod N$
- RSA without special padding is homomorphic ($Enc(m_1, pk).Enc(m_2, pk) = Enc(m_1.m_2, pk)$) and deterministic
- use RSA-OAEP if that is problematic

RSA - ATTACKS

- factoring N has complexity of about $exp(log(N)^{\frac{1}{3}}(loglogN)^{\frac{2}{3}})$, infeasible for reasonable choice of N
- in some cases, attacks in polynomial time possible:
  - small private exponent d ($d < \frac{1}{3}N^{\frac{1}{4}}$): Wiener's attack
  - for small public exponent or partially known prime factor: Coppersmith's attack
  - $m < N^{\frac{1}{e}}$: calculate message as root of ciphertext
  - message sent to many recipients using same public exponent: Hastad's Broadcast Attack

ELLIPTIC CURVES

- Elliptic Curve equation:
- Group: Generator point G, point addition and multiplication with natural number
- Cyclic EC group over $\mathbb{Z}_p, p > 3$
- Point (x, y) on curve iff $y^2 \equiv x^3 + ax + b \mod p$, plus (imaginary) point at infinity O, $a, b \in \mathbb{Z}_p$ with $4a^3 + 27b^2 \not\equiv 0 \mod p$
- Harder to attack, can use smaller keys for same security level

TYPICAL VULNERABILITIES

- Implementation mistake: incorrect/vulnerable custom implementation, incorporated incorrectly into application
- Conceptual mistake: incorrect use or not sufficient for use case
- Theoretic mistake: violated condition for security, advanced maths or theoretic computer science necessary, "read the paper"

TOOLS

- CyberChef
- https://factordb.com/
- sagemath (free open-source mathematics software system)
- Z3 (theorem prover)

FURTHER TOPICS

- Post-quantum cryptography (e.g., lattices)
- Pairing-based cryptography
- Zero Knowledge
- Last currently planned intro talk, ask for topics you are interested in or just make a talk yourself

TRY IT OUT

- https://intro.kitctf.de/
- Other challenges:
  - https://cryptohack.org/ (Easy to hard, with good explanations)
  - https://cryptopals.com/ (Implement cryptosystems and attacks)
  - https://overthewire.org/wargames/krypton (Classical crypto)