CS-438 Decentralized Systems Engineering

Fall 2024

Week 7

Adversaries and threat modeling

no system is 100% secure

assess relevant scopes:

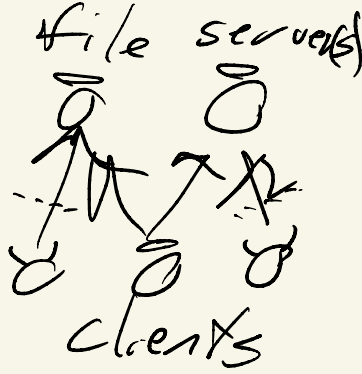
- assets - what needs to be protected
- boundaries - administrative domains, enclaves
- adversaries - realistic attacks, motivations

Algorithms — make security assumptions

nodes trusted / not

example: client/server

often assume servers trusted, clients not



extremes impractical:

- "trust everyone" — no security
- "trust no one" — no way to design systems

Adversaries – categorizations (more realistic)

Boundaries, admin domains — internal adversaries vs external adversaries

Local vs global

Ephemeral vs persistent ("advanced persistent threat" APT)

Passive vs active – Byzantine

("honest but curious"

Threshold assumptions

- f of n nodes are faulty / malicious
- >50% mining power is "good"

Common threat vectors

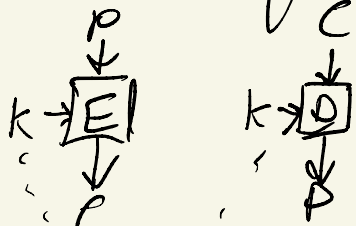
STRIDE mode

- Spoofing
- Tampering
- Repudiation
- Information disclosure (privacy leaks)
- Denial of service
- Elevation of privilege

Cryptography basics - review

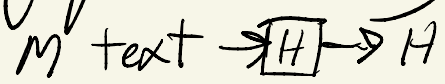
Symmetric crypto - both/all parties share a key

- symmetric encryption



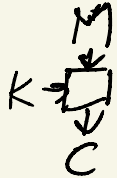
attacker get C
hard to recover P

- cryptographic hashes



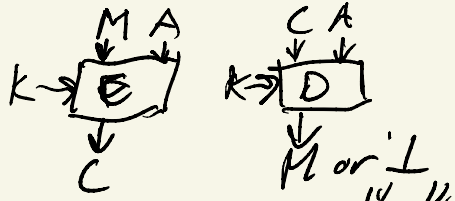
2 properties: non-invertible, collision-resistance

- message authentication checks (MAC)
  example: HMAC code HMAC(k,M)



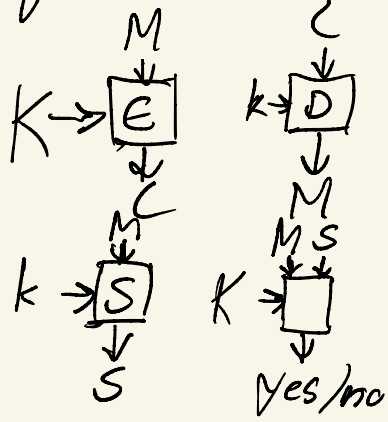
attacker sees M, C not k

- authenticated encryption (AEAD)
  (w/ "additional data")



Asymmetric crypto

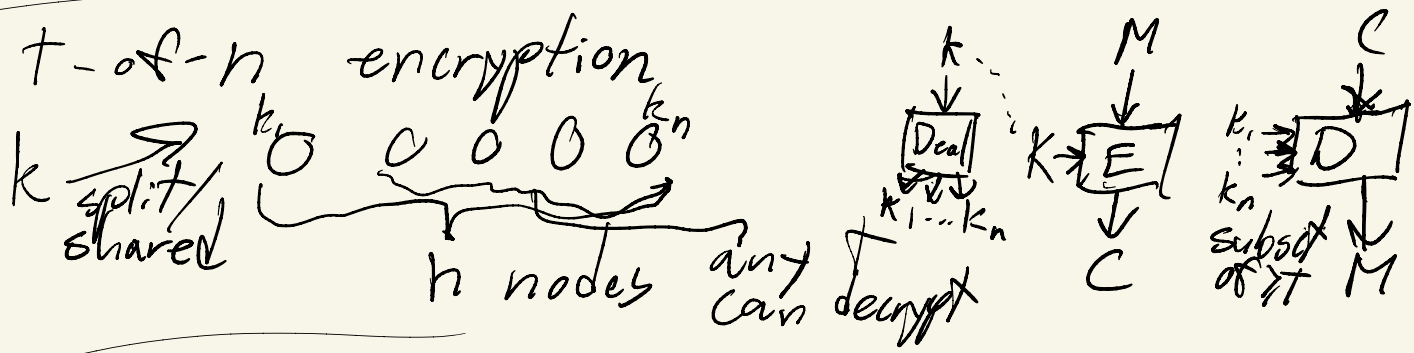
- keys in pairs: (k, K) — k: private, K: public
- public-key encryption (RSA, DH, ECDH, ...)



- digital signatures

Threshold cryptography

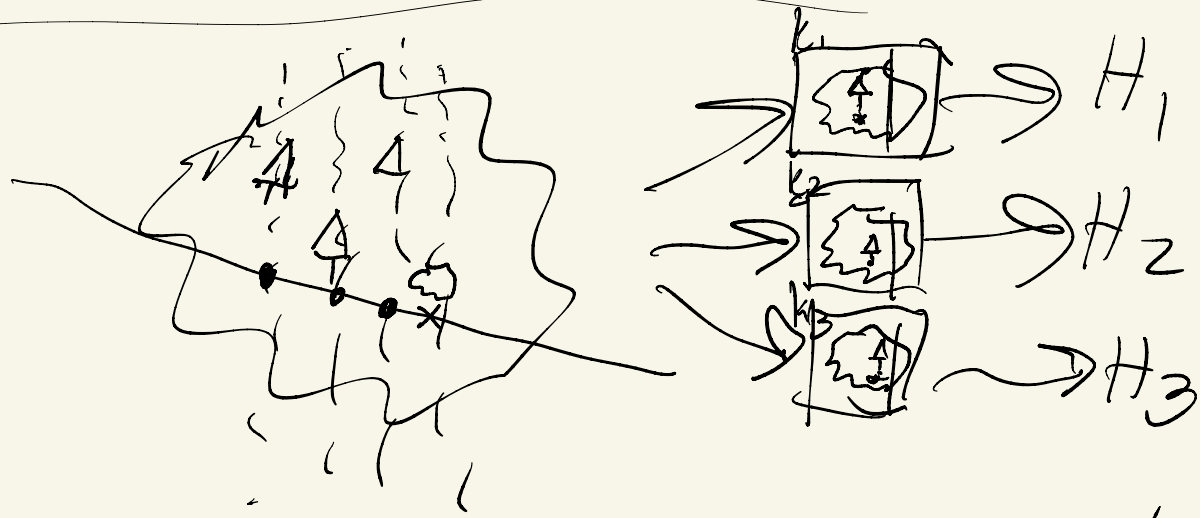
- t-of-n encryption



typical implementation:

Shamir secret sharing

example: 2-of-3 Pirate treasure



general: t+1 degree polynomial

Lagrange interpolation