Overview

Cryptography applications

Core

- **Secret key establishment** = find a secret shared key k
- **Secure comunication** (knowing k we can talk privately)

More apps

- **Digital signatures** = I want to sign a digital document
  - Compute a signature using the document and some private key
  - An attacker can't copy my signature
- **Anonymous communication** (Ex: mix net)
  - Alice talks to Bob
  - They don't know who they talked to
- **Anonymous digital cash**
  - Spend a *Digital coin* without knowint who I am
  - How to prevent double spending? => If we spend once it's anonymous, else our identity is revealed

Protocols

- **Elections** = Compute the winner of an election without revealing everything about the votes
  - Ex: each party sends an encryption of the vote s.t the election center can compute then winner but nothing else
- **Private auctions** = The 2nd highest bid and the identity of the winner should be public, nothing else
- **Secure multi party computations**

More magic

- **Privately outsourcing computation**
  - Encrypt the search query and return encrypted results
  - The search engine has no idea what she searched about
- **Zero knowledge proofs**
  - Alice can give a proof about something to Bob

- Bob doesn't know anything bout the solution
- Bob will not learn anything new about the solution

Theorem

> Anything that can be done with a trusted auth. can also be done without

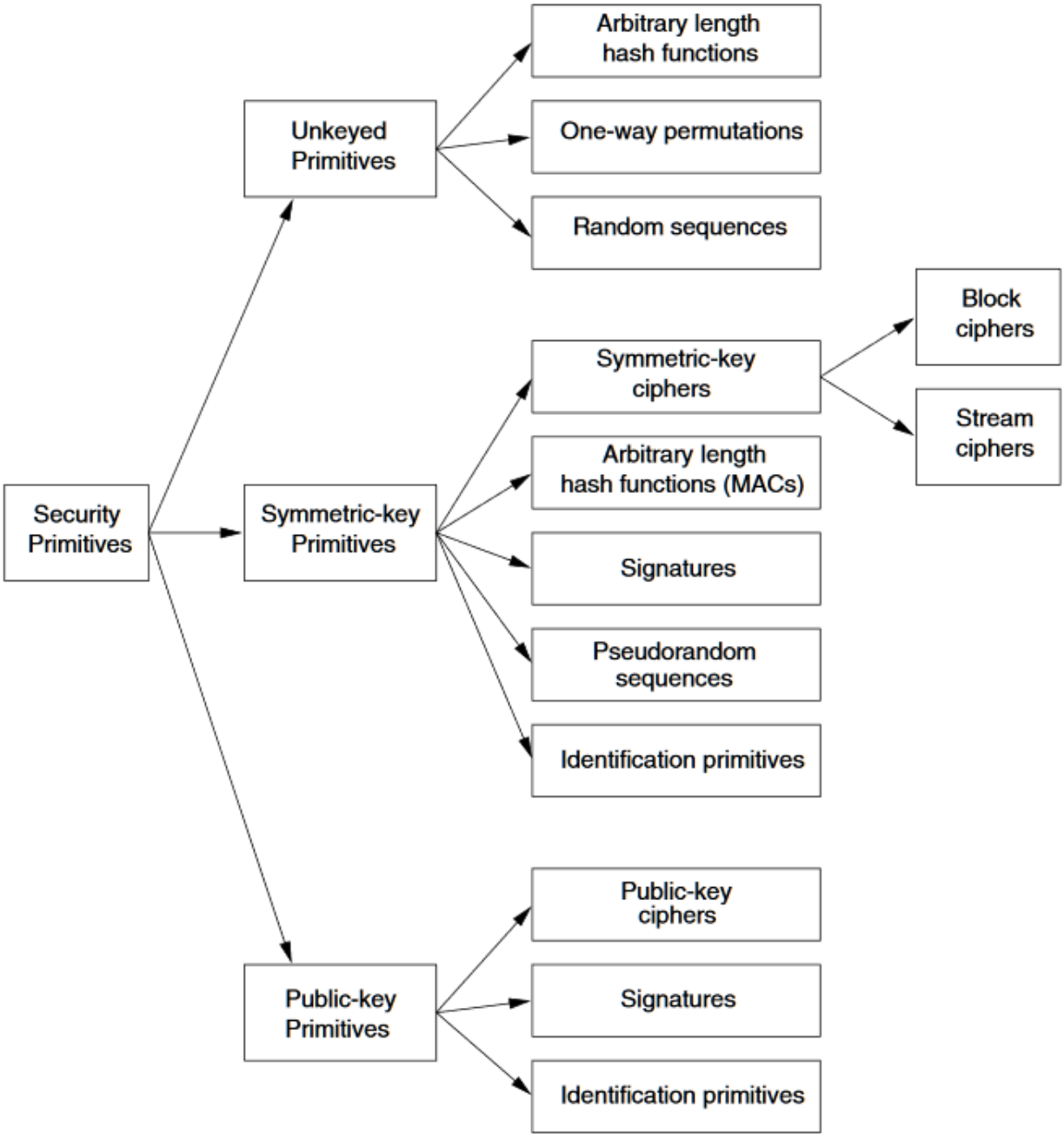

Figure 1.1: *A taxonomy of cryptographic primitives.*

Three steps

1. Specify threat model
2. Propose a construction
3. Prove that breaking construction under threat mode means solving an underlying hard problem

Remarks

- Encryption schemes should NOT be a secret
    - They must be scrutinized and vetted by profesionals
- Only the keys need to be secret
- Don’t roll your own crypto