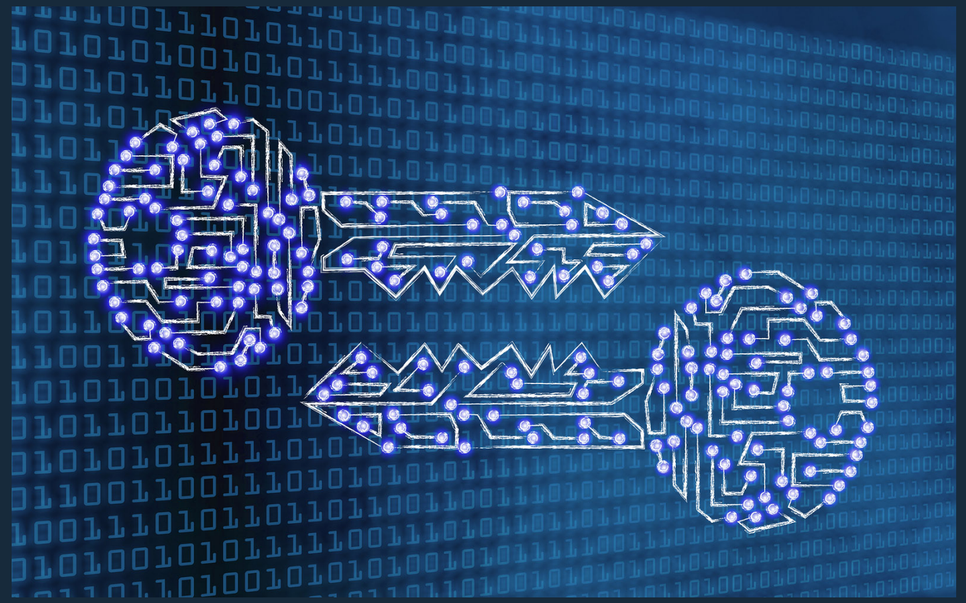
Kryptografie

Johanna Quednau - Julia Bremer

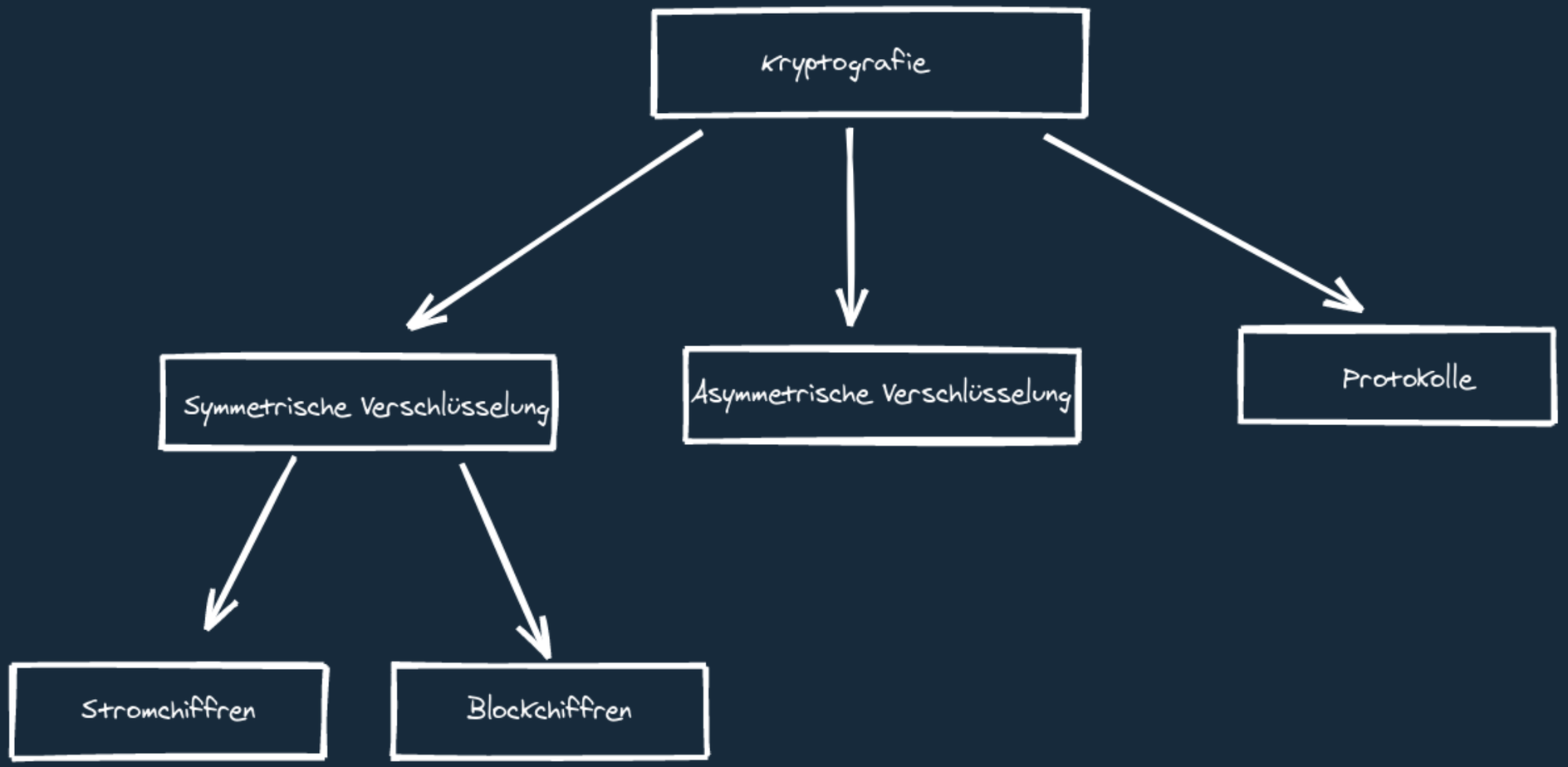
Agenda

- Kryptografie
- Obfuscation / Hashing / Encryption
- Block cipher / Stream cipher
- **Symmetrische** Verschlüsselung
- **Asymmetrische** Verschlüsselung
- Signaturen
- Zertifikate



Kryptografie

Die Wissenschaft der **Verschlüsselung**



Obfuscation

- **Security by Obscurity**
- Bei Obfuscation geht es nur darum Dinge **unverständlicher** zu machen
- Für Computer leicht drüber hinwegzusehen, für Menschen schwer
- Keine gute Sicherheitsmaßnahme

Hash

- Im Gegensatz zur Verschlüsselung eine mathematische **Einwegsfunktion**
  - Das heißt, man kann aus einem Hash **nicht** die originalen Daten rekonstruieren
- Verwendungszweck meist in Kombination mit Verschlüsselung
- **Rainbowtables** sind Tabellen, in denen die Hashes von häufigen Passwörtern eingetragen sind
- Damit diese Rainbowtables nicht funktionieren **saltet** man
- Beispiel: **MD5** und **SHA**

Salt

- Ein Extrawort, dass dem Passwort vor dem Hashing hinzugefügt wird.
- Dabei ist es nicht wichtig, dass der Salt geheim ist, nur dass er bei **jedem Benutzer unterschiedlich** ist.
- Durch die Zuführung eines Saltes, kann man den Hash nicht mehr in Rainbowtables nachsehen.
- Und gleiche Passwörter anhand der Gleichheit des Hashes erkennen.

Encryption

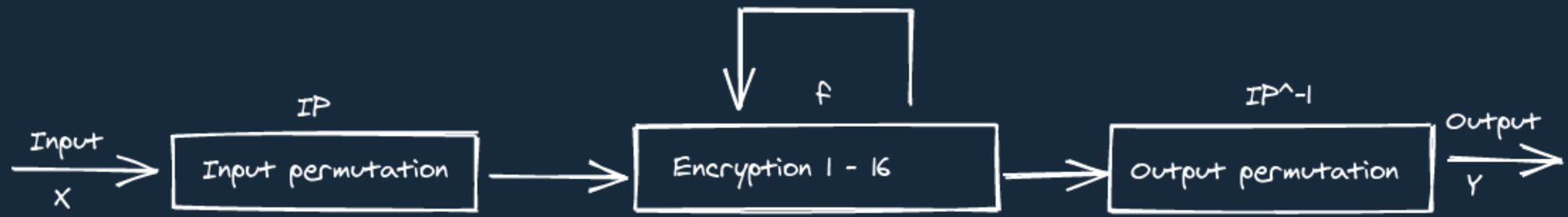
- Der eigentliche Fokus der Kryptografie
- Eine **umkehrbare** Funktion, bei der es eine Funktion **e**(ncryption) und eine Funktion **d**(ecryption) gibt.
- Zwei grundlegende Operationen um starke Verschlüsselung zu erreichen sind **Konfusion** und **Diffusion**
- **Konfusion**: Verschleierung von Zusammenhang von Schlüssel und Chiffre
- **Diffusion**: Verschleierung von statistischen Eigenschaften des Klartextes

Symmetrische Verschlüsselung

- In der symmetrischen Verschlüsselung kann mit **dem selben Schlüssel ver- und entschlüsselt** werden
- Beide Teilnehmer müssen im Besitz dieses Schlüssels sein
- Die Verteilung des Schlüssels ist ein Hauptproblem beim symmetrischen Verfahren
- Darum verwendet man auch **hybride** Verfahren, bei denen der Schlüsselaustausch asymmetrisch vollstreckt wird
- Man teilt die symmetrischen Verfahren in **Blockchiffren** und **Stromchiffren** auf

Stream vs Block Cipher

- Bei der Stromchiffre wird **jeder Klartextbit einzeln** verschlüsselt
  - Dafür wird **XOR** als einzige balancierter boolscher Operator verwendet
  - Beispiel: **RC4**
- Bei der Blockchiffre wird immer ein ganzer **Block Klartextbits gleichzeitg** verschlüsselt
  - Die statistischen Eigenschaften bleiben im Chiffrat erhalten
  - Die Algrithmen arbeiten iterativ
  - Beispiel: **DES** und **AES**



Betriebsmodi

- Meistens werden mehr als nur ein Block von 64/128 Bit verschlüsseln sondern **mehrere Blöcke**
- Wie diese Blöcke verknüpft werden entscheidet der **Betriebsmodus**
- Beispielsweise **ECB** Electronic-Codebook-Modus ist vollständig deterministisch.
- **CBC** Cipher-Block-Chaining-Modus verkettet die verschlüsselten Blöcke und umgeht dieses Phänomen so

$$y_1 = e_k(x_i \oplus IV)$$
$$y_i = e_k(x_i \oplus y_{i-1}), i \geq 2$$

Asymmetrische Verschlüsselung

Public-Key-Kryptografie

- Hier gibt es **zwei verschiedene** Schlüssel, einen **verschlüsselnden** und einen **entschlüsselnden**
- Diese Schlüssel werden auch **public key** und **private key** genannt
- Der public key **muss nicht geheim gehalten** werden
- Man nennt dies auch **Einwegsfunktionen** die im Grunde **injektive** Abbildungen sind
- Wird oft nur für **Signaturen** und zum **Schlüsselaustausch** verwendet

Asymmetrisch vs Symmetrische Verschlüsselung

- Hauptproblem der symmetrischen Verschlüsselung ist das **Schlüsselaustauschproblem**
- Nur mit asymmetrischen Keys kann **signiert** werden
- Asymmetrische Keys müssen **sehr lang** sein um ähnlich sicher wie symmetrische Keys zu sein
- Symmetrische Verschlüsselung ist sehr **schnell und effizient**
- Bei Asymmetrischer Verschlüsselung muss die **authentizität** der public keys gewährleistet werden

Signaturen

- Soll **Zurechenbarkeit** und **Nichtabstreitbarkeit** erreichen, ähnlich eine gewöhnlichen Signatur
- Kann nur durch asymmetrische Verschlüsselung zustande kommen
- Dabei geht es **nicht um Verschlüsselung** der Nachricht. Nachrichten können signiert sein, ohne verschlüsselt zu sein
- **RSA-Signatur** wird häufig verwendet und basiert auf **RSA-Verschlüsselung**

Zertifikate

- Ist ein Public Key und verweist darauf
- Wird von einer **CA - Certificate Authority** "beglaubigt", also einer **trusted third party**
- Vertrauen der CA ist erforderlich um Sicherheit herzustellen
- Der Standart ist **X.509v3** und wird auch für **HTTPS** verwendet
- **HTTPS** steht für **HTTP over SSL bzw. neuer TLS** und verschlüsselt Web-Nachrichten mithilfer solcher Zertifikate
- Bei **HTTP** hingegen werden Nachrichten im Klartext ausgetauscht

Gute Quellen

- Buch Kryptografie verständlich von Christof Paar
- Kapitel IT-Sicherheit aus "Betriebssysteme" von Tanenbaum
- Einführung in DES: https://www.youtube.com/watch?v=H7bvLU-2JUI

Danke für eure Aufmerksamkeit