INSTITUT FÜR
RECHTSINFORMATIK
UNIVERSITÄT DES SAARLANDES

CISPA
Center for IT-Security, Privacy and Accountability

Technische Grundlagen der Blockchain

Prof. Dr. Christoph Sorge

juris-Stiftungsprofessur
für Rechtsinformatik

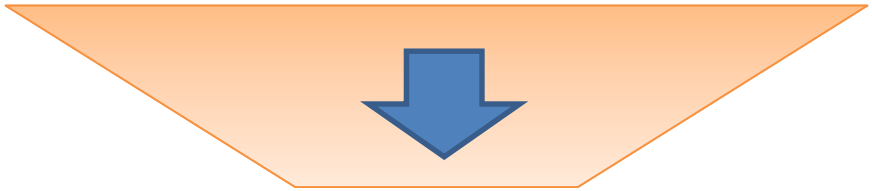
Hash-Funktionen

$$h : \Sigma^* \rightarrow \Sigma^n$$

Lorem ipsum dolor sit amet, consetetur sadipscing elitr, sed diam nonumy eirmod tempor invidunt ut labore et dolore magna aliquyam erat, sed diam voluptua. At vero eos et accusam et justo duo dolores et ea rebum. Stet clita kasd gubergren, no sea takimata sanctus est Lorem ipsum dolor sit amet. Lorem ipsum dolor sit amet, consetetur sadipscing elitr, sed diam nonumy eirmod tempor invidunt ut labore et dolore magna aliquyam erat, sed diam voluptua. At vero eos et accusam et justo duo dolores et ea rebum. Stet clita kasd gubergren, no sea takimata sanctus est Lorem ipsum dolor sit amet. Lorem ipsum dolor sit amet,, sed diam eirmod ut labore et dolore magna aliquyam erat, sed diam voluptua. At vero eos et accusam et duo dolores et ea. Stet clita kasd gubergren, no sea takimata sanctus est Lorem ipsum dolor sit amet.

da39a3ee5e6b4b0d3255

NoZXJuZCBhw59lbiBNw6R4Y2hlbnMZCBhw59lFLDvGJlbiwgSm9naHVydCB1bmQgUXV4Y2hlbnMgVsOZCBhw59l2R4Y2hlbnMgVsOZ2R4bnMgVsOVsIFLDhcms=

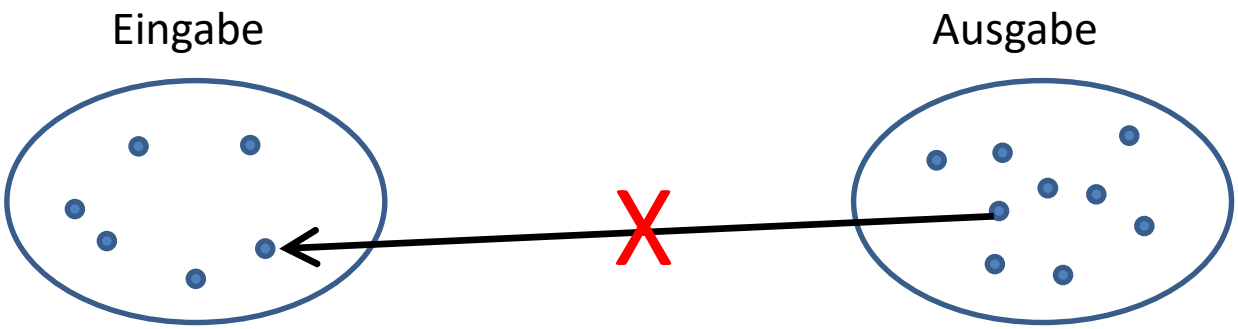


68ac906495480a3404be

Kryptographische Hash-Funktionen

- Grundidee Hash-Funktion:
  - Bilde Eingabe beliebiger Länge auf Ausgabe fester Länge ab (z.B. eine Textdatei auf 128 bit)
    - Funktion sollte effizient berechenbar sein

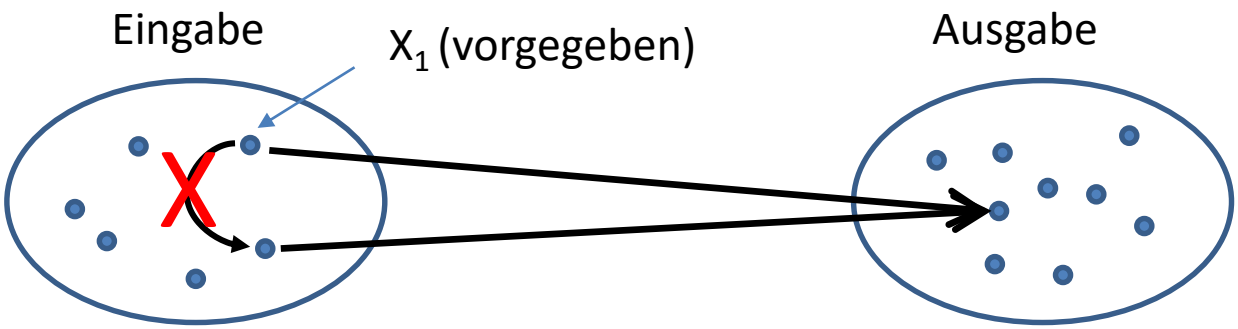
- Kryptographische Hash-Funktion h(x): Hash-Funktion mit folgenden Eigenschaften
  - Zu gegebenem h(x) kann kein passendes x effizient berechnet werden (Resistenz gg. Urbild-Angriff)



Kryptographische Hash-Funktionen

- Grundidee Hash-Funktion:
  - Bilde Eingabe beliebiger Länge auf Ausgabe fester Länge ab (z.B. eine Textdatei auf 128 bit)
    - Funktion sollte effizient berechenbar sein

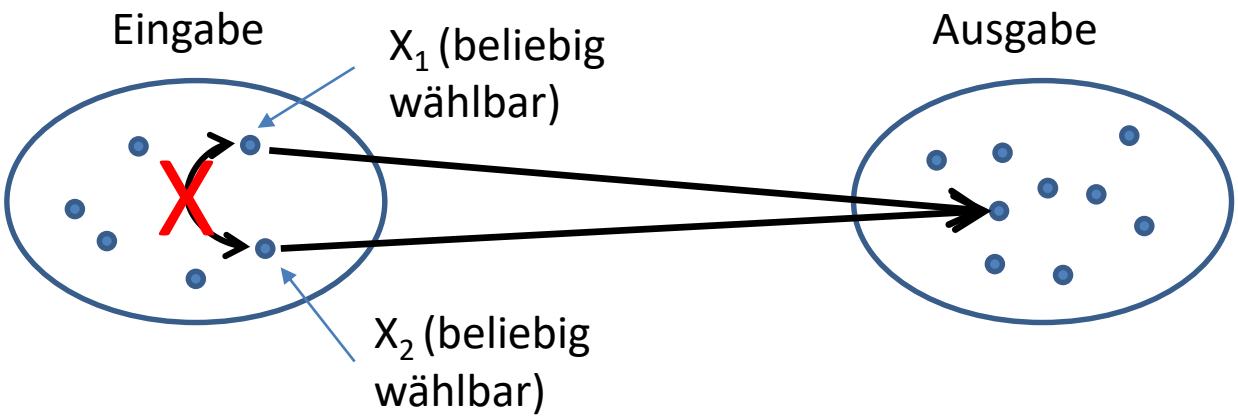
- Kryptographische Hash-Funktion h(x): Hash-Funktion mit folgenden Eigenschaften
  - Zu gegebenem x_1 kann kein x_2 effizient berechnet werden, so dass $h(x_1)=h(x_2)$ (Resistenz gg. Zweites-Urbild-Angriff)



Kryptographische Hash-Funktionen

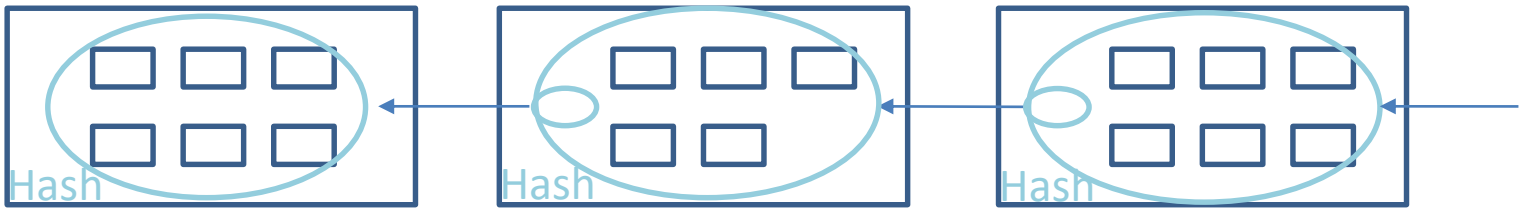
- Grundidee Hash-Funktion:
  - Bilde Eingabe beliebiger Länge auf Ausgabe fester Länge ab (z.B. eine Textdatei auf 128 bit)
    - Funktion sollte effizient berechenbar sein

- Kryptographische Hash-Funktion h(x): Hash-Funktion mit folgenden Eigenschaften
  - Es ist nicht effizient möglich, ein Paar (x_1,x_2) zu finden, so dass $h(x_1)=h(x_2)$



Die Blockchain

- Sammeln von Transaktionen (bzw. im Prinzip: beliebigen Daten) in Blöcken



- Jeder Block enthält Verweis auf vorherigen Block
  - Hashwert als Repräsentant aller Inhalte des vorherigen Blocks
- Ergebnis: Kette aus Blöcken
  - Jede Änderung in einem Block macht *alle* folgenden Blöcke ungültig

Sicherheit der Blockchain

- Eigenschaften kryptographischer Hashfunktionen garantieren:
  Nachträgliche Änderung eines Blocks erfordert auch Änderung aller Folgeblöcke

- Aber: Nachträgliche Änderung aller Folgeblöcke ist einfach

- Anwendung der Blockchain: Nachweis, dass Daten
  - in bestimmter Reihenfolge eingefügt wurden
  - oder zu einem bestimmten Zeitpunkt schon vorhanden waren

→ Es fehlt: Mechanismus, der Änderungen von Folgeblöcken schwierig macht

Änderungen erschweren

- Einfache Methode: Vertrauen in einzelne Instanzen
  - Beispiel: Hashwert des aktuellen Blocks wird von einer vertrauenswürdigen Instanz gespeichert – diese muss den Inhalt der Daten nicht kennen (nur den Hashwert)

- Alternative: Erzeugung neuer Blöcke schwierig machen. Hier: Bitcoin-Ansatz
  - Block wird nur durch kryptographischen Arbeitsbeweis gültig (rechen- und damit energieintensiv)
  - Aufwand für diesen Arbeitsbeweis von der Summe der Rechenleistung aller Teilnehmer abhängig; finanzieller Anreiz, Rechenleistung zu investieren
  - Blockchain mit allen Transaktionen öffentlich einsehbar
  - Ggf. gilt längste Blockchain als gültig → je älter ein Block, desto schwieriger zu fälschen

Anwendungen

- Grundprinzip der Blockchain nicht neu
  - Hash-Ketten: 1970er/80er Jahre, Vorschlag für Verwendung in Zeitstempel-Verfahren mindestens seit Anfang der 1990er
- Bitcoin: Geschickte Kombination mit Peer-to-Peer-System und kryptographischen Arbeitsbeweisen
  - „anonyme“, digitale „Währung“
  - Grundkonzept: Mitteilung aller Transaktionen an alle Teilnehmer, alle Daten öffentlich zugänglich
  - Vertrauen in die „Mehrheit der Rechenleistung“
  - Datenschutz/Anonymität kein Entwurfsziel, sondern erst in kryptographischen Erweiterungen enthalten

Anwendungen

- Bitcoin
  - Zeigt Robustheit des Grundkonzepts der Blockchain
  - Angreifbarkeit hauptsächlich auf Ebene des Peer-to-Peer-Netzes, der Bitcoin-Software und der Systeme, auf denen diese läuft
  - Ineffizientes System – nicht für große Transaktionsvolumina zu gebrauchen
    - Versuche der Verbesserung verletzen Prinzip der Dezentralität
- Weitere Anwendungen
  - Ablegen beliebiger Daten möglich, z.B. Smart Contracts
  - Grundproblem ebenfalls: Vertrauen in Echtheit der Transaktionen – Versprechen der Dezentralität nicht zu halten
  - Eigentliche Inhalte können, müssen aber nicht öffentlich gemacht werden

Fazit

- Blockchain als geschickte Technik, um Reihenfolgen von Transaktionen u.ä. festzulegen und Zeitstempel zu vereinfachen

- Kein Wundermittel – Vertrauen in die Echtheit muss immer noch etabliert werden
  - Zentrale Instanz(en)
  - oder Suche nach Surrogat für echt dezentrale Umsetzungen – z.B. Rechenleistung bei Bitcoin