Übersicht

Dienste, QoS

Sicherheit in Rechnernetzen (Security)

- Begriffe, Ziele und Maßnahmen (konzeptuell)
- Kryptographische Hashverfahren

Jonas Treumer <treumer@tu-freiberg.de>
Ben Lorenz <lorenzb@tu-freiberg.de>

Institut für Informatik

QoS – Quality of Service / Dienstegüte

Was versteht man unter dem Begriff Dienste?

Jonas Treumer <treumer@tu-freiberg.de>
Ben Lorenz <lorenzb@tu-freiberg.de>

Institut für Informatik

QoS – Quality of Service / Dienstegüte

Was versteht man unter dem Begriff Dienste?

Def: Ein Dienst ist eine Menge von Funktionen, die einem Nutzer von einem Erbringer zur Verfügung gestellt werden.

Jonas Treumer <treumer@tu-freiberg.de>
Ben Lorenz <lorenzb@tu-freiberg.de>

Institut für Informatik

QoS – Quality of Service / Dienstegüte

Was versteht man unter dem Begriff Dienste?

Def: Ein Dienst ist eine Menge von Funktionen, die einem Nutzer von einem Erbringer zur Verfügung gestellt werden.

Was versteht man unter dem Begriff Dienstegüte (Quality of Service)?

Jonas Treumer <treumer@tu-freiberg.de>
Ben Lorenz <lorenzb@tu-freiberg.de>

Institut für Informatik

QoS – Quality of Service / Dienstegüte

Was versteht man unter dem Begriff Dienste?

Def: Ein Dienst ist eine Menge von Funktionen, die einem Nutzer von einem Erbringer zur Verfügung gestellt werden.

Was versteht man unter dem Begriff Dienstegüte (Quality of Service)?

Def: Dienstegüte bezeichnet eine Menge quantitativer Kenngrößen, die die Eigenschaften eines Dienstes beschreiben.

Jonas Treumer <treumer@tu-freiberg.de>
Ben Lorenz <lorenzb@tu-freiberg.de>

Institut für Informatik

QoS – Quality of Service / Dienstegüte

Nennen und erklären Sie QoS-Parameter für Verkehrscharaktersistik und Zuverlässigkeit eines Dienstes.

- Verkehrscharakteristik:

Jonas Treumer <treumer@tu-freiberg.de>
Ben Lorenz <lorenzb@tu-freiberg.de>

Institut für Informatik

QoS – Quality of Service / Dienstegüte

Nennen und erklären Sie QoS-Parameter für Verkehrscharaktersistik und Zuverlässigkeit eines Dienstes.

- Verkehrscharakteristik:
  - Durchsatz (throughput)

Jonas Treumer <treumer@tu-freiberg.de>
Ben Lorenz <lorenzb@tu-freiberg.de>

Institut für Informatik

QoS – Quality of Service / Dienstegüte

Nennen und erklären Sie QoS-Parameter für Verkehrscharaktersistik und Zuverlässigkeit eines Dienstes.

- Verkehrscharakteristik:
  - Durchsatz (throughput)
    - Verhältnis der Größe einer PDU zu der Zeit bis die nächste PDU übertragen wird
    - Abhängig von Leitung, Puffer, Protokoll…

Jonas Treumer <treumer@tu-freiberg.de>
Ben Lorenz <lorenzb@tu-freiberg.de>
Institut für Informatik

QoS – Quality of Service / Dienstegüte

Nennen und erklären Sie QoS-Parameter für Verkehrscharaktersistik und Zuverlässigkeit eines Dienstes.

- Verkehrscharakteristik:
  - Durchsatz (throughput)
  - Verzögerung (delay)

Jonas Treumer <treumer@tu-freiberg.de>
Ben Lorenz <lorenzb@tu-freiberg.de>

Institut für Informatik

QoS – Quality of Service / Dienstegüte

Nennen und erklären Sie QoS-Parameter für Verkehrscharaktersistik und Zuverlässigkeit eines Dienstes.

- Verkehrscharakteristik:
  - Durchsatz (throughput)
  - Verzögerung (delay)
    - Zeit von der Übergabe einer PDU an den Dienst bis zur Ablieferung beim Epfänger (Ende-zu-Ende-Verzögerung)
    - Zeit, die eine PDU ‚im' Dienst verweilt

Jonas Treumer <treumer@tu-freiberg.de>
Ben Lorenz <lorenzb@tu-freiberg.de>

Institut für Informatik

QoS – Quality of Service / Dienstegüte

Nennen und erklären Sie QoS-Parameter für Verkehrscharaktersistik und Zuverlässigkeit eines Dienstes.

- Verkehrscharakteristik:
  - Durchsatz (throughput)
  - Verzögerung (delay)
  - Verzögerungsschwankung (jitter)

Jonas Treumer <treumer@tu-freiberg.de>
Ben Lorenz <lorenzb@tu-freiberg.de>

Institut für Informatik

QoS – Quality of Service / Dienstegüte

Nennen und erklären Sie QoS-Parameter für Verkehrscharaktersistik und Zuverlässigkeit eines Dienstes.

- Verkehrscharakteristik:
  - Durchsatz (throughput)
  - Verzögerung (delay)
  - Verzögerungsschwankung (jitter)
    - Varianz der Verzögerung
    - Pakete eines Paketstroms werden in (annähernd) gleichen Zeitabständen beim Sender abgeschickt, kommen aber in paketvermittelten Netzen selten mit identischer Verzögerung beim Empfänger an.

Jonas Treumer <treumer@tu-freiberg.de>
Ben Lorenz <lorenzb@tu-freiberg.de>
Institut für Informatik

QoS – Quality of Service / Dienstegüte

Nennen und erklären Sie QoS-Parameter für Verkehrscharaktersistik und Zuverlässigkeit eines Dienstes.

- Zuverlässigkeit:

Jonas Treumer <treumer@tu-freiberg.de>
Ben Lorenz <lorenzb@tu-freiberg.de>

Institut für Informatik

QoS – Quality of Service / Dienstegüte

Nennen und erklären Sie QoS-Parameter für Verkehrscharaktersistik und Zuverlässigkeit eines Dienstes.

- Zuverlässigkeit:
  - Unter Zuverlässigkeit eines Kommunikationsnetzes versteht man die Häufigkeit mit der bei einer Übertragung keine Fehler auftreten. Die Angabe der Zuverlässigkeit erfolgt für gewöhnlich in Prozent.
  - Beispiele für Fehler:

Jonas Treumer <treumer@tu-freiberg.de>
Ben Lorenz <lorenzb@tu-freiberg.de>
Institut für Informatik

QoS – Quality of Service / Dienstegüte

Nennen und erklären Sie QoS-Parameter für Verkehrscharaktersistik und Zuverlässigkeit eines Dienstes.

- Zuverlässigkeit:
  - Unter Zuverlässigkeit eines Kommunikationsnetzes versteht man die Häufigkeit mit der bei einer Übertragung keine Fehler auftreten. Die Angabe der Zuverlässigkeit erfolgt für gewöhnlich in Prozent.
  - Beispiele für Fehler:
    - Bitfehler
    - Burstfehler
    - Paketverlust

Jonas Treumer <treumer@tu-freiberg.de>
Ben Lorenz <lorenzb@tu-freiberg.de>
Institut für Informatik

QoS – Quality of Service / Dienstegüte

Nennen und erklären Sie QoS-Parameter für Verkehrscharaktersistik und Zuverlässigkeit eines Dienstes.

- Zuverlässigkeit:
  - Unter Zuverlässigkeit eines Kommunikationsnetzes versteht man die Häufigkeit mit der bei einer Übertragung keine Fehler auftreten. Die Angabe der Zuverlässigkeit erfolgt für gewöhnlich in Prozent.
  - Eine Verbindung zu n Empfängern ist k-zuverlässig (0 <= k <= n), falls für jede PDU gilt, dass sie bei mindestens k Empfängern ankommt.

Jonas Treumer <treumer@tu-freiberg.de>
Ben Lorenz <lorenzb@tu-freiberg.de>
Institut für Informatik

QoS – Quality of Service / Dienstegüte

Nennen Sie Dienstegüteklassen und erläutern Sie deren Semantik.

Jonas Treumer <treumer@tu-freiberg.de>
Ben Lorenz <lorenzb@tu-freiberg.de>

Institut für Informatik

QoS – Quality of Service / Dienstegüte

Nennen Sie Dienstegüteklassen und erläutern Sie deren Semantik.

- Deterministische DG
- Statistische DG
- Bestmögliche Dienstegüte (best effort)

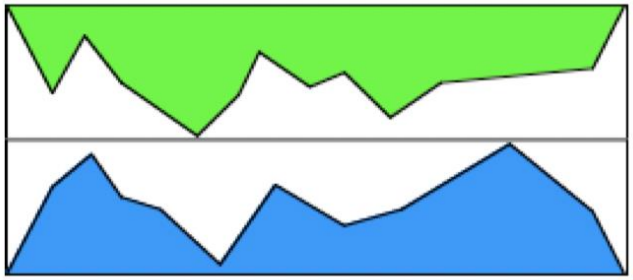
Jonas Treumer <treumer@tu-freiberg.de>
Ben Lorenz <lorenzb@tu-freiberg.de>

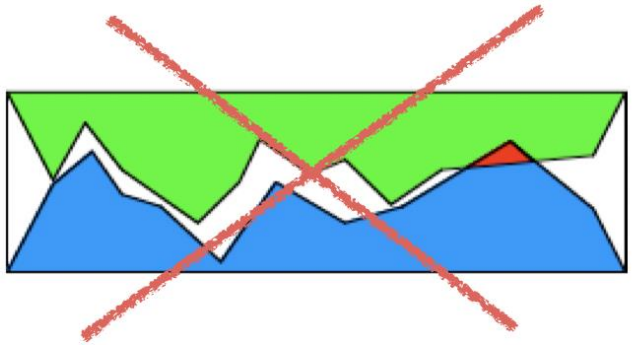
Institut für Informatik

QoS – Quality of Service / Dienstegüte

Nennen Sie Dienstegüteklassen und erläutern Sie deren Semantik.

- Deterministische DG
  - Vorgegebene Schranken der QoS-Parameter werden exakt eingehalten
  - Ressourcen stehen dem Dienstnutzer exklusiv zur Verfügung
  - Keine Konflikte möglich (außer „Besetztfall“)
  - Pessimistische Annahme des Systems



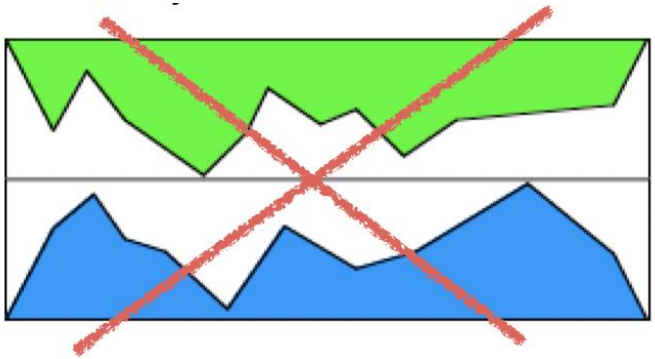


Jonas Treumer <treumer@tu-freiberg.de>
Ben Lorenz <lorenzb@tu-freiberg.de>
Institut für Informatik

QoS – Quality of Service / Dienstegüte

Nennen Sie Dienstegüteklassen und erläutern Sie deren Semantik.

- Statistische DG
  - Vorgegebene Schranken der QoS-Parameter müssen mit einer gewissen Wsk. eingehalten werden
  - zB.: Verzögerung muss für 90% der Pakete unter 100ms liegen
  - Ressourcen werden bis zu einem gewissen Grad überbelegt
  - Konflikte möglich!



Jonas Treumer <treumer@tu-freiberg.de>
Ben Lorenz <lorenzb@tu-freiberg.de>
Institut für Informatik

QoS – Quality of Service / Dienstegüte

Nennen Sie Dienstegüteklassen und erläutern Sie deren Semantik.

- best effort
  - Keinerlei Garantie für Dienstegüteparameter
  - Keine Ressourcenreservierung
  - Immer Konflikte!

Jonas Treumer <treumer@tu-freiberg.de>
Ben Lorenz <lorenzb@tu-freiberg.de>
Institut für Informatik

Security - Ziele

Vertraulichkeit

Jonas Treumer <treumer@tu-freiberg.de>
Ben Lorenz <lorenzb@tu-freiberg.de>

Institut für Informatik

Security - Ziele

Vertraulichkeit

- Speicherung der Daten
- Übertragung der Daten
- Lesen von Daten nur durch autorisierte Personen

Maßnahmen

Jonas Treumer <treumer@tu-freiberg.de>
Ben Lorenz <lorenzb@tu-freiberg.de>

Institut für Informatik

Security - Ziele

Vertraulichkeit

- Speicherung der Daten
- Übertragung der Daten
- Lesen von Daten nur durch autorisierte Personen

Maßnahmen

- (symmetrische/asymmetrische) Verschlüsselung von Daten

Jonas Treumer <treumer@tu-freiberg.de>
Ben Lorenz <lorenzb@tu-freiberg.de>

Institut für Informatik

Security - Ziele

Integrität

Jonas Treumer <treumer@tu-freiberg.de>
Ben Lorenz <lorenzb@tu-freiberg.de>

Institut für Informatik

Security - Ziele

Integrität

- keine unbemerkte Manipulation von Daten
- Änderungen an Daten müssen immer nachvollziehbar sein

Maßnahmen

Jonas Treumer <treumer@tu-freiberg.de>
Ben Lorenz <lorenzb@tu-freiberg.de>

Institut für Informatik

Security - Ziele

Integrität

- keine unbemerkte Manipulation von Daten
- Änderungen an Daten müssen immer nachvollziehbar sein

Maßnahmen

- Hashverfahren (u.a. Prüfsummen)

Jonas Treumer <treumer@tu-freiberg.de>
Ben Lorenz <lorenzb@tu-freiberg.de>

Institut für Informatik

Security - Ziele

Authentizität

Jonas Treumer <treumer@tu-freiberg.de>
Ben Lorenz <lorenzb@tu-freiberg.de>

Institut für Informatik

Security - Ziele

Authentizität

- Echtheit, Überprüfbarkeit
- Hier: Datenursprung, Sender

Jonas Treumer <treumer@tu-freiberg.de>
Ben Lorenz <lorenzb@tu-freiberg.de>

Institut für Informatik

Security - Ziele

Authentizität

- Echtheit, Überprüfbarkeit
- Hier: Datenursprung, Sender

Verbindlichkeit (non repudiation)

Jonas Treumer <treumer@tu-freiberg.de>
Ben Lorenz <lorenzb@tu-freiberg.de>

Institut für Informatik

Security - Ziele

Authentizität

- Echtheit, Überprüfbarkeit
- Hier: Datenursprung, Sender

Verbindlichkeit (non repudiation)

- „Unabstreitbarkeit“ einer Nachricht
- Bsp. Abschluss eines (Kauf)-Vertrages

Maßnahmen

Jonas Treumer <treumer@tu-freiberg.de>
Ben Lorenz <lorenzb@tu-freiberg.de>

Institut für Informatik

Security - Ziele

Authentizität

- Echtheit, Überprüfbarkeit
- Hier: Datenursprung, Sender

Verbindlichkeit (non repudiation)

- „Unabstreitbarkeit“ einer Nachricht
- Bsp. Abschluss eines (Kauf)-Vertrages

Maßnahmen

- digitale Signatur (z.B. Updates, Pakete aus Paketquellen)

Jonas Treumer <treumer@tu-freiberg.de>
Ben Lorenz <lorenzb@tu-freiberg.de>

Institut für Informatik

Krypt. Hashfunktion

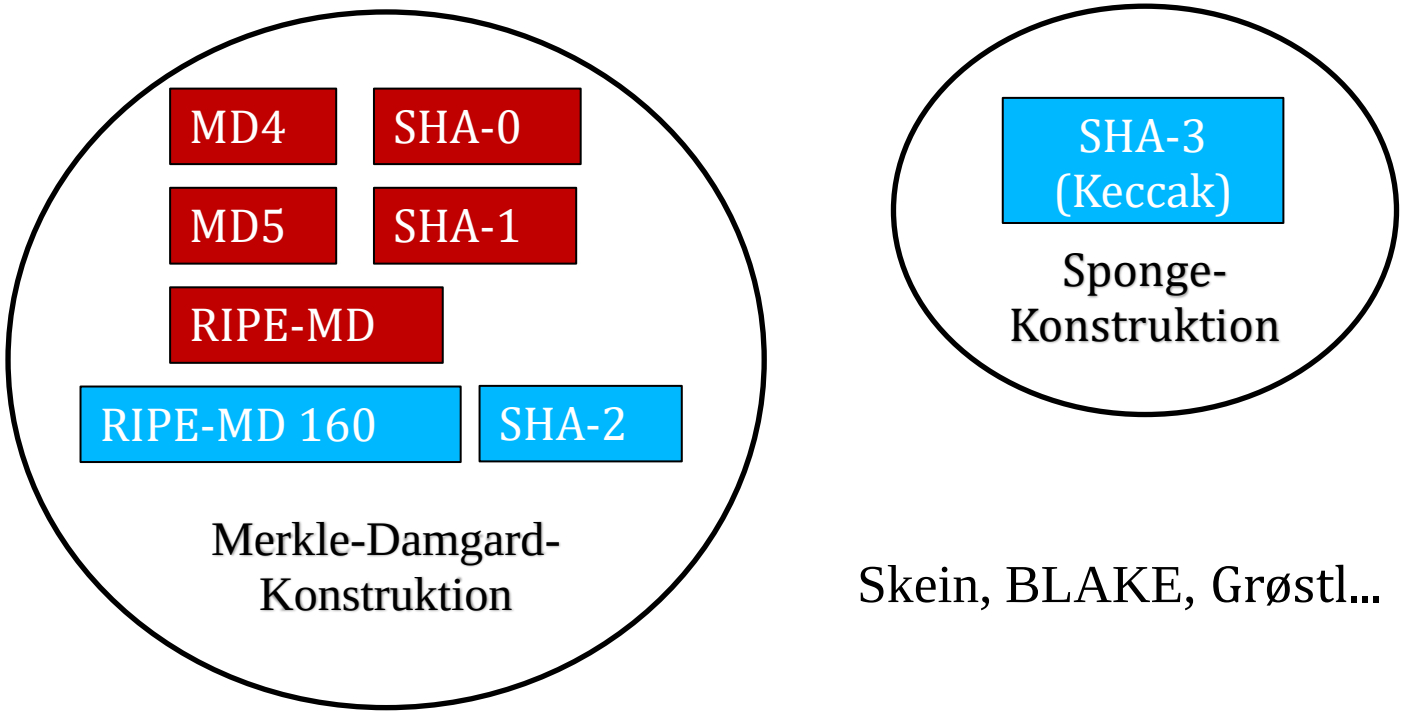
HASHEN != VERSCHLÜSSELN

Jonas Treumer <treumer@tu-freiberg.de>
Ben Lorenz <lorenzb@tu-freiberg.de>

Institut für Informatik

Krypt. Hashfunktion

HASHEN != VERSCHLÜSSELN



Skein, BLAKE, Grøstl...

Jonas Treumer <treumer@tu-freiberg.de>
Ben Lorenz <lorenzb@tu-freiberg.de>

Institut für Informatik

Krypt. Hashfunktion

$$h: \{0,1\}^m \rightarrow \{0,1\}^n \; mit \; m \geq n$$

Eigenschaften

Jonas Treumer <treumer@tu-freiberg.de>
Ben Lorenz <lorenzb@tu-freiberg.de>

Institut für Informatik

Krypt. Hashfunktion

$$h: \{0,1\}^m \rightarrow \{0,1\}^n \; mit \; m \geq n$$

Eigenschaften

- **Einwegfunktion** $y = f(x)$ „$einfach$“, $f^{-1}(y) = x$ „$schwer$“

Jonas Treumer <treumer@tu-freiberg.de>
Ben Lorenz <lorenzb@tu-freiberg.de>

Institut für Informatik

Krypt. Hashfunktion

$$h: \{0,1\}^m \rightarrow \{0,1\}^n \; mit \; m \geq n$$

Eigenschaften

- **Einwegfunktion** $y = f(x)$ „$einfach$“, $f^{-1}(y) = x$ „$schwer$“
- **nicht injektiv → Kollisionen möglich aber unerwünscht (Kollisionsresistenz)**

Jonas Treumer <treumer@tu-freiberg.de>
Ben Lorenz <lorenzb@tu-freiberg.de>
Institut für Informatik

Krypt. Hashfunktion

$$h: \{0,1\}^m \rightarrow \{0,1\}^n \; mit \; m \geq n$$

Eigenschaften

- **Einwegfunktion** $y = f(x)$ „$einfach$“, $f^{-1}(y) = x$ „$schwer$“
- **nicht injektiv → Kollisionen möglich aber unerwünscht (Kollisionsresistenz)**
  - **Schwache Kollisionsresistenz → finde kein x' zu x mit $h(x) = h(x')$**
  - **Starke Kollisionsresistenz → finde keine x, x' mit $h(x) = h(x')$**

Jonas Treumer <treumer@tu-freiberg.de>
Ben Lorenz <lorenzb@tu-freiberg.de>

Institut für Informatik

Krypt. Hashfunktion

$$h: \{0,1\}^m \rightarrow \{0,1\}^n \; mit \; m \geq n$$

Eigenschaften

- **Einwegfunktion** $y = f(x)$ „$einfach$“, $f^{-1}(y) = x$ „$schwer$“
- **nicht injektiv → Kollisionen möglich aber unerwünscht (Kollisionsresistenz)**
- **(wünschenswert) surjektiv** $\forall y \in Y \; \exists x \in X: f(x) = y$

Jonas Treumer <treumer@tu-freiberg.de>
Ben Lorenz <lorenzb@tu-freiberg.de>

Institut für Informatik

Krypt. Hashfunktion

$$h: \{0,1\}^m \rightarrow \{0,1\}^n \; mit \; m \geq n$$

Eigenschaften

- **Einwegfunktion** $y = f(x)$ „$einfach$“, $f^{-1}(y) = x$ „$schwer$“
- **nicht injektiv → Kollisionen möglich aber unerwünscht (Kollisionsresistenz)**
- **(wünschenswert) surjektiv** $\forall y \in Y \; \exists x \in X: f(x) = y$
- **Effizienz**

Jonas Treumer <treumer@tu-freiberg.de>
Ben Lorenz <lorenzb@tu-freiberg.de>

Institut für Informatik

Krypt. Hashfunktion

$$h: \{0,1\}^m \rightarrow \{0,1\}^n \; mit \; m \geq n$$

Eigenschaften

- **Einwegfunktion** $y = f(x)$ „$einfach$“, $f^{-1}(y) = x$ „$schwer$“
- **nicht injektiv → Kollisionen möglich aber unerwünscht (Kollisionsresistenz)**
- **(wünschenswert) surjektiv** $\forall y \in Y \; \exists x \in X: f(x) = y$
- **Effizienz**
- **Lawineneffekt/Chaoseffekt**

Jonas Treumer <treumer@tu-freiberg.de>
Ben Lorenz <lorenzb@tu-freiberg.de>

Institut für Informatik

Krypt. Hashfunktion

Geburtstagsparadox

Jonas Treumer <treumer@tu-freiberg.de>
Ben Lorenz <lorenzb@tu-freiberg.de>

Institut für Informatik

Krypt. Hashfunktion

Geburtstagsparadox

$\boldsymbol{p(n)}$ → Wahrscheinlichkeit dafür, dass mind. 2 Personen am gleichen Tag Geburtstag haben.

$\mathbf{q(n)}$ → Wahrscheinlichkeit dafür, dass mind. 2 Personen an einem bestimmten gleichen Tag Geburtstag haben.

Wie muss groß muss n sein,

damit $p, q > 0.5$?

Jonas Treumer <treumer@tu-freiberg.de>
Ben Lorenz <lorenzb@tu-freiberg.de>

Institut für Informatik

Krypt. Hashfunktion

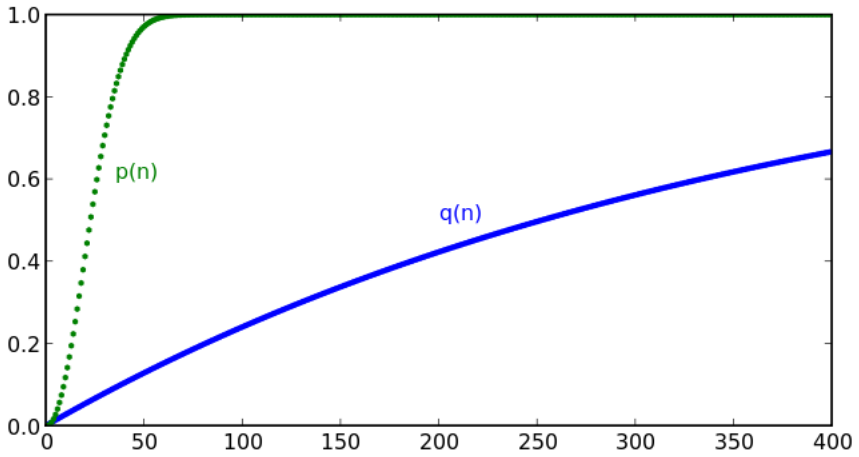
Geburtstagsparadox

$p(n)$ → Wahrscheinlichkeit dafür, dass mind. 2 Personen am gleichen Tag Geburtstag haben.

$q(n)$ → Wahrscheinlichkeit dafür, dass mind. 2 Personen an einem bestimmten gleichen Tag Geburtstag haben.

Wie muss groß muss n sein,

damit $p, q > 0.5$?



Jonas Treumer <treumer@tu-freiberg.de>
Ben Lorenz <lorenzb@tu-freiberg.de>

Institut für Informatik

Krypt. Hashfunktion

Geburtstagsparadox

$p(n)$ → Wahrscheinlichkeit dafür, dass mind. 2 Personen am gleichen Tag Geburtstag haben.

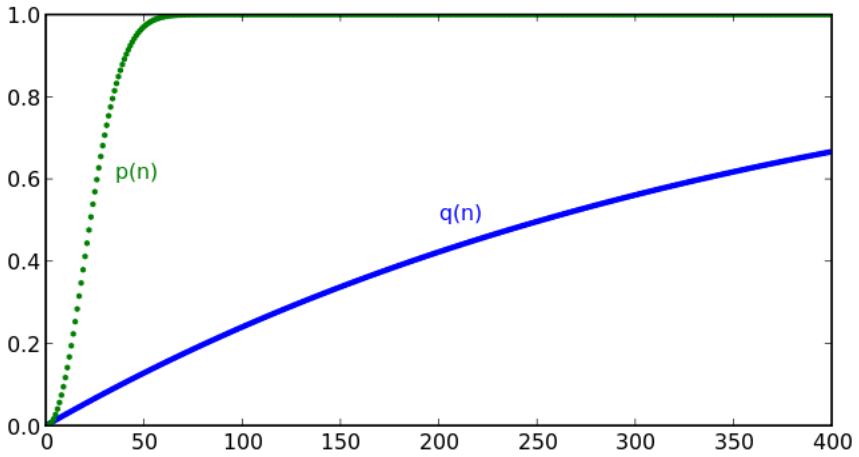
n = 23

$q(n)$ → Wahrscheinlichkeit dafür, dass mind. 2 Personen an einem bestimmten gleichen Tag Geburtstag haben.

n = 253

Wie muss groß muss n sein,

damit $p, q > 0.5$?



Jonas Treumer <treumer@tu-freiberg.de>
Ben Lorenz <lorenzb@tu-freiberg.de>

Institut für Informatik

Krypt. Hashfunktion

Verwendung

- **(nicht krypto.) Hashmaps/Hashtabellen**
- **(nicht zwangsläufig krypto.) Prüfsummen**
- **Signieren von Nachrichten, Nachrichten-Headern**
- **Integritätsprüfung**
- **Passwörter hashen**
- **Pseudozufallsgeneratoren**

Jonas Treumer <treumer@tu-freiberg.de>
Ben Lorenz <lorenzb@tu-freiberg.de>

Institut für Informatik

Krypt. Hashfunktion

Programmieraufgabe:

Untersuchen Sie mit Hilfe von C oder C++ das MD5-Hashverfahren hinsichtlich der **schwachen** und **starken** Kollisionsresistenz.

Beschränken Sie sich dabei auf die ersten 4-8 Zeichen des Hashes.

Idee schwache Kollisionsresistenz:

Erstellen Sie einen Target-String (random oder per Eingabe). Generieren Sie einen Target-Hash zu diesem String. Erzeugen Sie anschließend solange zufällige Zeichenketten und deren Hash, bis ein Hash mit dem Target-Hash übereinstimmt.

Idee schwache Kollisionsresistenz:

Erzeugen Sie solange zufällige Zeichenketten und deren Hash, bis ein Hash mit einem bereits erzeugten Hash übereinstimmt. Verwenden Sie eine möglichst zugriffseffiziente Datenstruktur, um die bekannten Hashes zu speichern und auf Kollision zu prüfen. Welche Datenstruktur könnte sich hier anbieten?

Jonas Treumer <treumer@tu-freiberg.de>
Ben Lorenz <lorenzb@tu-freiberg.de>
Institut für Informatik