Sichere Hashfunktionen

Prof. Dr.-Ing. Damian Weber

Hochschule für Technik und Wirtschaft des Saarlandes

Was tut eine Hashfunktion?

Hashfunktionen (Definition)

$$h: \; U \longrightarrow V$$

U = Universum, V = Hashwerte, $|V| < \infty$

Urbilder

Bilder

Hashfunktionen (Beispiel)

EAN (ISBN-13) 978047111709-4

- $U = \{ n \in \mathbf{N} \mid n<10^{12} \}$
- V={0,1,2,3,4,5,6,7,8,9} .... 10 Kisten

9+3·7+8+3·0+4+3·7+1+3·1+1+3·7+0+3·9
=1116

10-6=4 978047111709 → 4

Hashfunktionen (Anwendung)

- Integritätschecks (Software, Dokumente)
- Digitale Signaturen
- Erzeugen kryptographischer Schlüssel

  aus Passwörtern
- Identifizieren von Malware

Integritätscheck

Gleiche Kiste, also Dokument gleich
(ungleiche mit 75% Chance in anderer Kiste)

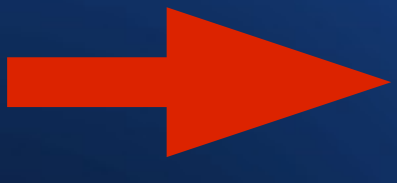
Datenstrukturen: 10^8 Kisten
Krypto: $2^{256} \approx 10^{80}$ Kisten

Tools: sha1, sha256 (Linux sha1sum, sha256sum), Hashes of tar.gz

Digitale Signaturen

gnupg, SSL/TLS, ssh = Message authentication (Echtheit)

Digitale Signaturen
Gefahr: 2 Dokumente, gleiche Kiste



Kollision

Digitale Signaturen

Gefahr: 2. Dokument für diese Kiste künstlich erzeugen

Hashfunktionen (Anforderungen)

Sicher, wenn „unmöglich“:

- Urbild **(preimage-resistance)**
  (Anwendung: finde gültiges Passwort)
- 2. Urbild **(2nd preimage-resistance)**
  (Anwendung: finde 2. Dokument für gegebene Signatur)
- Kollision **(collision resistance)**
  (Anwendung: finde 2 Dokumente mit gleicher Signatur)

Hashfunktionen (Urbild)

Wir knacken die Urbildeigenschaft der EAN:

finde eine EAN mit Prüfziffer 7

a+3b+c+3d+e+3f+g+3h+i+3j+k+3l

muss als Endziffer 7 haben

Hashfunktionen (2. Urbild)

Wir knacken die 2. Urbildeigenschaft der EAN:

Kenne EAN mit Prüfziffer 4:

978047111709-4

Finde noch eine mit Prüfziffer 4
a+3b+c+3d+e+3f+g+3h+i+3j+k+3l

a und c tauschen ändert nichts am Ergebnis:
879047111709-4

Hashfunktionen (Kollision)

Kollision finden leicht, wenn man 2. Urbild kann

Wichtige Attacke: viele Hashes zufällig erzeugen

prüfen, ob Hashwert schon einmal gesehen

```
978-0312979478 And then there were none
978-0062073563 Murder of Roger Ackroyd
978-0062073495 Murder on the Orient Express
978-0573619236 The Mousetrap
978-0002315968 Miss Marple Final Cases
```

Kollision mit Hashwert 8 gefunden

Hashfunktionen (Anforderungen)

Effizienz

- Anwendungen mit vielen Signaturen
- Signieren von großen Dokumenten
- Signieren von Software

Sicherheit gegen
- Urbildsuche
- Kollisionen

Problematik Urbildsuche

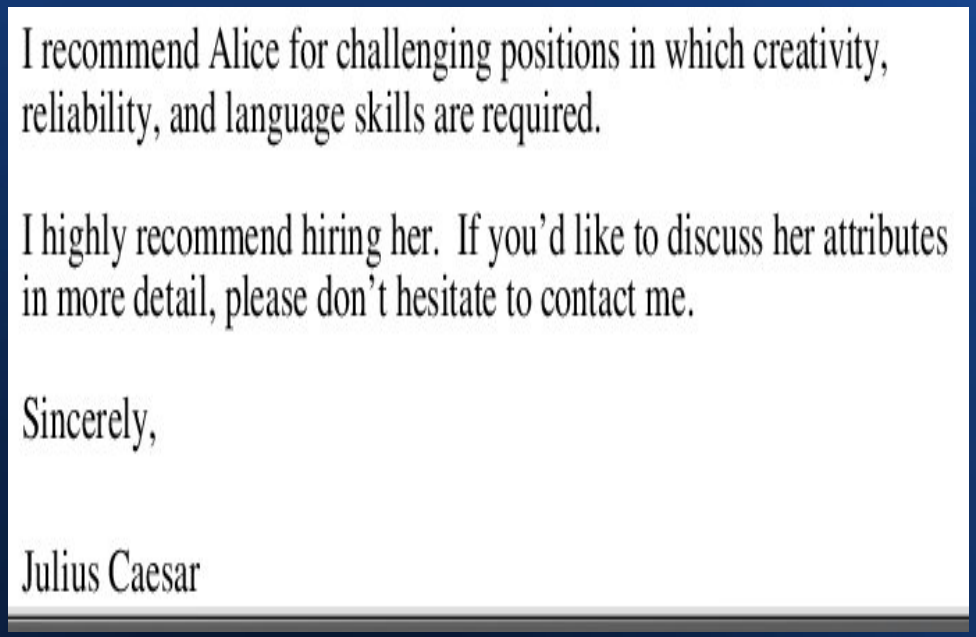
- MD5-Hashwerte von Passwörtern in `/etc/passwd`
- `1234` `81dc9bdb52d04dc20036dbd8313ed055`
- `hallo` `598d4c200461b81522a3328565c25f7c`
- `secret` `5ebe2294ecd0e0f08eab7690d2a6ee69`

- wer aus `5ebe2294ecd0e0f08eab7690d2a6ee69`
  den String „secret“ errechnen kann, kann Passwörter knacken

MD5 ist bei Kollisionen gebrochen, aber nicht bei Urbildsuche

Problematik Kollision

- zwei Dokumente mit gleichem Hashwert

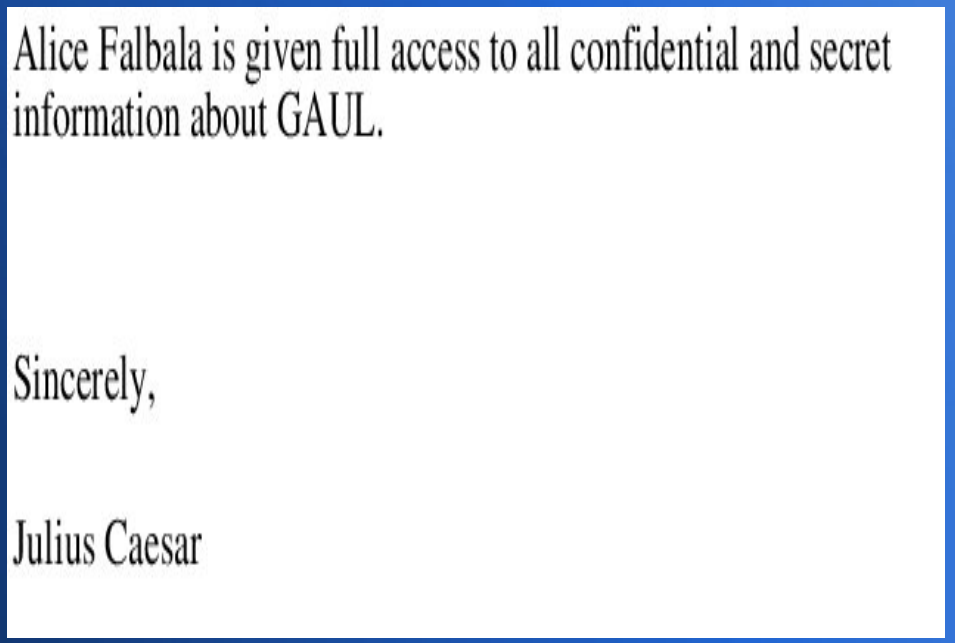


I recommend Alice for challenging positions in which creativity, reliability, and language skills are required.

I highly recommend hiring her. If you'd like to discuss her attributes in more detail, please don't hesitate to contact me.

Sincerely,

Julius Caesar



Alice Falbala is given full access to all confidential and secret information about GAUL.

Sincerely,

Julius Caesar

`MD5: a25f7f0b29ee0b3968c860738533a4b9`

`MD5: a25f7f0b29ee0b3968c860738533a4b9`

(Beispiel von Stefan Lucks, Bauhaus-Univ. Weimar)

Kollision: Mindestanforderung

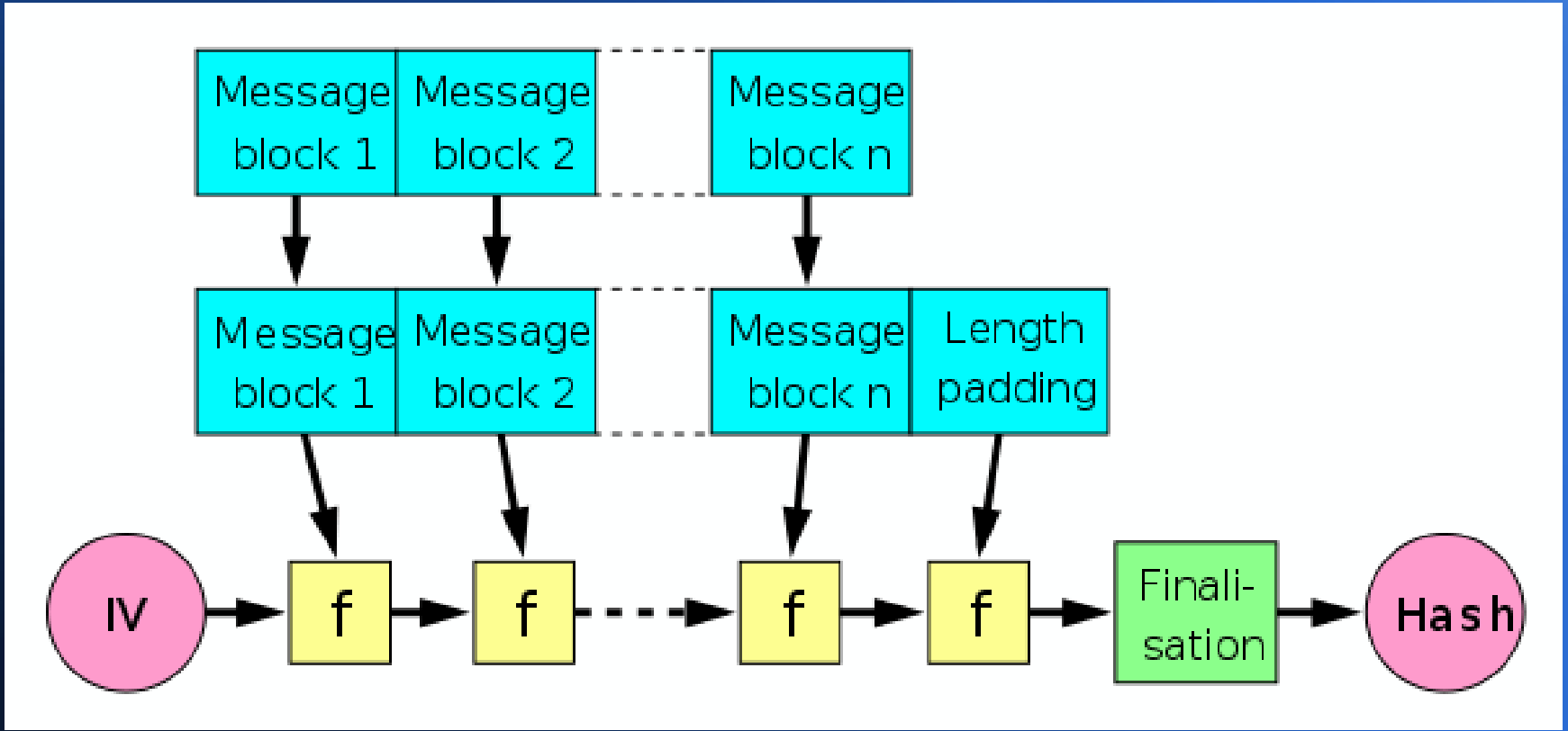
- robust gegen Pollard-λ Attacke mit O(sqrt(n)) Laufzeit
  - Bildbereich der Hashfunktion > 2^{160}
  - daher neue Hashfunktionen mit 256 Bits

Sichere Hashfunktionen

aktuelle Situation:

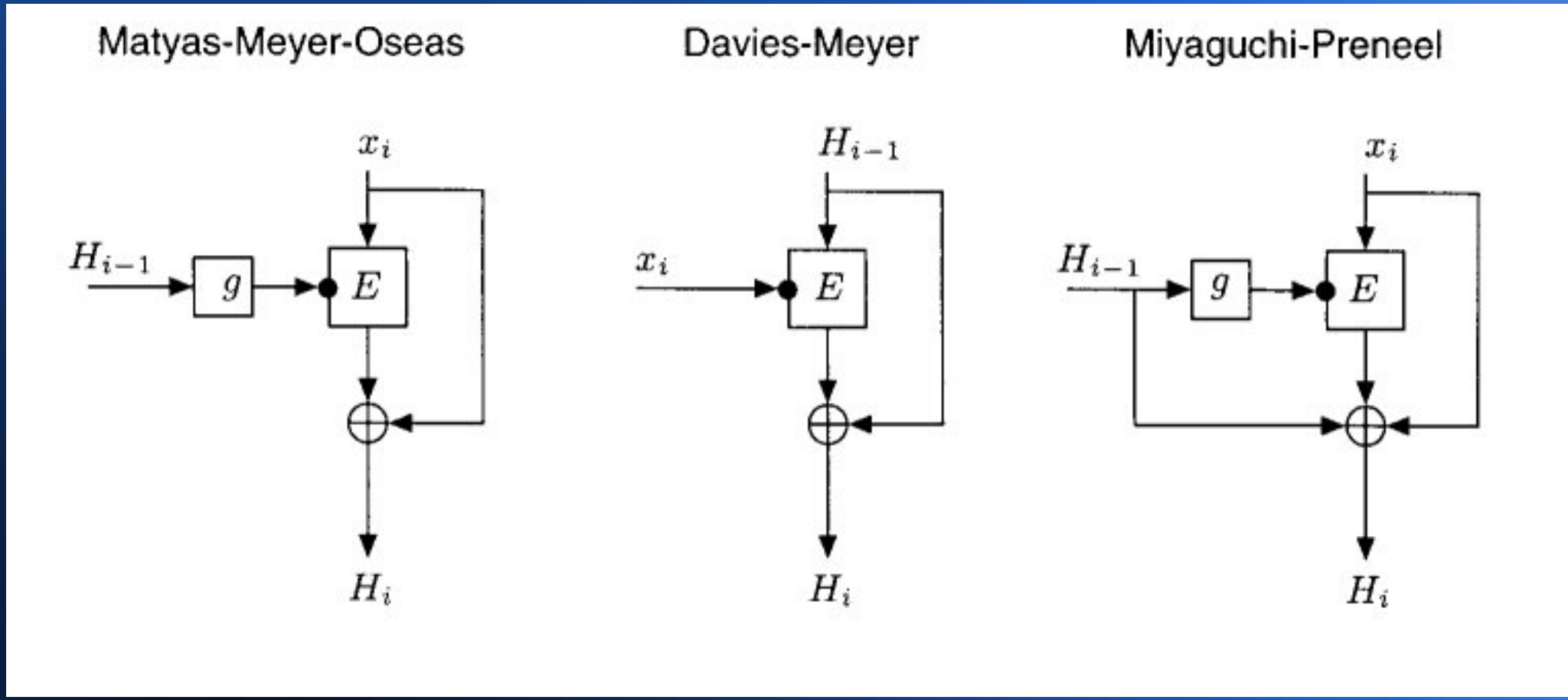
- schwache Algorithmen
  MD5 (Rivest 1991), Kollisionen in Minuten
  SHA-1 (NIST 1994) (Kollisionen nach 2^{70} Op.)
- Interimslösung SHA-256 (NIST 2002)
- SHA-3 Wettbewerb von NIST 2012

Merkle-Damgard-Construction



https://en.wikipedia.org/wiki/Image:Merkle-damgard.png

Merkle-Damgard-Construction



Handbook of Applied Cryptography (Menezes et al)

Hashwettbewerb NIST

SHA-3 Hashwettbewerb

Runde 1: Nov 2008
Runde 2: Juli 2009
Runde 3: Dez 2010

5 Finalisten
`Blake, Grøstl, JH, Keccak, Skein`

Auswahl Okt. 2012 **Keccak**

Empfehlungen Hashfunktionen:
Wertebereich mindestens 256 Bit