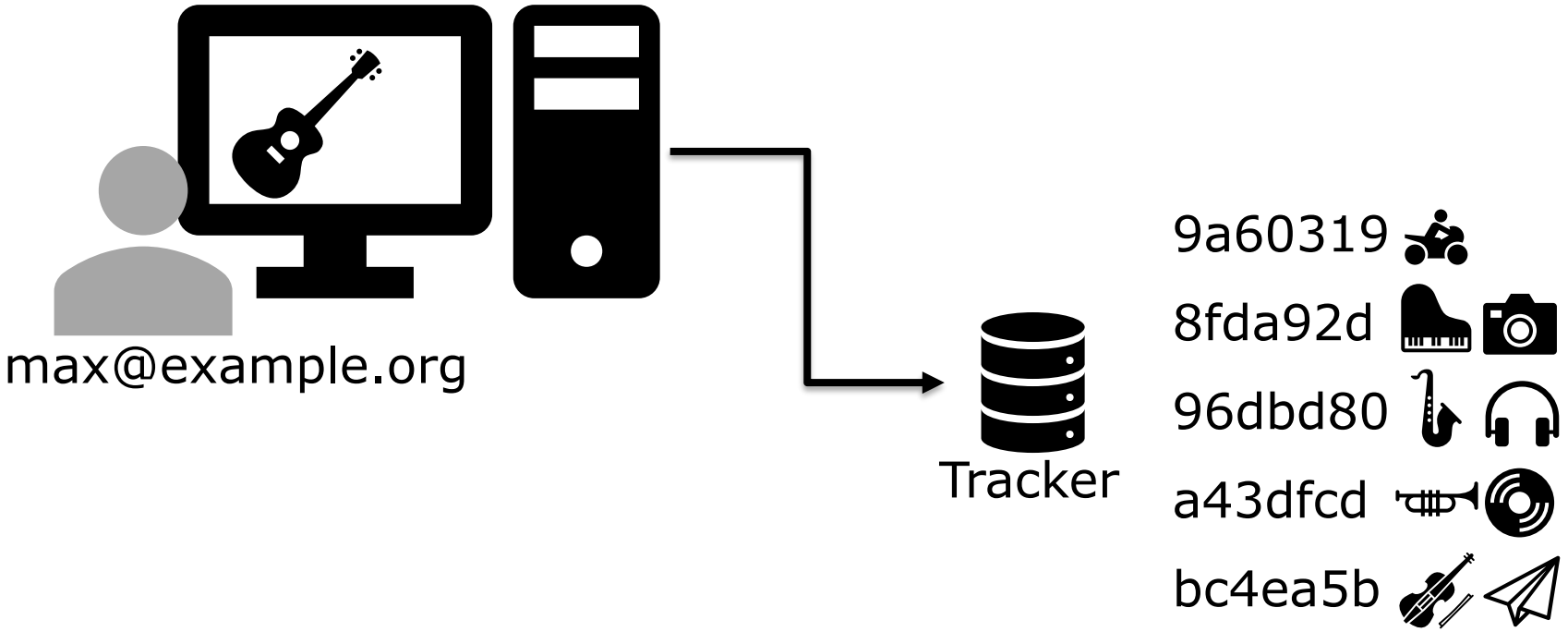
Universität Hamburg
DER FORSCHUNG | DER LEHRE | DER BILDUNG

25. April 2018

Hashing of personally identifiable information is not sufficient

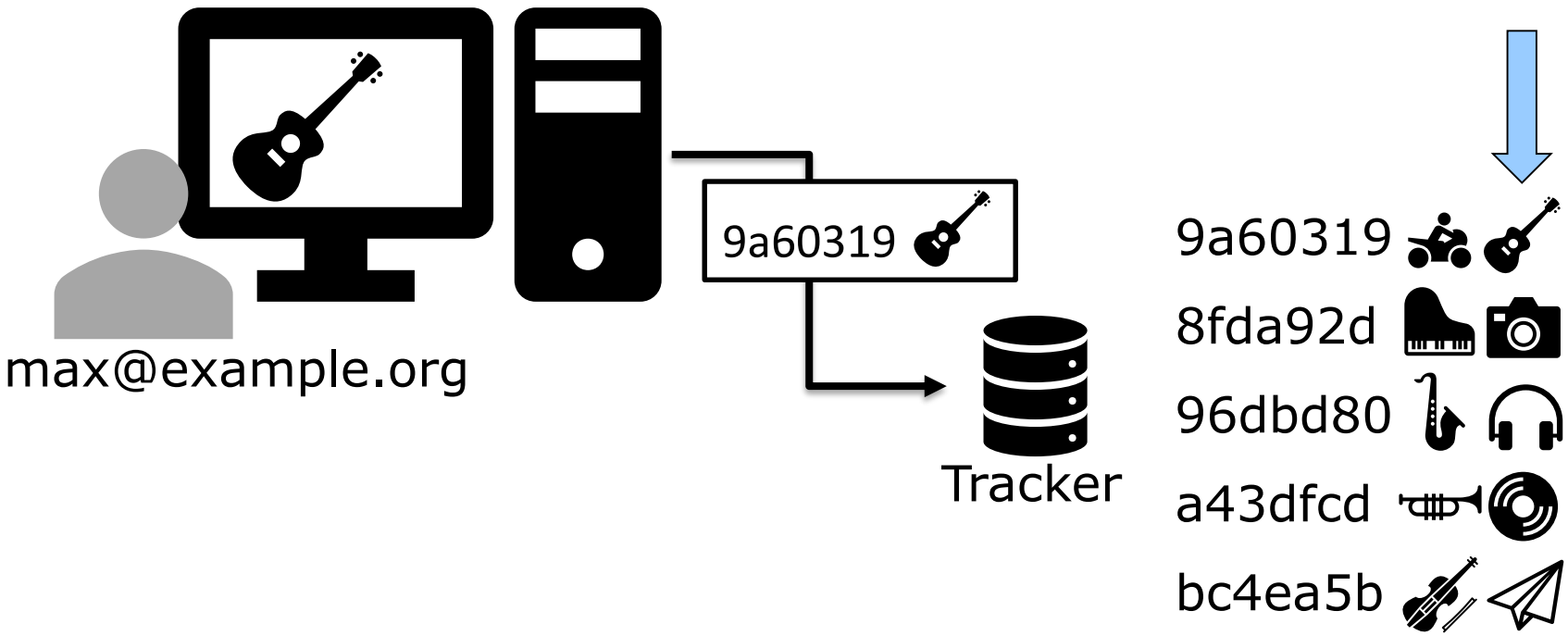
Matthias Marx, Ephraim Zimmer, Tobias Mueller, Maximilian Blochberger, Hannes Federrath

Motivation



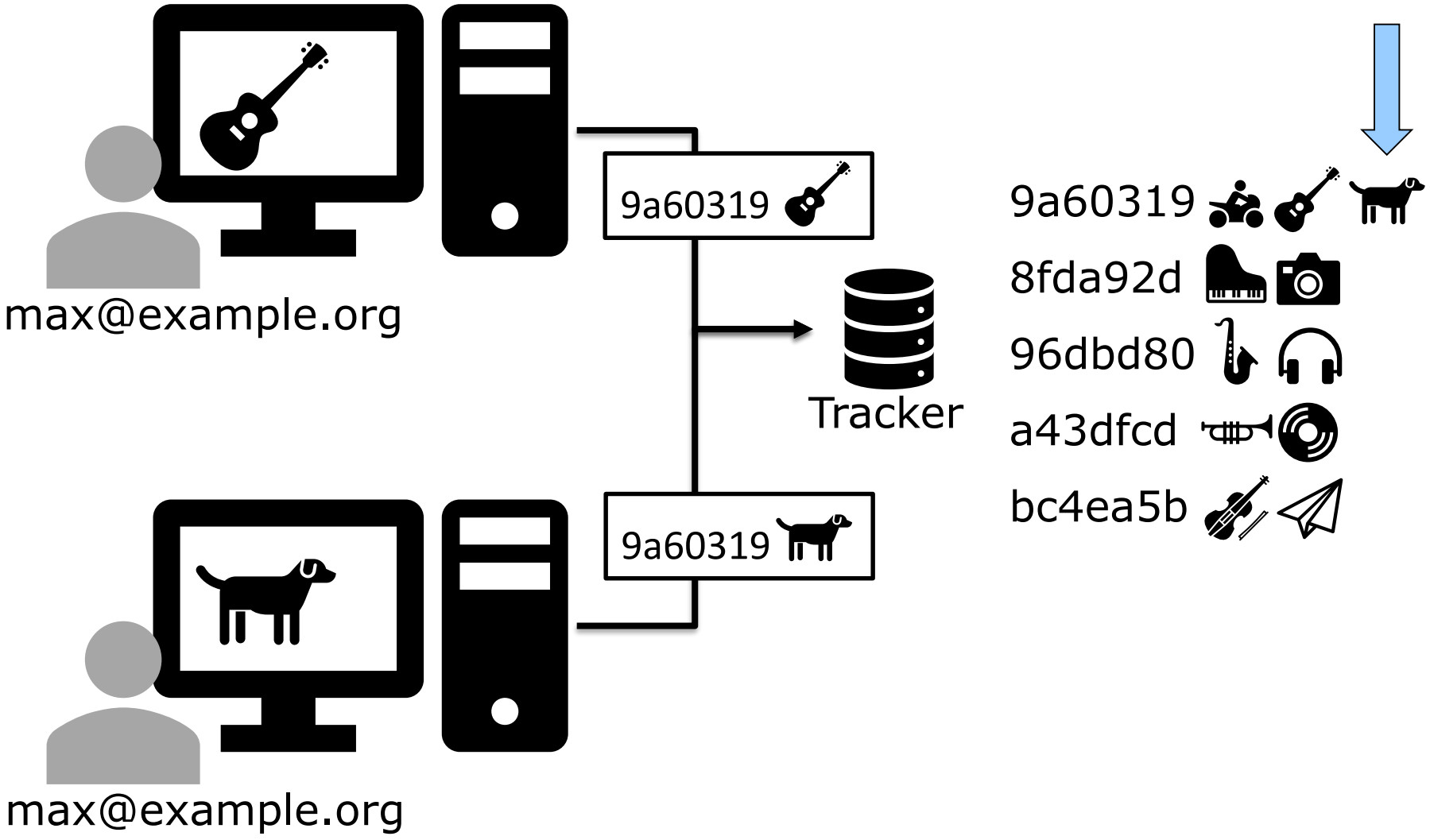
- Cookies sind weniger zuverlässig als andere Tracking-Methoden
- E-Mail-Adressen werden selten gewechselt
  - Eignen sich nicht direkt als Identifier
- Kryptographische Hashfunktionen
  - hash(max@example.org) = 9a60319

Motivation

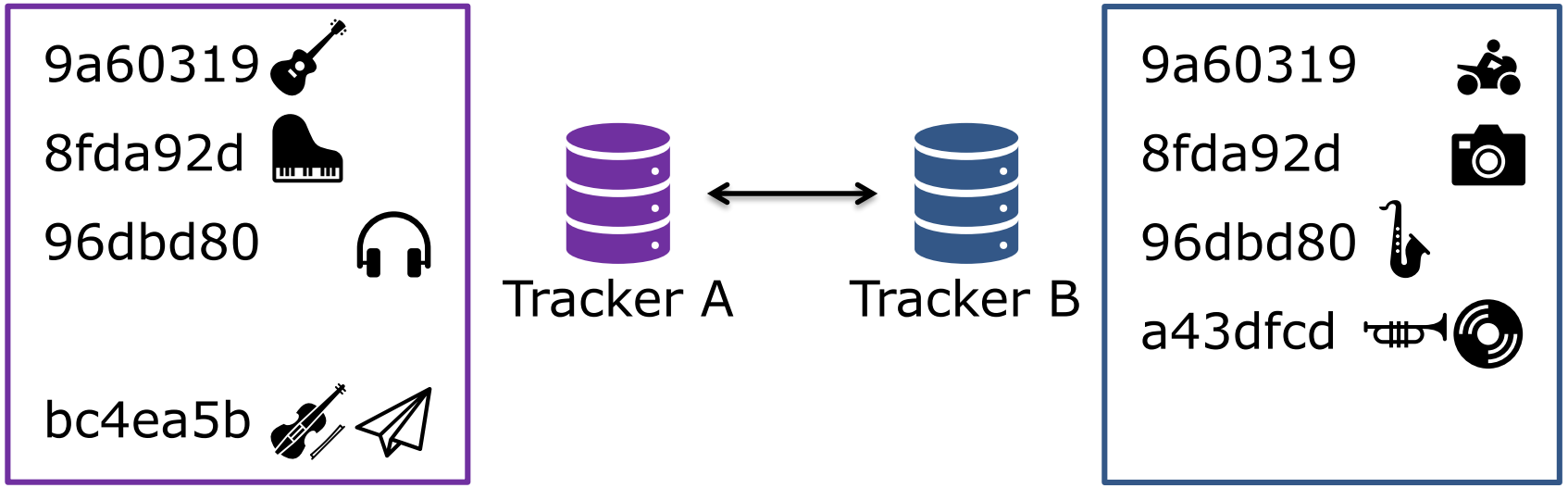


- Cookies sind weniger zuverlässig als andere Tracking-Methoden
- E-Mail-Adressen werden selten gewechselt
  - Eignen sich nicht direkt als Identifier
- Kryptographische Hashfunktionen
  - hash(max@example.org) = 9a60319

Motivation


max@example.org
9a60319
Tracker
9a60319
max@example.org
9a60319
8fda92d
96dbd80
a43dfcd
bc4ea5b

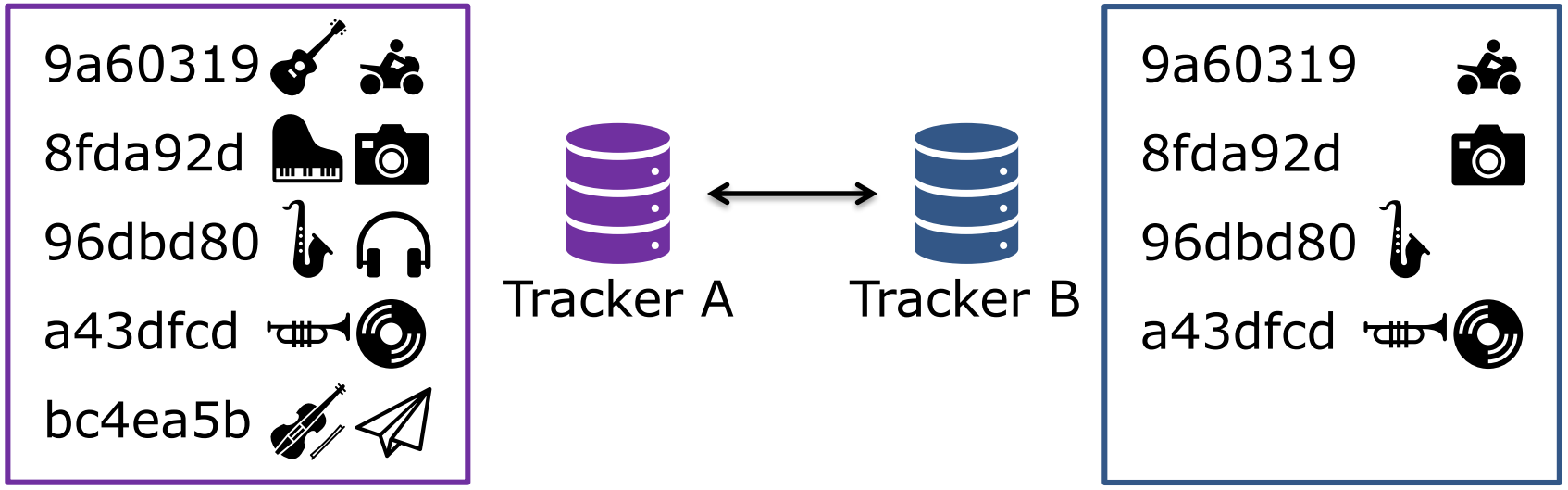
Motivation



- Tracker können verschiedene Datensätze zusammenführen
- Welche Folgen hat es, wenn die Hashfunktion keine Einwegfunktion ist?

  9a60319→max@example.org

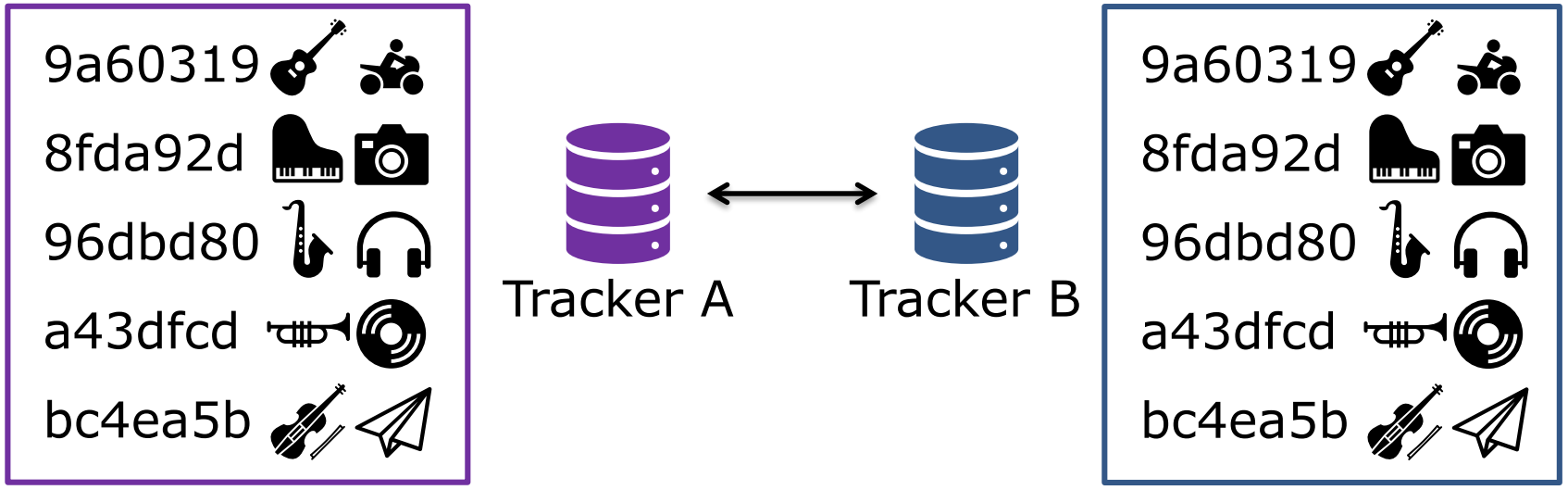
Motivation



- Tracker können verschiedene Datensätze zusammenführen
- Welche Folgen hat es, wenn die Hashfunktion keine Einwegfunktion ist?

  9a60319→max@example.org

Motivation



- Tracker können verschiedene Datensätze zusammenführen
- Welche Folgen hat es, wenn die Hashfunktion keine Einwegfunktion ist?

  9a60319→max@example.org

Verwandte Arbeiten

- Wiederherstellen von gehashten Passwörtern
  - Unter Berücksichtigung der Struktur von Passwörtern

    [Narayanan und Shmatikov, 2005; Bonneau, 2012]

- Social Security Numbers werden durch SHA-1 nicht ausreichend geschützt

  [Felten, 2012]

- Angriffe auf gehashte E-Mail- und MAC-Adressen sind theoretisch möglich

  [Demir et al., 2017]

Relevanz

*There are several aspects of privacy that are built into our platform, such as the **use of hash algorithms (SHA-1) to anonymize MAC addresses** safely when it comes to tracking and analysis purposes.*

[infsoft GmbH, 2018]

*Another acceptable option is to pass to Google Analytics an **encrypted identifier that is based on PII** that is not Protected Health Information, as long as you use the proper encryption level. Google has a **minimum hashing requirement of SHA256** and strongly recommends the use of a salt, minimum 8 characters.*

[Google, 2017]

***MD5 and SHA-256 hashed email addresses or phone numbers** that have been automatically generated from raw personally identifiable information (PII) using BlueKai code*

[Oracle, 2018]

Vorgehen

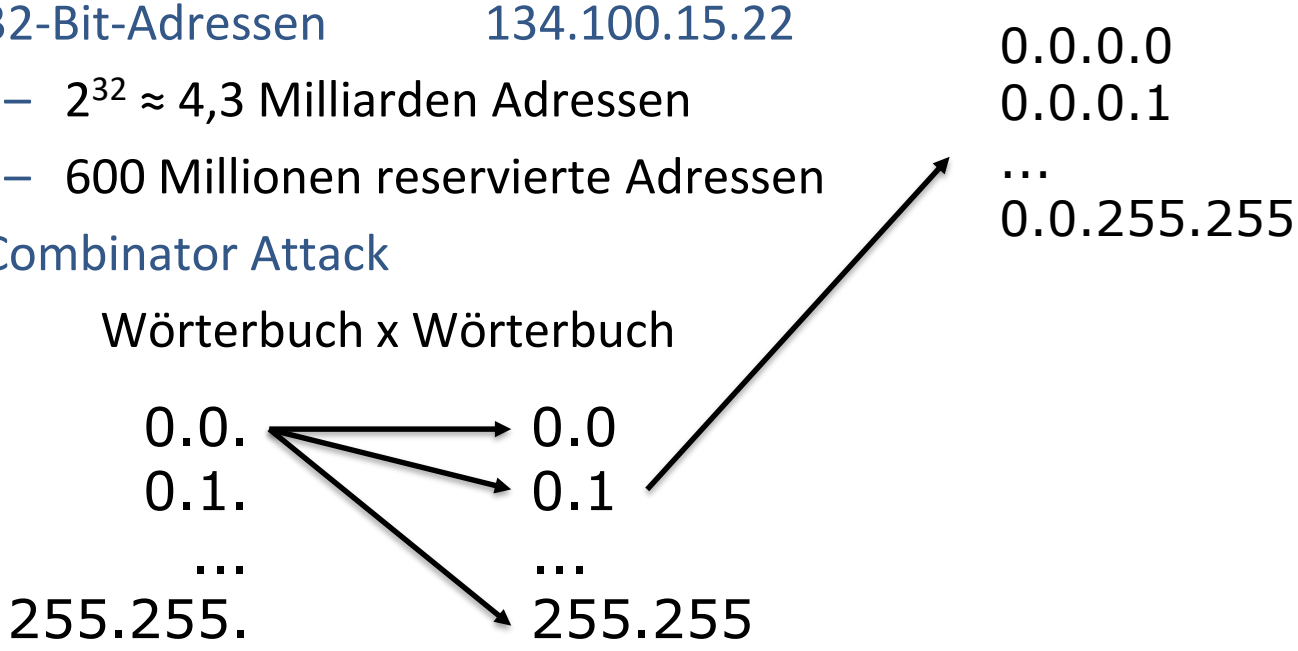
1. Generieren bzw. Sammeln von Eingabedaten
   - Zufällig generierte, gültige Telefonnummern, IP- und MAC-Adressen
   - E-Mail-Adressen wurden zufällig aus einem Leak ausgewählt
2. Hashwerte berechnen
   - MD5
   - SHA-256
3. Hashwerte brechen
   - Durch Ausnutzen der besonderen Struktur der Eingabedaten

Ausstattung

- Desktop-Computer
  - Intel Core i5-6500 CPU mit 3,2 GHz
  - 16 GB RAM
  - Nvidia GeForce GTX 1050 Ti mit 4GB GDDR5
  - Ubuntu 16.04
- Hashcat
  - *„world‘s fastest password recovery tool“*
  - GPU-basiertes Werkzeug, berechnet
    - 6,021 Milliarden MD5-Hashes pro Sekunde
    - 0,844 Milliarden SHA-256-Hashes pro Sekunde
  - Brute-Force- und Wörterbuchangriffe

IPv4-Adressen

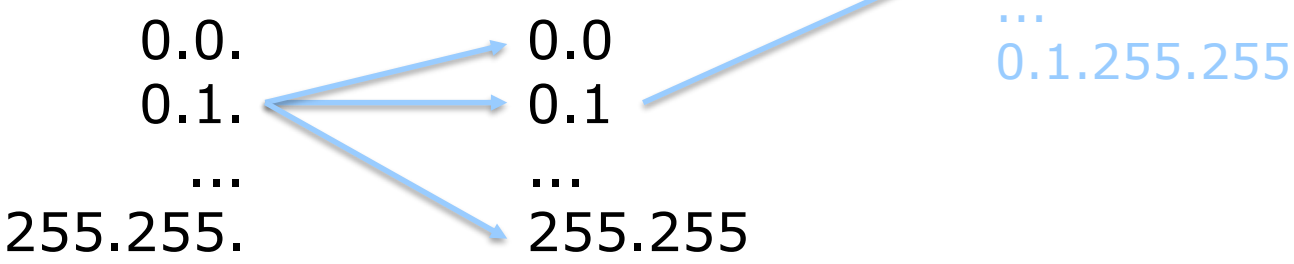
- 32-Bit-Adressen 134.100.15.22
  - $2^{32} \approx 4{,}3$ Milliarden Adressen
  - 600 Millionen reservierte Adressen
- Combinator Attack



IPv4-Adressen

- 32-Bit-Adressen 134.100.15.22
  - $2^{32} \approx 4{,}3$ Milliarden Adressen
  - 600 Millionen reservierte Adressen
- Combinator Attack

Wörterbuch x Wörterbuch

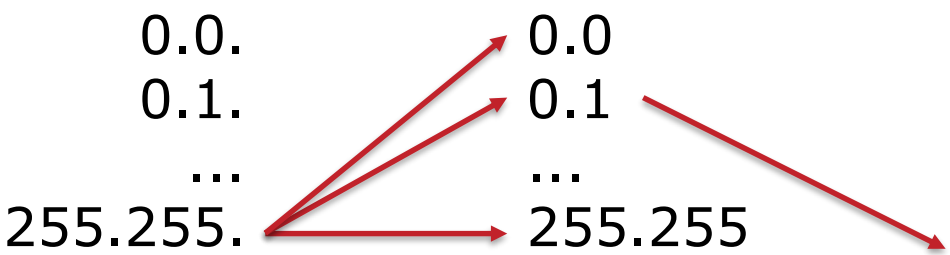


0.0.0.0
0.0.0.1
…
0.0.255.255
0.1.0.0
0.1.0.1
…
0.1.255.255

IPv4-Adressen

- 32-Bit-Adressen 134.100.15.22
  - $2^{32} \approx 4{,}3$ Milliarden Adressen
  - 600 Millionen reservierte Adressen
- Combinator Attack

Wörterbuch x Wörterbuch

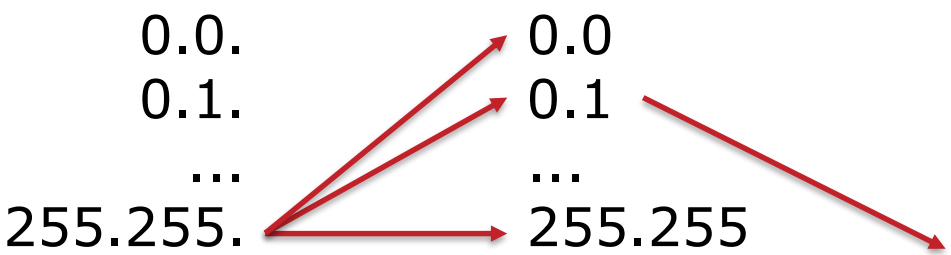


0.0.0.0
0.0.0.1
…
0.0.255.255
0.1.0.0
0.1.0.1
…
0.1.255.255
…
255.255.0.0
255.255.0.1
…
255.255.255.255

IPv4-Adressen

- 32-Bit-Adressen 134.100.15.22
  - $2^{32} \approx 4{,}3$ Milliarden Adressen
  - 600 Millionen reservierte Adressen
- Combinator Attack

Wörterbuch x Wörterbuch



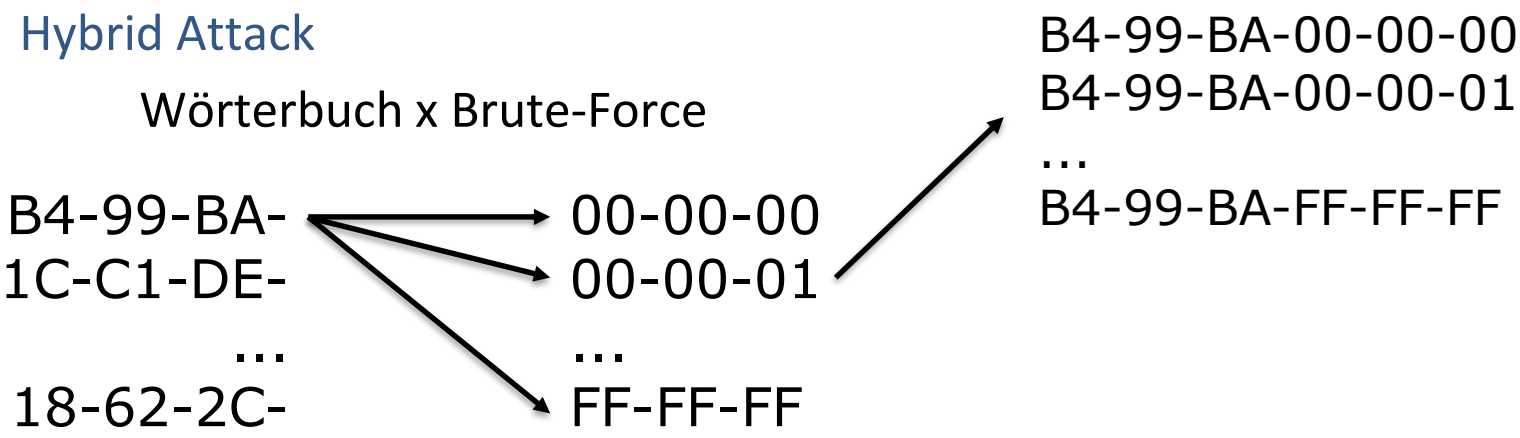
0.0.0.0
0.0.0.1
…
0.0.255.255
0.1.0.0
0.1.0.1
…
0.1.255.255
…
255.255.0.0
255.255.0.1
…
255.255.255.255

Suchraum	Laufzeit MD5	Laufzeit SHA-256	Wiederhergestellte IPv4-Adressen	Wiederherstellungs-rate
$4{,}3 \cdot 10^9$	00:00:25	00:00:30	1 000 000	100%

MAC-Adressen

- 48-Bit-Adresse 1C-C1-DE-FE-23-21
  - $2^{48} \approx 281$ Billionen Adressen
  - Ersten 24 Bit: Organisationally Unique Identifier (OUI)
    - $24215 \cdot 2^{24} \approx 406$ Milliarden Adressen
- Hybrid Attack

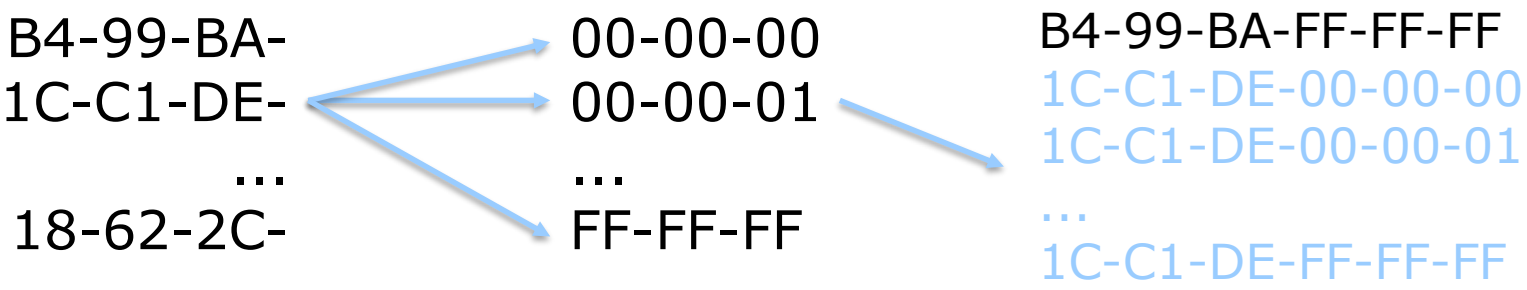
  Wörterbuch x Brute-Force



MAC-Adressen

- 48-Bit-Adresse 1C-C1-DE-FE-23-21
  - $2^{48} \approx 281$ Billionen Adressen
  - Ersten 24 Bit: Organisationally Unique Identifier (OUI)
    - $24215 \cdot 2^{24} \approx 406$ Milliarden Adressen
- Hybrid Attack

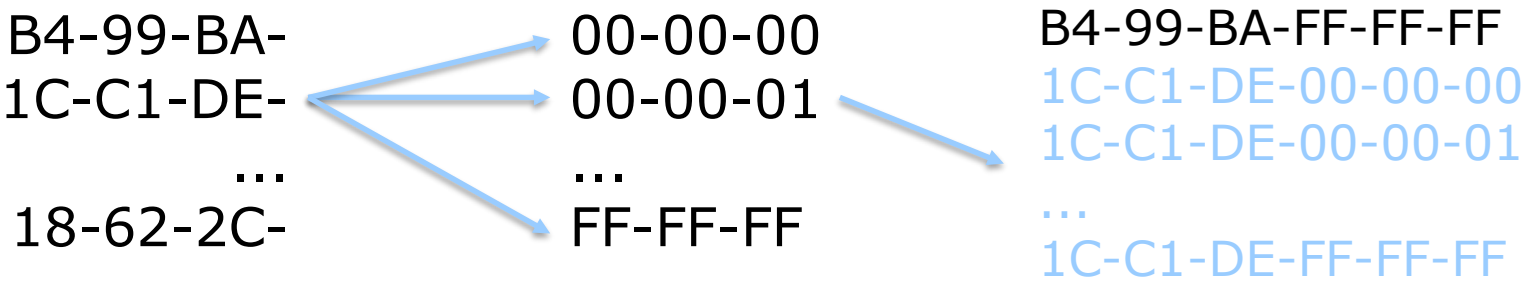
  Wörterbuch x Brute-Force



MAC-Adressen

- 48-Bit-Adresse 1C-C1-DE-FE-23-21
  - $2^{48} \approx 281$ Billionen Adressen
  - Ersten 24 Bit: Organisationally Unique Identifier (OUI)
    - $24215 \cdot 2^{24} \approx 406$ Milliarden Adressen
- Hybrid Attack

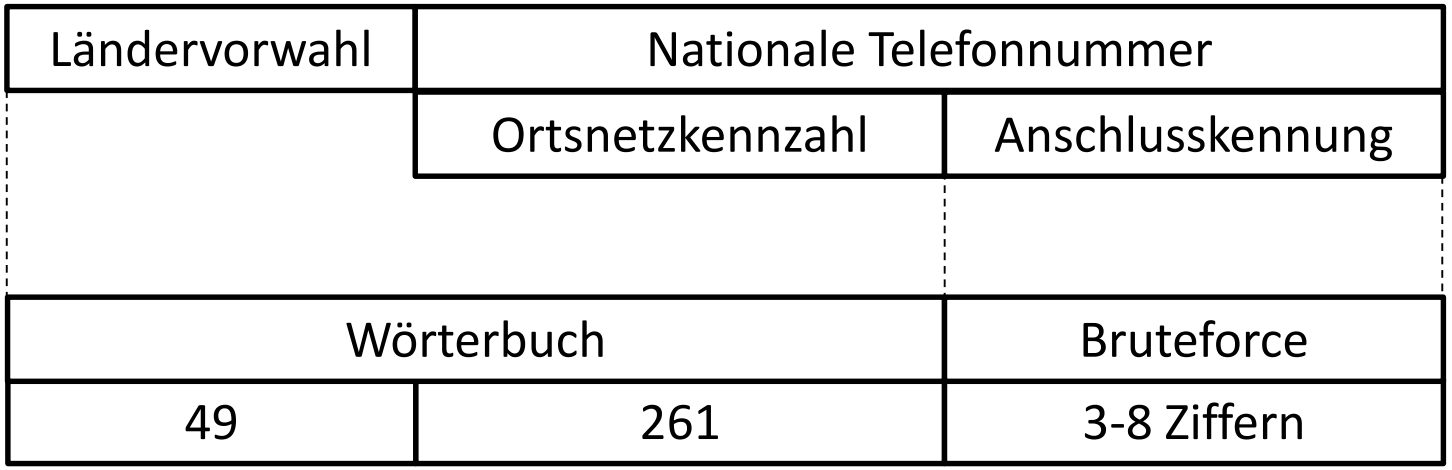
  Wörterbuch x Brute-Force



Suchraum	Laufzeit MD5	Laufzeit SHA-256	Wiederhergestellte MAC-Adressen	Wiederherstellungs-rate
$4{,}1 \cdot 10^{11}$	00:04:01	00:13:22	1 000 000	100%

Telefonnummern

- International Telecommunication Union gibt die Struktur vor
  - Bis zu 15 Ziffern



- Hybrid Attack: Wörterbuch x Brute-Force

Suchraum	Laufzeit MD5	Laufzeit SHA-256	Wiederhergestellte Telefonnummern	Wiederherstellungs-rate
$4 \cdot 10^{11}$	02:28:24	02:34:16	1 000 000	100%

E-Mail-Adressen

- E-Mail-Adressen haben ein bestimmtes Format

  local-part@domain

- Local-Part
  - 64 ASCII-Zeichen
  - Kleinbuchstaben, Zahlen, Sonderzeichen
  - $56^{64} \approx 2^{372} \approx 7{,}7 \cdot 10^{111}$
- Domain
  - 330 Millionen registrierte Domains
  - Wenige Domains werden häufig genutzt

Insgesamt $2{,}2 \cdot 10^{120}$ mögliche E-Mail-Adressen. Zu viel zum Ausprobieren.

E-Mail-Adressen

- Domains
  - Top 10 & Top 100 Domains von OpenPGP-Keyservern
- Local-Part
  - Brute-Force-Angriffe
  - Wörterbuchangriffe
    - Nutzernamen, Passwörter und Wörterbücher
    - Hashcat‘s Rule Engine
      - Verändert Wörter on-the-fly
      - Ermöglicht hohe Auslastung der Grafikkarte

E-Mail-Adressen

- Domains
  - Top 10 & Top 100 Domains von OpenPGP-Keyservern
- Local-Part
  - Brute-Force-Angriffe
  - Wörterbuchangriffe
    - Nutzernamen, Passwörter und Wörterbücher
    - Hashcat‘s Rule Engine
      - Verändert Wörter on-the-fly
      - Ermöglicht hohe Auslastung der Grafikkarte
- Kleines Wörterbuch x Top 100 Domains

Rule Engine	Suchraum	Laufzeit MD5	Laufzeit SHA-256	Wiederhergestellte E-Mail-Adressen	Wiederherstellungs-rate
ohne	$9{,}7 \cdot 10^{8}$	00:01:36	00:01:28	35 117	3,51%
mit	$1{,}9 \cdot 10^{10}$	00:02:25	00:02:45	46 306	4,63%

E-Mail-Adressen

- Bruteforce

Domains	Suchraum	Laufzeit MD5	Laufzeit SHA-256	Wiederhergestellte E-Mail-Adressen	Wiederherstellungs-rate
Top 10	$3{,}5 \cdot 10^{12}$	00:38:34	01:57:03	166 152	16,62%
Top 100	$3{,}5 \cdot 10^{13}$	05:18:20	16:40:17	199 377	19,94%

- Insgesamt wiederhergestellt (Kombination aller Attacken)
  - Top 10: 35,52% nach 3,5 Stunden (MD5)
  - Top 100: 42,52% nach 21 Stunden (MD5)

  Doppelte Laufzeit für SHA-256

Gegenmaßnahmen

- Langsame Hashverfahren
  - Unser Setup berechnet
    - 6 Milliarden MD5-Hashes pro Sekunde
    - 4000 bcrypt-Hashes pro Sekunde
  - Vorherberechnen von 3,3 Milliarden bcrypt-Hashwerten dauert 9,5 Tage

- Salt

  hash(max@example.org + sVm7ifqj)
  - Ein zufälliger Wert wird angehängt
  - Verhindert das Vorherberechnen von Hashwerten
  - Daten können nicht länger einfach getauscht werden

Future Work

- Kontakt zu Datenschutzbehörden
- Welche Struktur haben E-Mail-Adressen?
- Untersuchung von IPv6- und anderen Hashwerten
- Datenschutzfreundliches Tracking

Fazit

- Hashbasierte Pseudonymisierung bietet keinen ausreichenden Schutz personenbezogener Daten
- Aufwendigere Verfahren sind notwendig, um die Daten zu schützen
- Angriffe sind nicht nur theoretisch, sondern auch praktisch und auf günstiger Hardware durchführbar

Hashing of personally identifiable information is not sufficient

Matthias Marx, Ephraim Zimmer, Tobias Mueller, Maximilian Blochberger, Hannes Federrath

marx@informatik.uni-hamburg.de

Literatur

- Joseph Bonneau. “The Science of Guessing: Analyzing an Anonymized Corpus of 70 Million Passwords”. In: IEEE S&P. IEEE, 2012, pp. 538–552.
- Levent Demir et al. “The Pitfalls of Hashing for Privacy”. In: Commun. Surveys Tuts. (99 2017).
- Ed Felten. Does Hashing Make Data “Anonymous”? Federal Trade Commission. Apr. 22, 2012. url: https://www.ftc.gov/node/605301 (besucht am 14.12.2017).
- Google. Integrating CRM Data with Google Analytics to create AdWords Remarketing Audiences. Jan. 2017. url: https://developers.google.com/analytics/solutions/crm-integration (besucht am 14.12.2017).
- infsoft GmbH. FAQ – Indoor Positioning – Data Protection. 2018. url: https://www.infsoft.com/technology/faq (besucht am 14.02.2018).
- Arvind Narayanan and Vitaly Shmatikov. “Fast dictionary attacks on passwords using time-space tradeoff”. In: ACM CCS. ACM, 2005, pp. 364–372.
- Oracle Corporation. BlueKai Platform – Offline match integration. 2018. url: https://docs.oracle.com/cloud/latest/marketingcs_gs/OMCDA/IntegratingBlueKaiPlatform/DataIngest/offline_match.html (besucht am 14.02.2018).