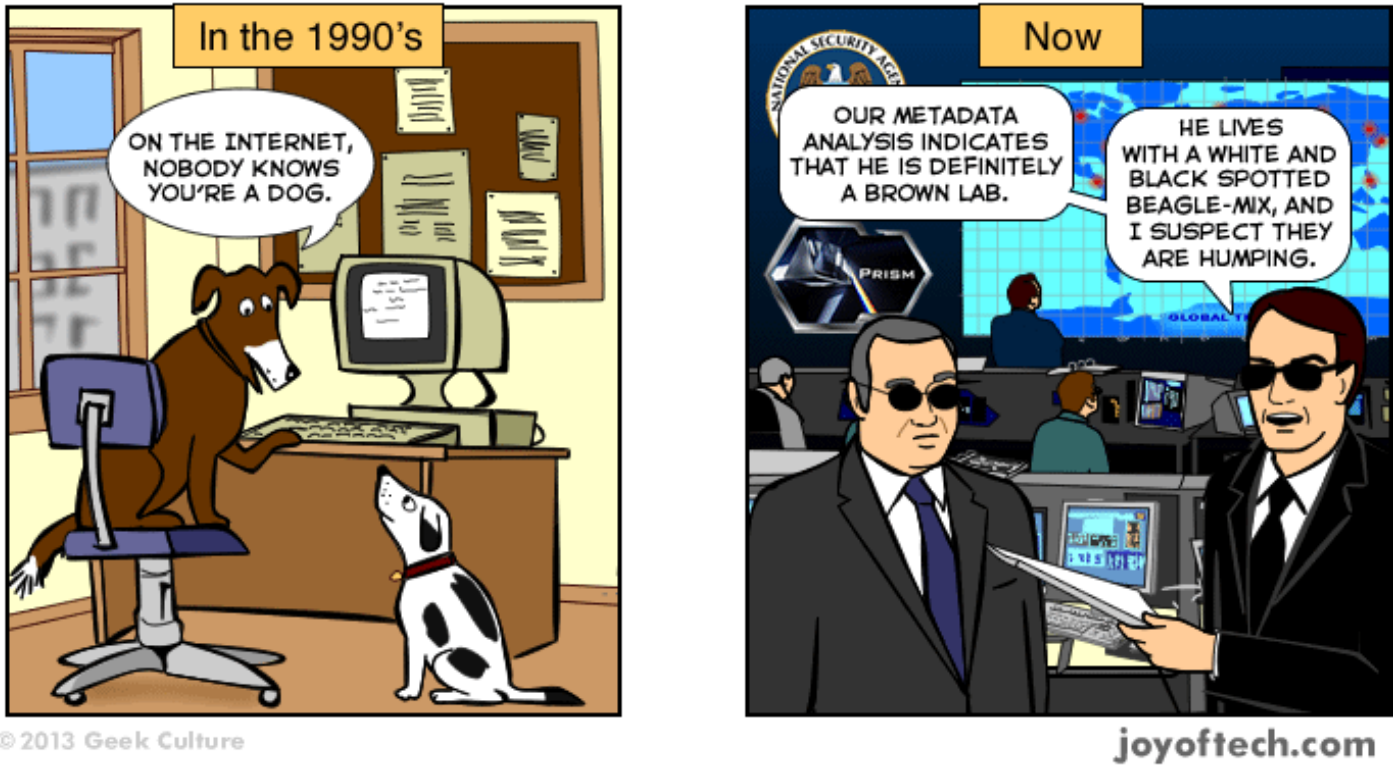
Wie Profilbildung im Netz funktioniert

– und du dich technisch schützen kannst

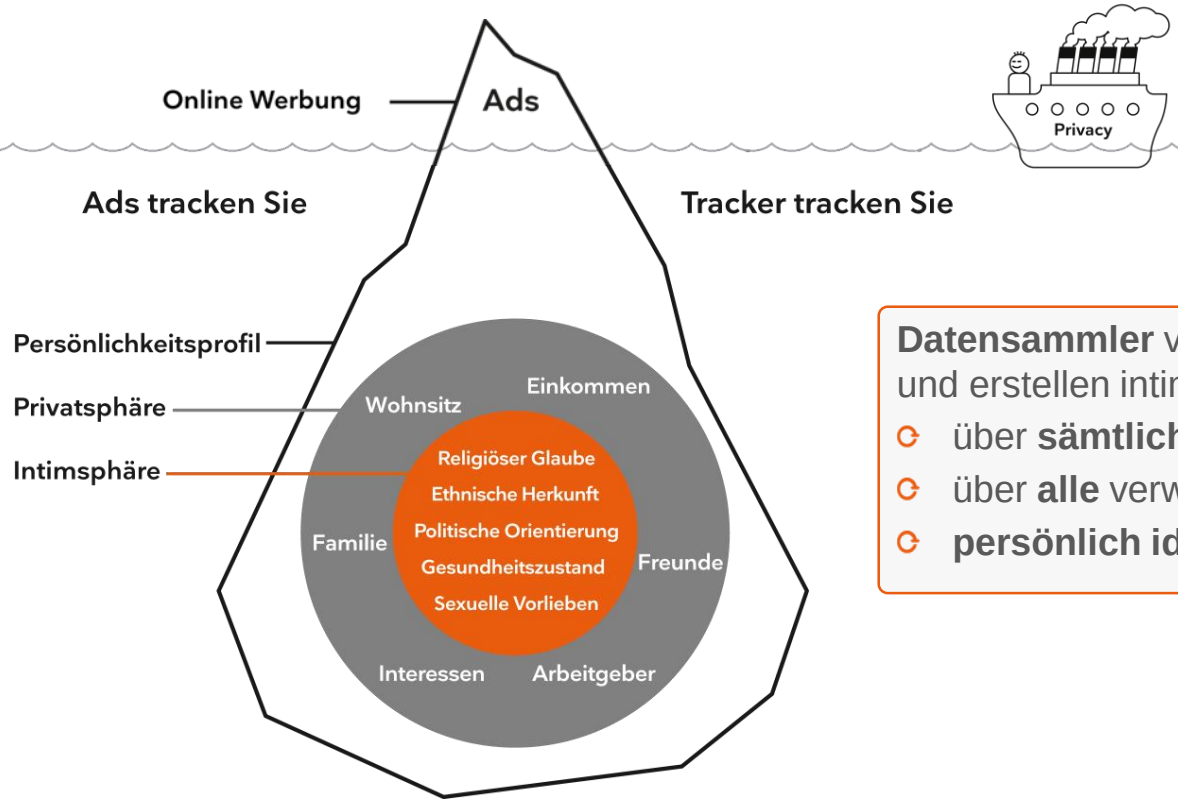
Christian Bennefeld (aka „Benne“)
Gründer und Gesellschafter **etracker**
Gründer und Geschäftsführer **eBlocker**



Ein typisches Online-Kauferlebnis

Zitat A. Merkel am 05.11.2014

Wir sehen nur die Spitze des Eisbergs



Datensammler verfolgen unser Surfverhalten und erstellen intime **Persönlichkeitsprofile**

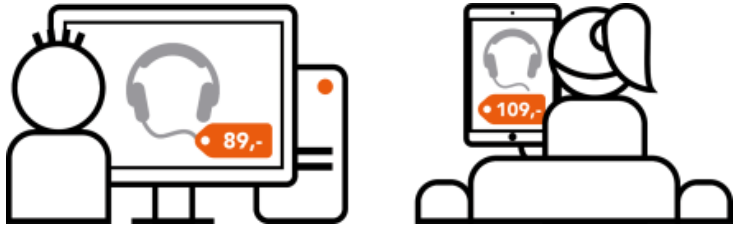
- über **sämtliche** besuchten Websites
- über **alle** verwendeten Geräte
- **persönlich identifizierbar**

Gefahren von Persönlichkeitsprofilen

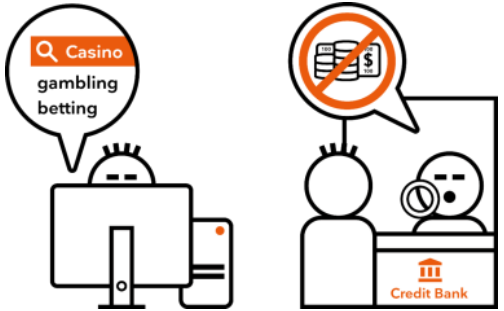
„Ich hab‘ doch nix zu verbergen.“

Gefahren von Persönlichkeitsprofilen

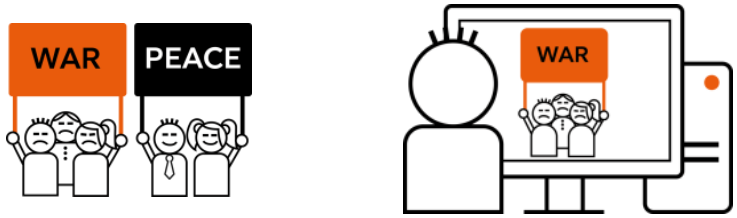
eBlocker. Switch on Privacy.



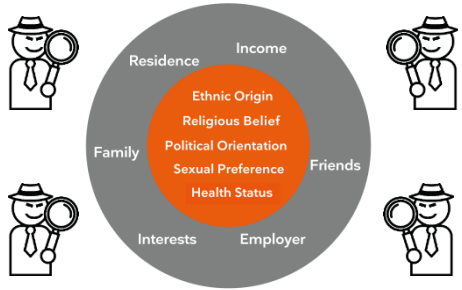
Preisdiskriminierung



Bonitäts- und Gesundheits-Scoring



Manipulation & Filter Blase



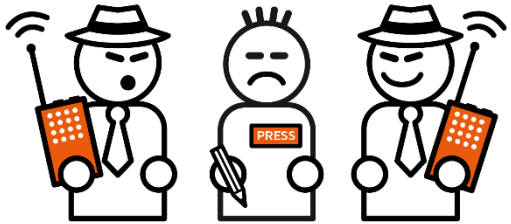
Identitätsdiebstahl & Datenhandel

Besondere Gefahren für Aktivisten

eBlocker
Switch on Privacy.

Identifikation

Diskriminierung



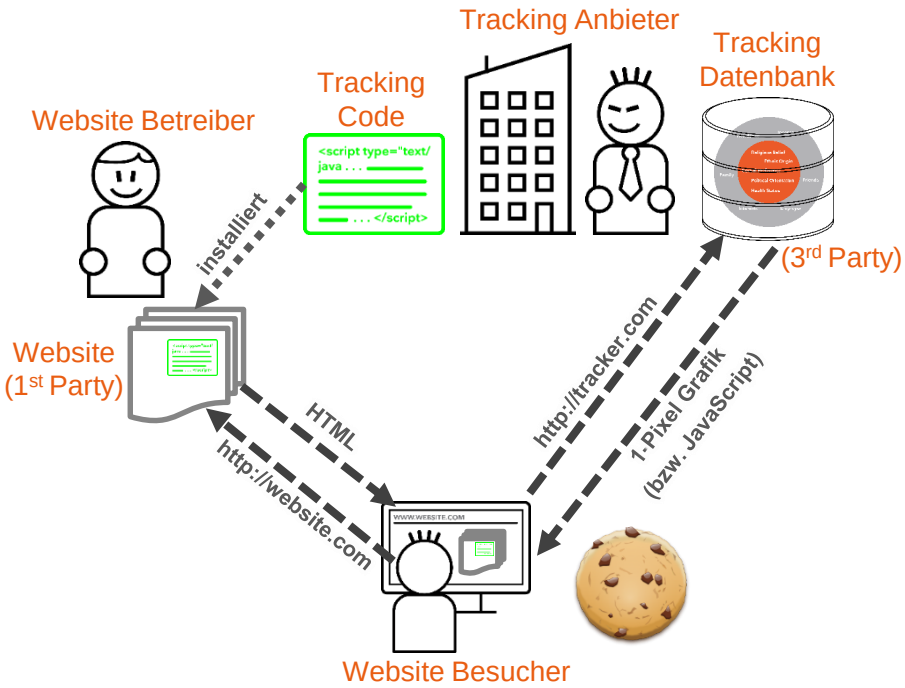
Observation

Verhaftung

Wie Tracking funktioniert

Vom Seitenaufruf zum Persönlichkeitsprofil

Wie funktioniert Tracking?

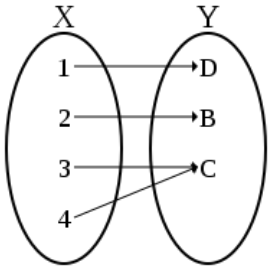


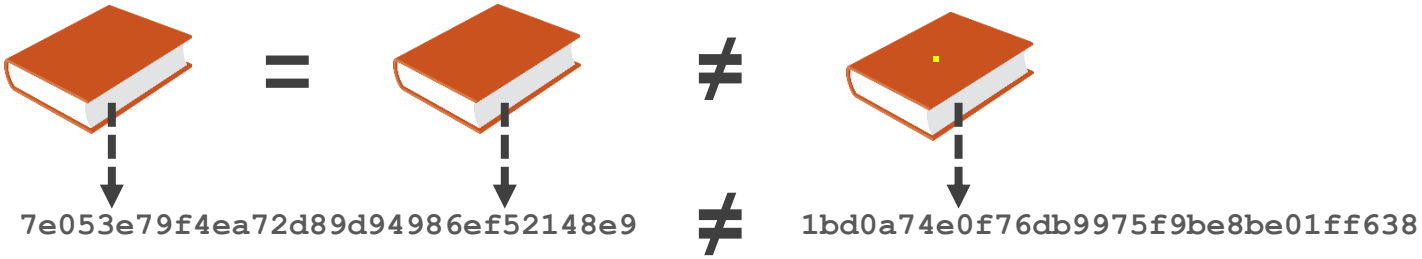
Sichtbare 3rd Party „Tracking-Pixel“

- **Social Plugins** wie Facebook Like-Button, Google +1
- **Online-Werbung** wie DoubleClick, Adsense, Taboola, …
- **Eingebundene Inhalte** wie Google Maps, Youtube, Twitter, …

Exkurs: Wie funktioniert Hashing?

- Mathematische (surjektive) Abbildung **Y:= Hash(X)**
- Für **A = B** gilt stets auch **Hash(A) = Hash(B)**
- Aber: **Hash(A) = Hash(B) bedeutet nicht, dass A = B**
- Hash hat immer die **gleiche Länge** (≈ Kompression)
  - Beispiel **MD5 Hash** = 128 Bit Länge (typische Darstellung in 32 Zeichen á 4 Bit)





Merke: **Ein Hash ist ähnlich einer „Prüfsumme“ –** der Eingangswert ist nicht eindeutig „rückrechenbar“

Techniken zur Besucher-Wiedererkennung

eBlocker. Switch on Privacy.

Profilbildung über verschiedene Sessions und Websites
erfordert Wiedererkennung des Besuchers

- **Cookies & Tags**

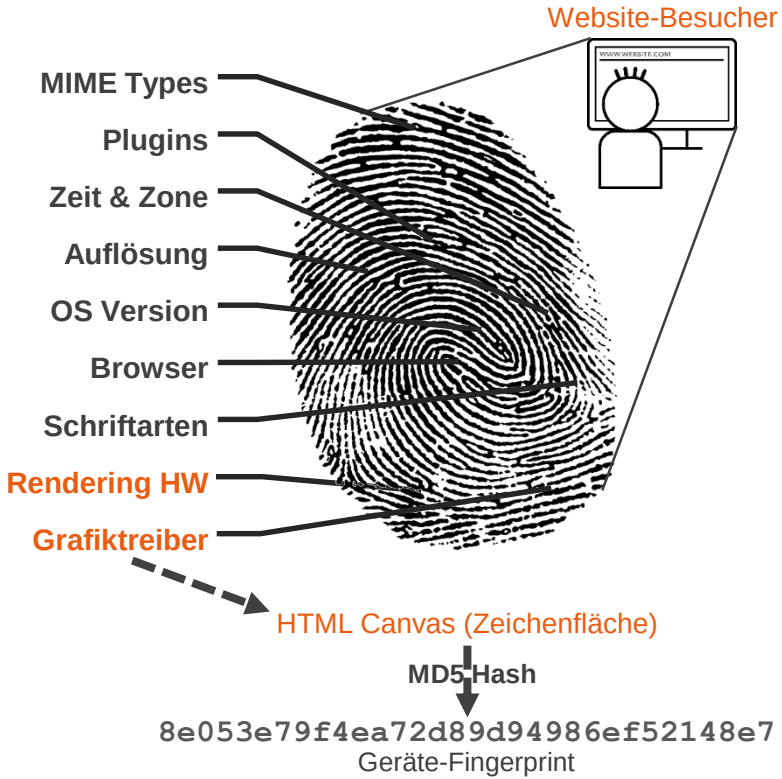
  - enthalten **eindeutige ID** – häufig **Hash über E-Mail / Handynummer** des Nutzers
  - können bei **Folgebesuchen wieder ausgelesen** und dem Profil zugeordnet werden
  - werden über JavaScript **als „1st Party“ Cookies getarnt**, obwohl meist „3rd Party“
  - **„Tagging“** mit LSOs: **L**ocal **S**hared **O**bjects
    - HTML5 localStorage, Flash-Cookies, Silverlight-Cookies, …
    - LSOs können nicht so leicht gelöscht werden

Cookies sind häufig nur noch Fall-Back-Methode,
da sie leicht geblockt und gelöscht werden können

Nicht-invasive Wiedererkennung

- **Fingerprinting**
  - Ermittlung **technischer Merkmale** des Gerätes/Browsers **ohne LSOs**
  - **Canvas Fingerprinting:** Eigenschaften der Grafik-Hardware und Schriftarten
  - **Cross-Browser Identifikation** eines einzelnen Gerätes möglich
- **App IDs & Geräte IDs**
  - Apps generieren UUID und/oder lesen Geräte IDs (IMEI, MAC, Telefonnr.) aus

Professionelle Tracker benötigen keine Cookies oder LSOs



Und was ist mit IP-Adressen?

eBlocker
Switch on Privacy.

IP-Adressen sind **personenbezogene Daten**

- Ohne Einwilligung des Nutzers **nicht verwendbar** (in der EU)
- Speicherung & Verarbeitung **nur nach Verkürzung** (123.235.146.**XXX**)
- **Geografische Lokalisierung** (Ortsebene) mit verkürzter IP-Adresse möglich
- IP-Anonymisierung über **Tor/VPN/Proxy kann geo. Herkunft verschleiern**

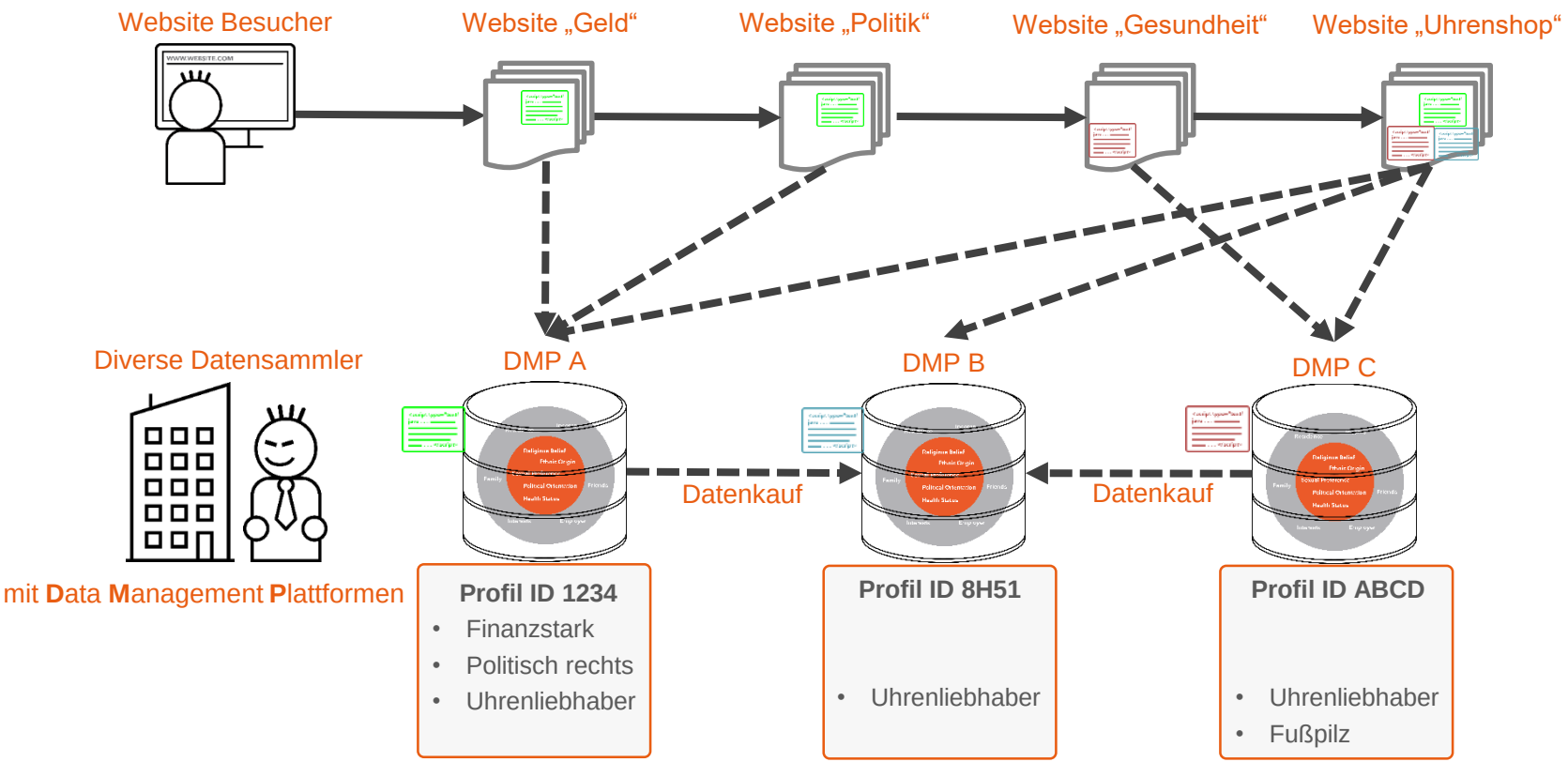
Aber: (Europäische) Tracker nutzen
gar keine IP-Adressen zur Profilbildung

Achtung: Tracker erfassen Profildaten
auch bei IP-Anonymisierung

Datenhandel und personenbezogene Überwachung

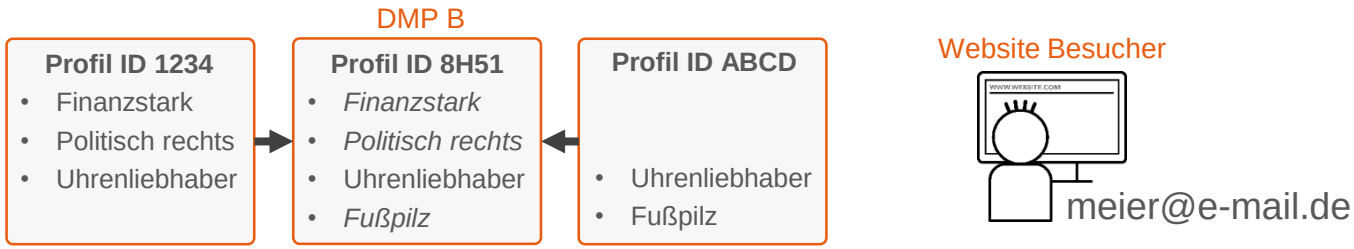
Datenanreicherung, De-Pseudonymisierung und personenbezogene Profile

Profilanreicherung & Datenhandel



Datenanreicherung mit „Cookie-Matching“

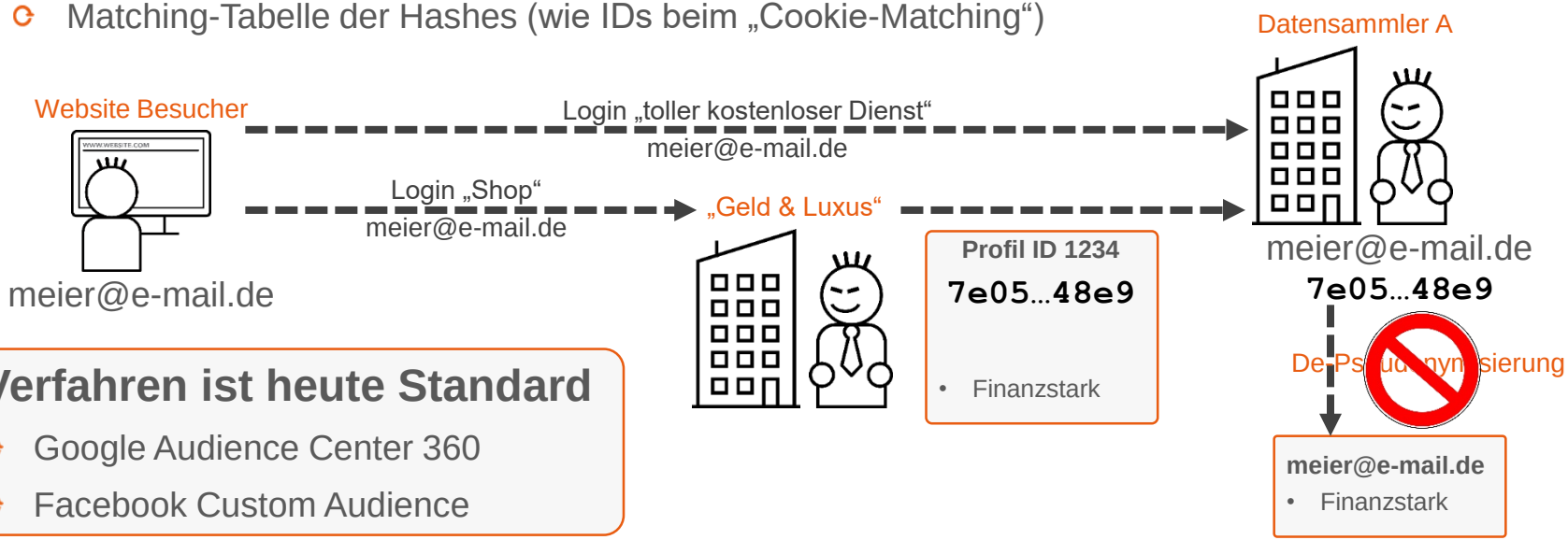
- Datenanreicherung durch **Client-seitigen Austausch** der DMPs
  - Bei Tracker-Aufruf werden IDs und/oder Profildaten in Variablen gegenseitig übergeben
  - **ID Matching-Tabelle** wird bei jeder DMP geführt
  - Beispiel: **DMP B kauft Daten** von DMP A und DMP C



DMP A	DMP B	DMP C	Hash über E-Mail Adresse
1234	8H51	ABCD	`7e053e79f4ea72d89d94986ef52148e9`

Datenanreicherung über Pseudonym / Hash

- Nicht personenbezogene, **pseudonyme Daten dürfen gehandelt werden**
- **Hashes von E-Mail / Handynummer sind pseudonyme Daten**
- Datenanreicherung durch **serverseitigen Daten- und Hash-Austausch**
  - Matching-Tabelle der Hashes (wie IDs beim „Cookie-Matching“)

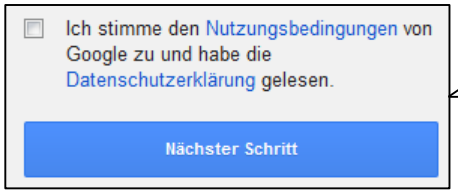


Verfahren ist heute Standard

- Google Audience Center 360
- Facebook Custom Audience

Personenbezogene Persönlichkeitsprofile

- **Personenbezogene Profilbildung nur bei Einwilligung** zulässig
- Aber: **Einwilligung wird regelmäßig erteilt** bei „gratis“ Account-Anlage



34 DIN-A4 Seiten Datenschutzerklärung

Google Datenschutzerklärung Stand: 31.03.2020

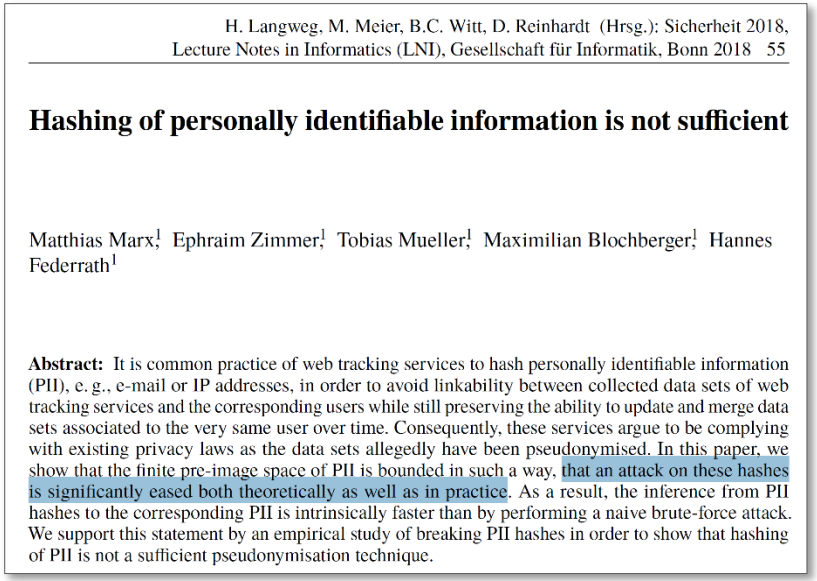
„… können Ihre **Aktivitäten auf anderen Websites und in Apps** mit Ihren **personenbezogenen Daten verknüpft** werden“

„Wenn Sie Websites besuchen, auf denen **Google Analytics** eingesetzt wird, […wird Google…] Daten über Ihre Aktivitäten auf dieser Website mit **Aktivitäten auf anderen Websites verknüpfen**, auf denen ebenfalls unsere **Werbedienste** genutzt werden.“

Personenbezogene Totalüberwachung durch Google

>80% der Websites nutzen **Google Analytics**
>90% der Nutzer verwenden **Google Suche**

De-Pseudonymisierung leicht gemacht



H. Langweg, M. Meier, B.C. Witt, D. Reinhardt (Hrsg.): Sicherheit 2018,
Lecture Notes in Informatics (LNI), Gesellschaft für Informatik, Bonn 2018 55

Hashing of personally identifiable information is not sufficient

Matthias Marx[1], Ephraim Zimmer[1], Tobias Mueller[1], Maximilian Blochberger[1], Hannes Federrath[1]

Abstract: It is common practice of web tracking services to hash personally identifiable information (PII), e. g., e-mail or IP addresses, in order to avoid linkability between collected data sets of web tracking services and the corresponding users while still preserving the ability to update and merge data sets associated to the very same user over time. Consequently, these services argue to be complying with existing privacy laws as the data sets allegedly have been pseudonymised. In this paper, we show that the finite pre-image space of PII is bounded in such a way, that an attack on these hashes is significantly eased both theoretically as well as in practice. As a result, the inference from PII hashes to the corresponding PII is intrinsically faster than by performing a naive brute-force attack. We support this statement by an empirical study of breaking PII hashes in order to show that hashing of PII is not a sufficient pseudonymisation technique.

Universität Hamburg, April 2018:
https://dl.gi.de/bitstream/handle/20.500.12116/16294/sicherheit2018-04.pdf

- **„Zurückrechnung“** von Hashes **unter Bedingungen** möglich
- Klassischer „Brute-Force“ Ansatz
  - **Alles** ausprobieren = zu **viele Kombinationen** = zu **viel Zeit** / Ressourcen notwendig
- **Reduktion der Kombinationen**
  - **Kenntnis über Aufbau** z.B. E-Mail-Adresse: first.last@provider.TLD

PC in <24h auf 1 Mio. Hashes

- **IP-Adressen: 100%** Recovery
- **Telefonnummern: 100%** Recovery
- **E-Mail Adressen: 43%** Recovery

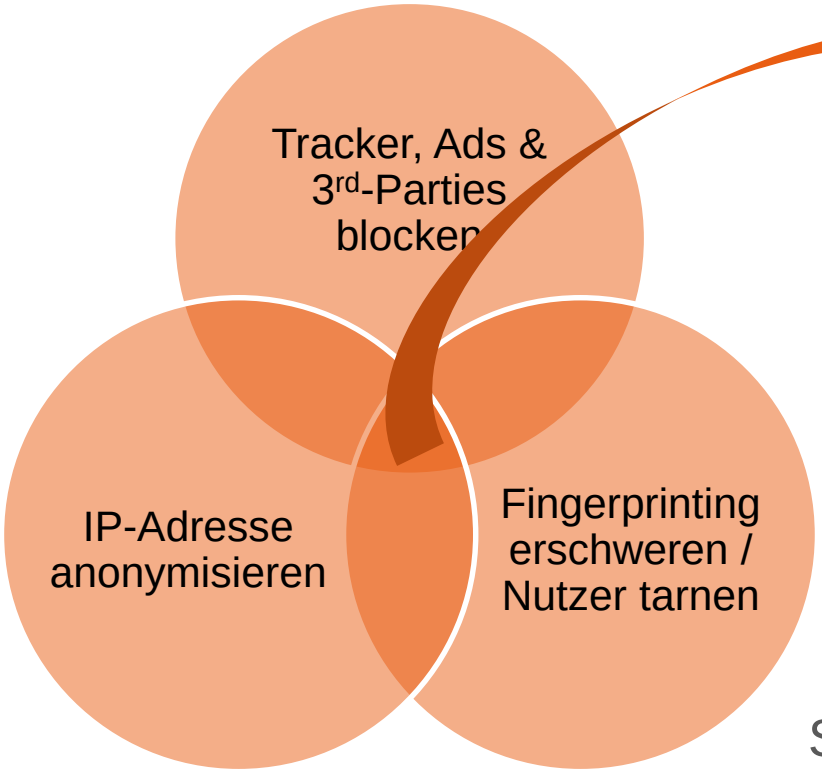
Wie man sich technisch schützen kann

Betriebsgeheimnisse der eBlocker Technologie

eBlocker vs. eBlocker Open Source

- **eBlocker GmbH**
  - Okt. 2014 gegründet
  - Entwicklung eines Plug & Play Gerätes zum Schutz der Privatsphäre
  - Business Modell: Verkauf von Geräten & Software-Lizenzen zum Selbstbau
  - Zehntausende Kunden gewonnen, zahlreiche Innovationspreise erhalten
  - 2019 insolvent, da Hauptinvestor kurzfristig abgesprungen war
- **eBlocker Open Source UG**
  - Dez. 2019 gegründet (von ehemaligen eBlocker GmbH Gründern)
  - Übernahme der eBlocker Technologie vom Insolvenzverwalter
  - Ziel: eBlocker Technologie **kostenfrei** jedermann zur Verfügung stellen
    - **Non-Profit** auf **ehrenamtlicher** Basis (wir haben andere Jobs ☺)
    - **Software für Raspberry Pi** zum Geräte-Selbstbau
    - **Open Source Entwicklung** gemeinsam mit Community
    - Keine Hintertüren, **kein Business Modell**
    - Deckung der Kosten **über Spenden**

Komponenten für Privatsphäreschutz



Die eBlocker Idee:
Schutz **aller** Geräte auf Netzwerkebene

Bekannte Tools: Keine guten Lösungen

- Browser Private Mode oder Cookies löschen
  - Nur Schutz vor lokalem Storage, kaum Tracker-Schutz, keine IP-Anon.
- Browser Plugins (Ghostery, Privacy Badger, uBlock, etc.)
  - Schützt keine Apps, nicht für alle Geräte / OS, keine IP-Anon.
- Tor-Browser
  - Nur IP-Anonymisierung, Zwang zu neuem Browser, Apps werden nicht geschützt, nicht für alle OS
- Lokales Tor-Gateway oder VPN-Gateway auf Client Rechner oder im LAN
  - Nur IP-Anonymisierung, kein Tracker-Schutz, gut: gesamte Kommunikation wird „IP-anonymisiert“
- Internet Gateway (disconnect.me, hidemyass, etc.)
  - Gut: Tracker-Schutz inkl. IP-Anonymisierung, Nachteil: großes Vertrauen in Anbieter, SSL-Traffic?
- Für alle Tools gilt: **Für nicht-Techniker kaum zu bedienen**
- **Fazit:** Gute, einfache Lösung für jedermann und alle Geräte existiert nicht

eBlocker Solution Approach

eBlocker
Switch on Privacy.

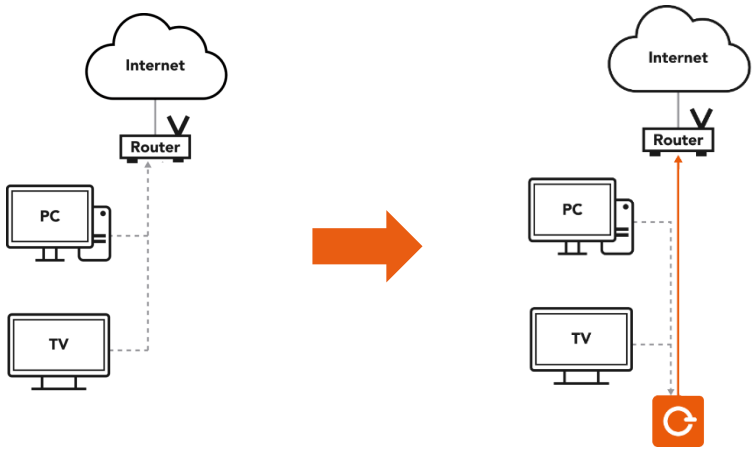
Technical approach

- Plug & Play setup; easy to operate
- All network devices are protected
- Individual protection per device/user
- All devices and apps work as usual
- Existing WiFi router remains in place
- No changes to network topology
- Privacy by design
- No user data in the cloud
- Hardware independent development

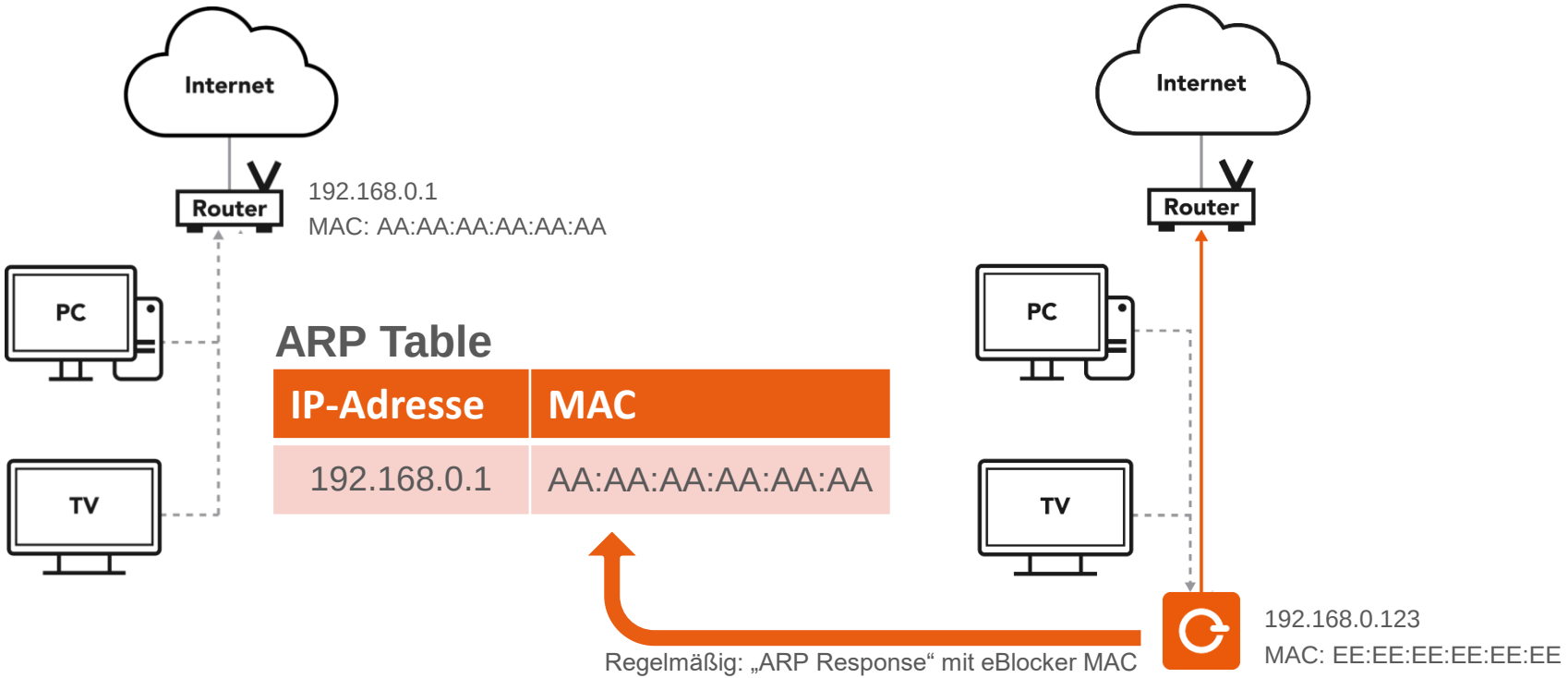
- **eBlockerOS**
  - Open Source Code
    - Core in Java, JavaScript, C & Ruby
    - UI based on SPAs w/ REST-API
  - Based on standard protocols & OSS
    - HTTP, TLS, DNS, DHCP, ICAP
    - Debian Linux
- **Hardware Recommendations**
  - Standard ARM SBC (Raspberry Pi 4)
    - 4 core, 1 GHz, 2 GB RAM, 8 GB eMMC
  - Runs on any Linux system, incl. VM
    - Also prototyped on standard routers

Wie funktioniert „Plug & Play“?

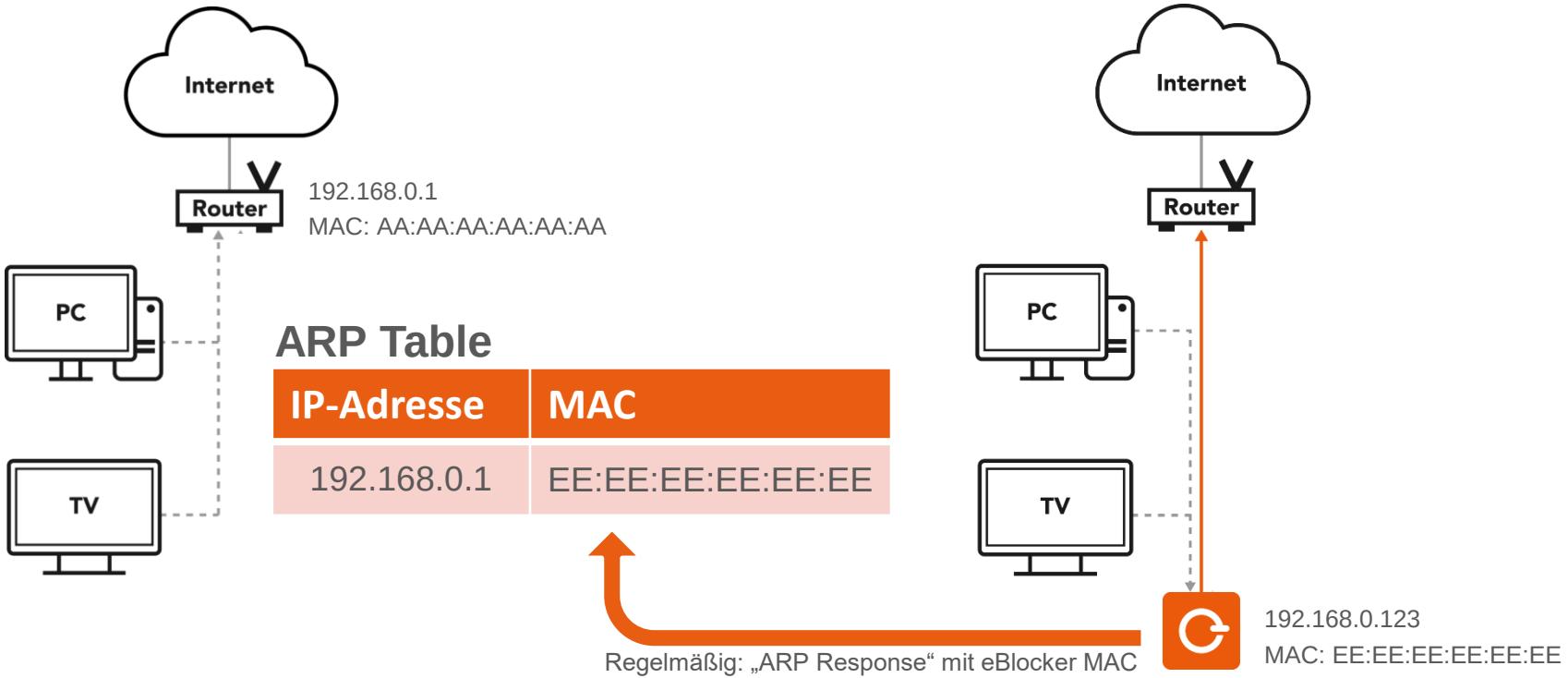
- eBlocker muss als Gateway gesamten IP-Verkehr zum Filtern erhalten
- **Ziel:** „Plug & Play setup”
- **Aber:** Konfiguration von Gateway und DHCP zu komplex für „DAUs“
- **Lösung:** Hacker Angriffs-Technik „ARP-Spoofing“
- „Gespoofte“ Geräte schickt alle IP-Pakete zum eBlocker ✓



ARP-Spoofing

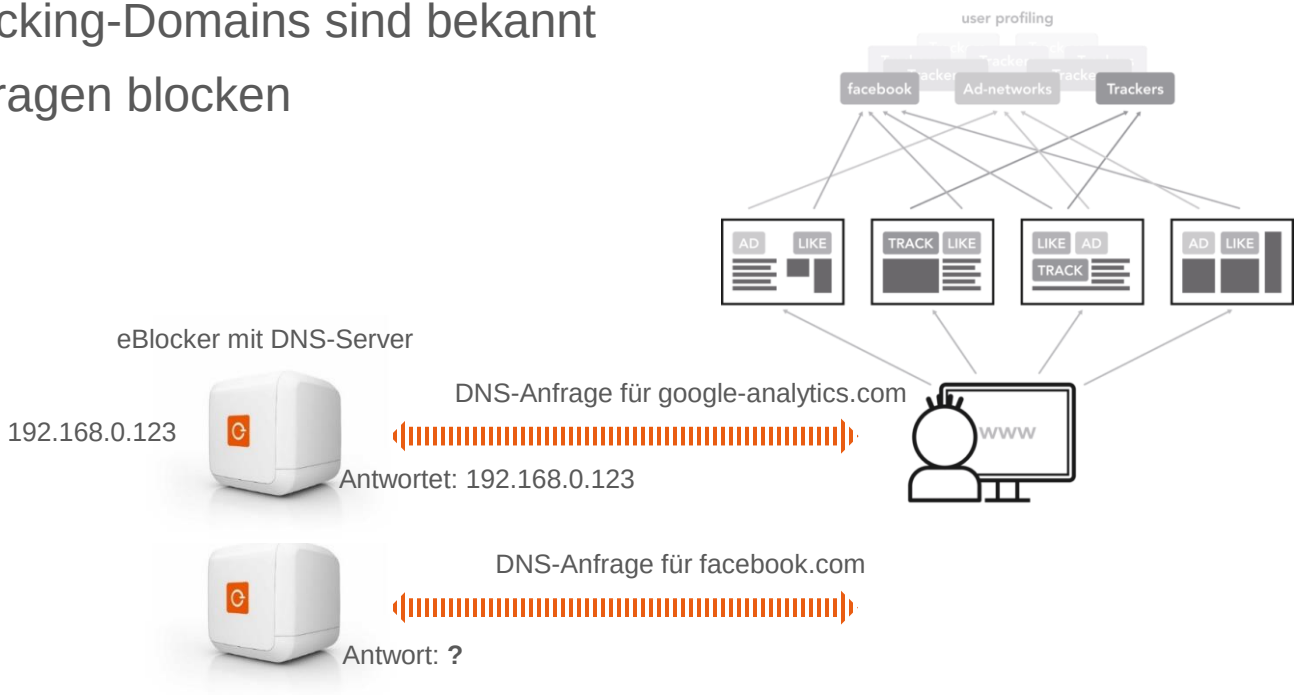


ARP-Spoofing



Verfahren zum „Tracker blocken“

- Wie kann man Tracker auf Netzwerkebene erkennen und blocken?
- **Annahme:** Tracking-Domains sind bekannt
- **Idee:** DNS-Abfragen blocken



- **Aber:**

Probleme beim DNS-Blocking & Alternative

- **DNS-Blocking Probleme**
  - **Overblocking**
    - Domain ist nicht grundsätzlich „böse“, sondern nur bestimmte URLs einer Domain – gesamte Domain wird aber trotzdem geblockt
  - **Underblocking**
    - IP-Adresse eines Trackers kann hinter beliebigem DNS-Eintrag sein (z.B. „test.anwender.com“)
- **Alternative zu DNS-Blocking:** Deep Packet Inspection (DPI)
  - **Ziel:** Pattern-Matching auf Target URL, um Tracker genauer zu erkennen z.B. /*tracker*
  - **Bei einem Match:** eBlocker beantwortet Request anstelle des Targets
  - **Herausforderung:** URL ist bei SSL nicht bekannt, sondern nur Domain

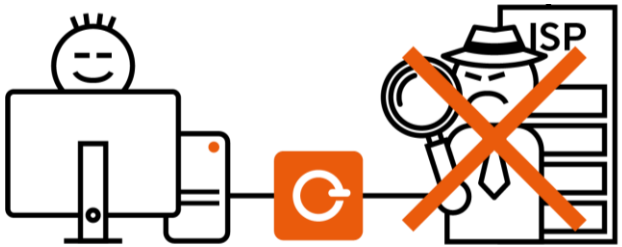
Exkurs: Deep Packet Inspection bei SSL

- **Ziel:** Mustervergleich auf Target URL
- **Problem:** Großteil des Datenverkehrs ist per SSL verschlüsselt
- **Lösung:** SSL-Bumping / SSL Man-in-the-middle
- eBlocker agiert als Certification Authority und terminiert die SSL-Verbindung ✓

- eBlocker nutzt DNS-Blocking als Fallback, falls SSL-Bumping nicht möglich ✓
  - Z.B. bei SmartTVs, Spielekonsolen oder anderen IoT-Geräten

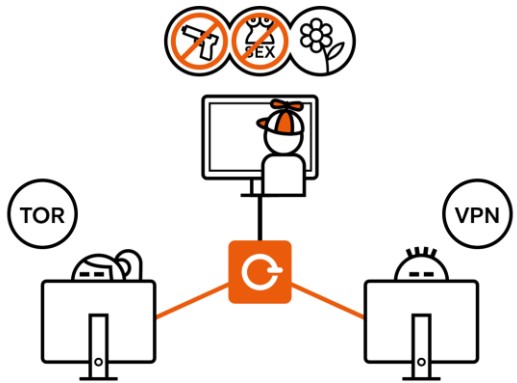
Exkurs: DNS-Spuren beim Provider

- **Problem:** Provider DNS erhält aufgerufene Domains, kann Nutzer profilieren
- **Lösung:** Anderen DNS-Server verwenden?
  - Nachteil: Anderer DNS-Server kann Nutzer genauso profilieren
  - Besser: DNS-Requests auf verschiedene DNS-Server verteilen (z.B Round-Robin auf DNS-Liste)
- **Alternative:** Nur DNS-Requests über Tor routen (nicht gesamten Traffic)
  - Vorteil: Gute Geschwindigkeit, da DNS nur wenige Bytes ausmacht (dann Caching)
- eBlocker beherrscht DNS-Round-Robin und DNS über Tor ✓



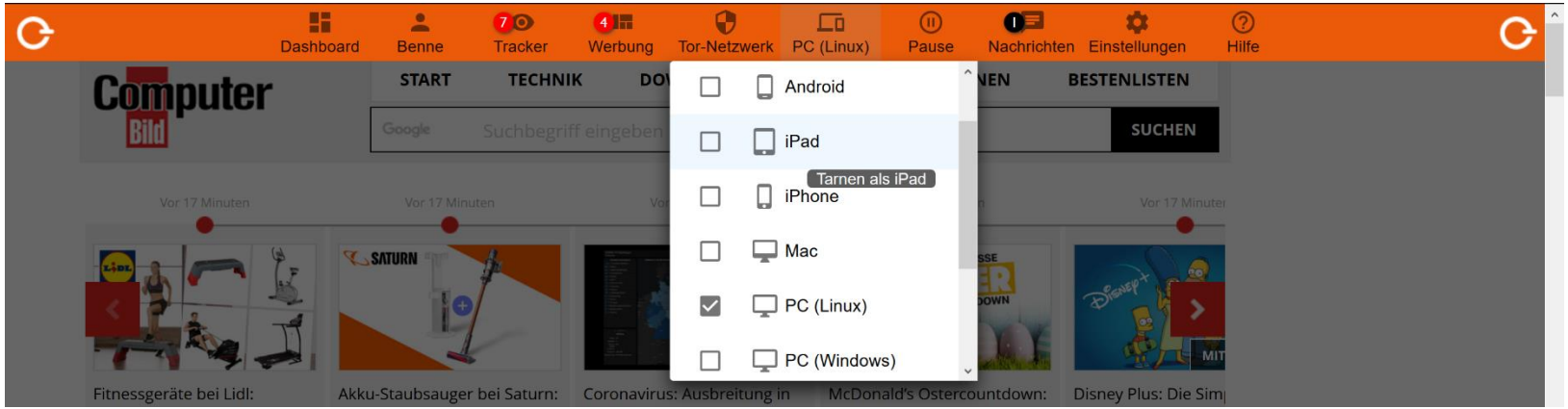
IP-Adresse anonymisieren

- **Ziel:** Eigene IP-Adresse soll verborgen / anonymisiert werden
- **Lösung:** Traffic muss über externen Service geroutet werden
  - **VPN:** OpenVPN Endpoint im eBlocker (kann unterschiedliche Provider parallel bedienen)
  - **Tor:** Tor Endpoint im eBlocker (Exit-node kann konfiguriert werden)
- Individuelles Traffic-Routing von jedem Endgerät zum jeweiligen Endpoint
  - Über **IPTables** im eBlocker wird der Traffic jedes Gerätes einem Endpoint zugeordnet ✓



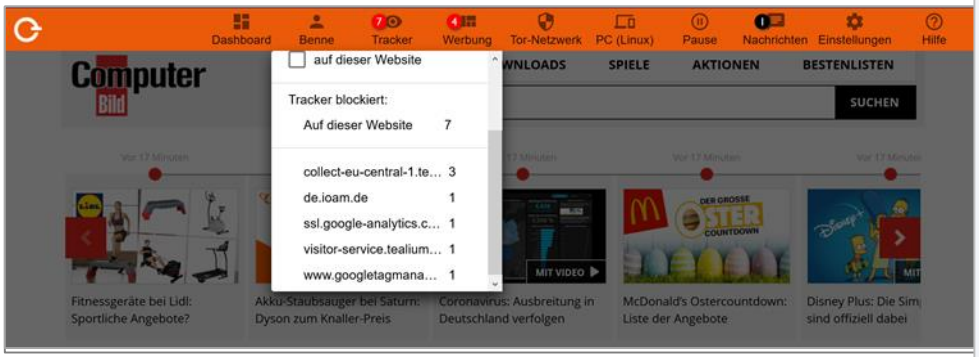
Fingerprinting erschweren

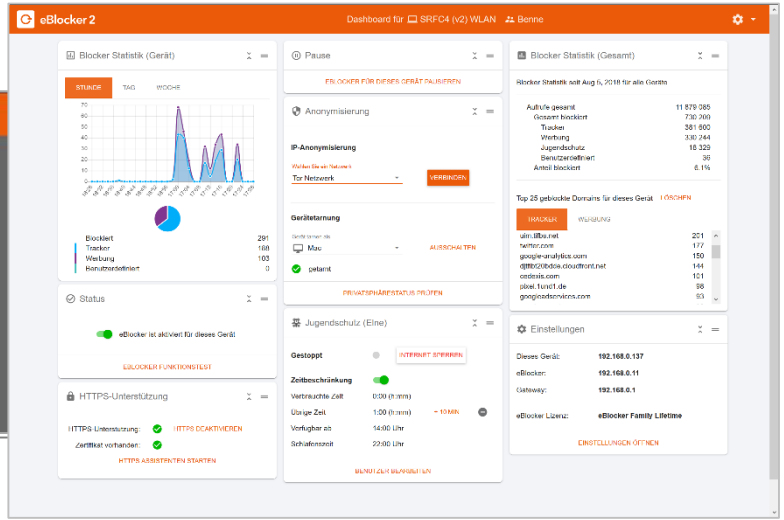
- **Ziel:** Geräteinformation auf Protokollebene verschleiern
- **Lösung:** User Agent verändern (nur mit SSL-Bumping bzw. unverschlüsselt)
- **Wichtig:** User Agent muss statistisch aus der „breiten Masse“ kommen



User Interface

- **Ziel:** Geräte-individuelle Steuerung des eBlockers durch den Nutzer
- **Lösung:** Einfügen einer „Controlbar“ in jeder HTML-Seite
  - JavaScript/HTML wird automatisch in HTML-Seite eingefügt und kommuniziert mit eBlocker
- **Alternative:** Steuerungs-Dashboard über einfache „eBlocker.box“

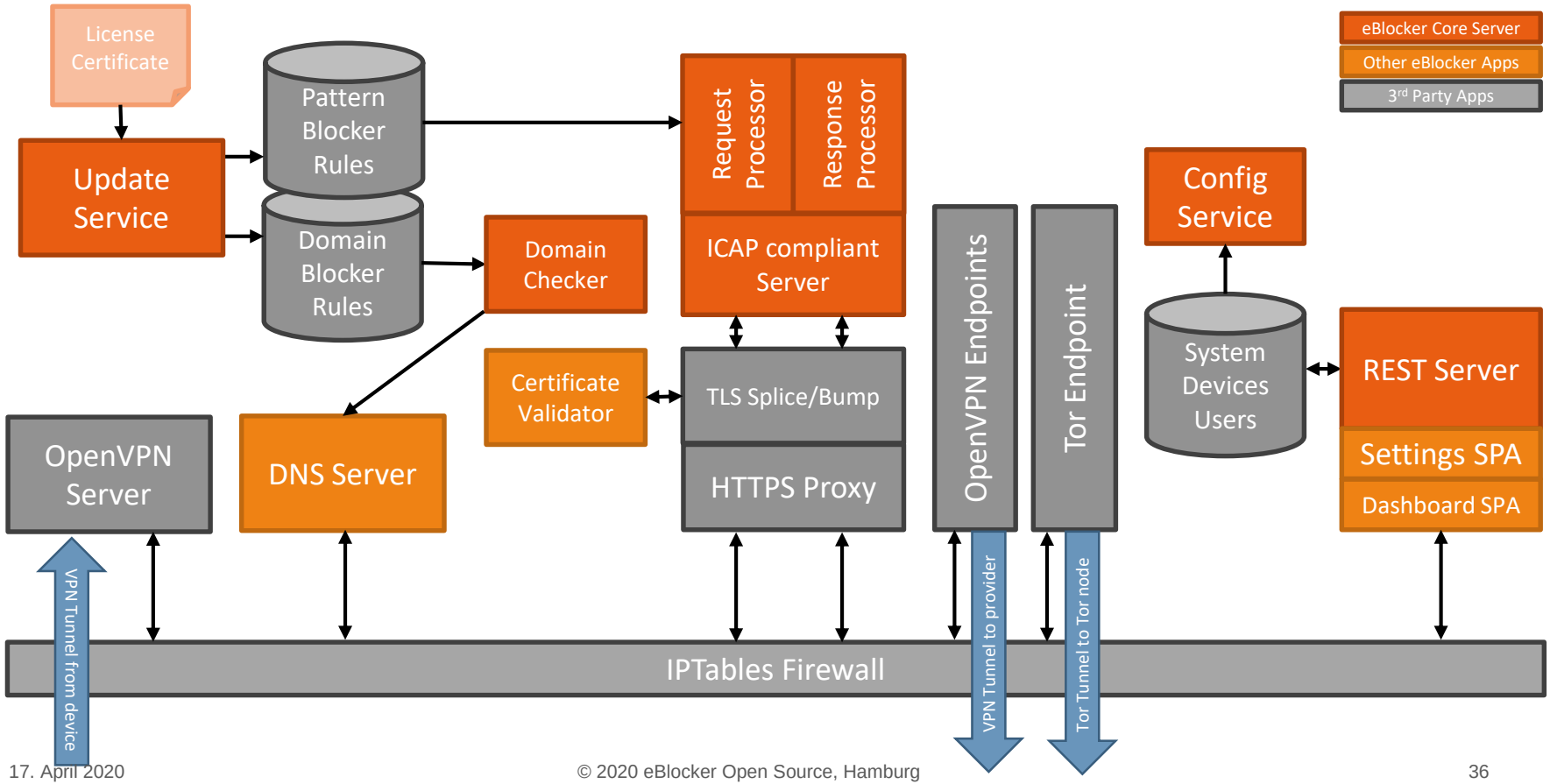




Schutz für unterwegs

- **Ziel:** Wenn der Nutzer nicht im Heimnetz ist, soll er geschützt werden
- **Lösung:** Gesamter mobiler Traffic wird zum eBlocker ins Heimnetz geschickt
- OpenVPN Server läuft auf eBlocker
- „eBlocker Cloud“ für dynamischen DNS-Dienst
- **Herausforderungen für DAUs:**
  - Mobiles Gerät muss OpenVPN-Client Installation zulassen
  - Port Forwarding 1194 UDP vom Router zum eBlocker

eBlocker Core Architecture



Fazit

Personenbezogene Massenüberwachung
und Profilbildung im Internet ist heute „normal“

Schutz vor Profilbildung und Anonymität ist
technisch komplex

Open Source Software + Raspberry Pi
hilft kostenlos

Switch on Privacy.

Vielen Dank!

eBlocker.org
Raspberry Pi Images

github.com/eblocker
Open Source Code

voluntary@eBlocker.org
Kontakt für Unterstützer