A QUIC Look at Web Tracking

Erik Sy, Christian Burkert, Hannes Federrath, Matthias Fischer
Transport protocols should not allow distinguishing Alice and Bob as the sender of a message.
Introduction to the QUIC Transport Protocol

- QUIC is going to replace TLS over TCP in HTTP/3

- Improves problems of TLS over TCP
  - Protocol Entrenchment
  - Implementation Entrenchment
  - Handshake Delay
  - Head-of-line Blocking
  - Mobility

- Google’s QUIC protocol is already widely deployed on the Internet
  - Accounts for 7% of global Internet traffic
  - Supported by Google Chrome (approx. 60% browser market share)
Source-address token speed up the validation of the client’s IP address in subsequent connections between the same peers.
Tracking via QUIC’s Server Config

- QUIC’s server config contains a public key used to bootstrap the cryptographic connection establishment

- Client reuses server config across different connections

- Tracking feasible if server distributes unique server configs/ server config identifiers to its clients
QUIC’s Connection Establishment

**a) Initial Handshake**

- Inchoate ClientHello
- Config, Token_1, ...
- ClientHello (ConfigID, Token_1, ...)
- Encrypted Request
- ServerHello (Token_2, ...)
- Encrypted Response

**b) Subsequent Handshake**

- ClientHello (ConfigID, Token_2, ...)
- Encrypted Request
- ServerHello (Token_3, ...)
- Encrypted Response
Opportunities and Limitations of Tracking via QUIC

- Independent of common tracking approaches like IP addresses, HTTP cookies and browser fingerprinting

- Opportunities compared to browser fingerprinting
  - Client cannot detect tracking via QUIC
  - Lower consumption of bandwidth and computational resources
  - Faster unique identification of a user
    - Relevant in the context of real-time bidding

- Limitations
  - Browser restarts terminate a tracking period
  - QUIC configuration of a browser
    - Lifetime of token and server configs
    - Feasibility of third-party tracking
Experiments to Test Browsers’ Default QUIC Configuration

- Measurement of QUIC’s Token lifetime within popular browsers
  - Maximum delay between two website visits for which the browser still attempts to establish the new connection with a cached Token

- Investigating the feasibility of third-party tracking via QUIC by comparing Tokens observed in both connections with T

Diagram:
- User
- Website A (incl. T)
- Third-party T
- Website B (incl. T)

Legend:
- Green dotted line: Loading Website A
- Blue dashed line: Loading Website B
## Summary on the Browser’s Default QUIC Configuration

<table>
<thead>
<tr>
<th>Browser</th>
<th>Lifetime of Token and Server Config</th>
<th>Third-party Tracking</th>
</tr>
</thead>
<tbody>
<tr>
<td>Chrome</td>
<td>unrestricted*</td>
<td>viable</td>
</tr>
<tr>
<td>Opera</td>
<td>unrestricted*</td>
<td>viable</td>
</tr>
<tr>
<td>Chromium</td>
<td>unrestricted*</td>
<td>viable</td>
</tr>
<tr>
<td>Chrome (mobile)</td>
<td>unrestricted*</td>
<td>viable</td>
</tr>
</tbody>
</table>

* evaluated for at least 11 days
Countermeasures

- Connection establishments based on public key cryptography require mechanisms to assure that public keys are not unique per user.

- Browser vendors should align tracking via QUIC with HTTP cookie policies
  - Preventing a bypass of HTTP cookie policies

- Limiting the lifetime of cached QUIC data to achieve an effective privacy protection

- Disabling third-party tracking via QUIC by limiting the reuse of third-party QUIC state only for revisits to the same first party.
Responses from Google

- “The 'cookie-like' mechanisms in QUIC are largely equivalent to the cookie handling in HTTP and thus do not substantially change the privacy posture of the browser.”
  - Only true, if tracking via HTTP cookies is unrestricted.

- “Blink (and hence the named browsers) implement TTL checking and additionally enforce a maximum TTL lifetime of one week.”
  - Browsers aim to restrict feasible tracking periods to seven days.
Future Work

- Privacy-friendly validation Token approving only a previously established connection between peers
  - Concept can be similar to “Privacy Pass: Bypassing Internet Challenges Anonymously” (PETS 2018)

- Design of a mechanism to detect servers issuing large numbers of public keys per epoch
  - Concept can be combination of Certificate Transparency logs and Online Certificate Status Protocol (OCSP)
  - Can be applied to Encrypted Server Name Indication (ESNI) for TLS 1.3
Conclusion

- QUIC combines great features with new privacy risks
- Tracking via QUIC is stealthy, fast and allows a unique user identification by third-party trackers
- Presented tracking mechanisms affect a huge user base and effective mitigations by browser vendors are not in sight
Thank you

Questions and Answers

E-mail: PETS@erik-sy.de
Slides: https://erik-sy.de/pets2019