WebSocket Extension to disable masking

draft-damjanovic-websockets-nomasking-02

Dragana Damjanovic

ALLDISPATCH

119 IETF, March 2024

WebSocket masking – what is it?

- Change the WebSocket payload on the wire:
  - Payload of all frames sent from the client to the server
  - Transmitted_payload = original_payload XOR masking_key
    - Additional processing
  - Masking-key is carried inside the frame header adding 4 extra bytes to each header.
    - Additional bytes send on the wire per frame

WebSocket masking – why?

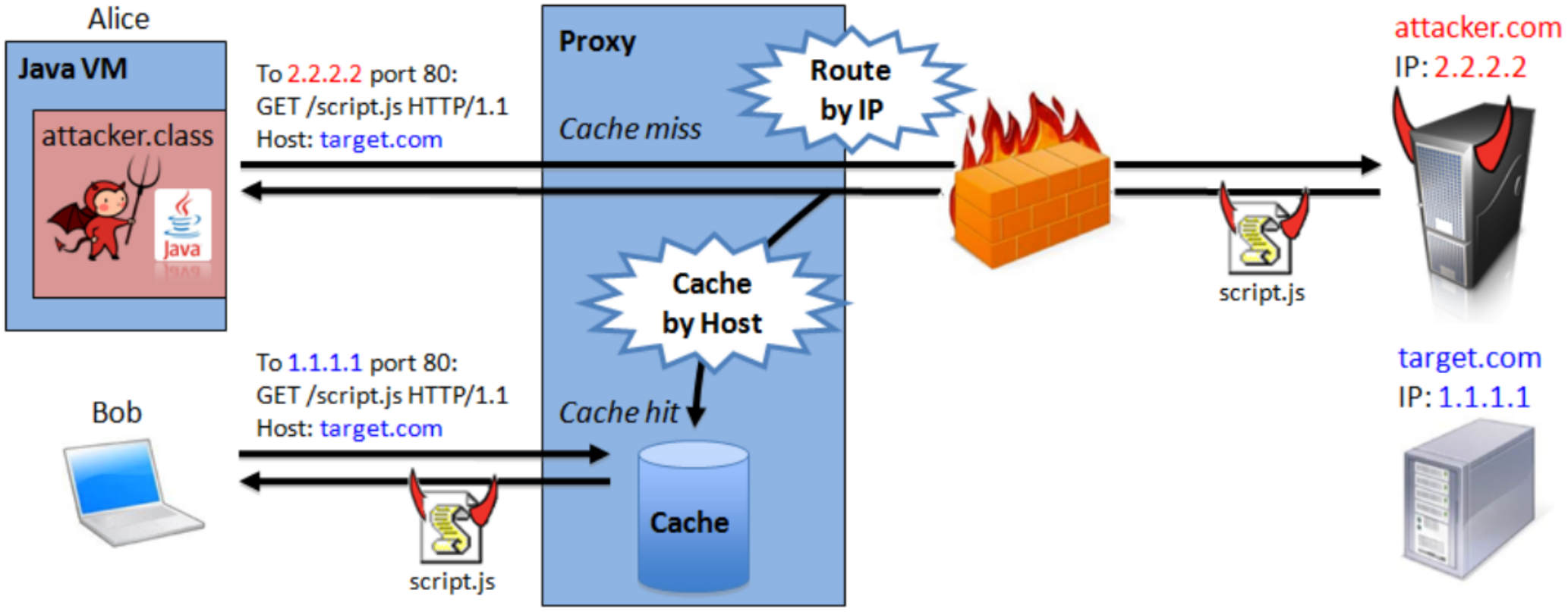

Fig. 2. Cache poisoning attack

- Huang, L-S., Chen, E., Barth, A., Rescorla, E., and C. Jackson, "Talking to Yourself for Fun and Profit", 2011.

WebSocket masking – why?

- End-to-end encryption does not help
  - The attacker has the TLS session keys because it owns both peers.
  - It can craft WebSocket messages that after encryption look like a cleartext HTTP request.
  - The transparent caching proxy could potentially cache that content.

Do we still need masking?

- The experiment was conducted in 2011.
  - 8 out of 47 338 paths were vulnerable to the attack.
- The HTTP upgrade mechanism was new at that time.

Do we still need masking?

When WSS is used on port 443(any port other than 80):

- The caching proxies need to cache content from the port that usually carries encrypted content.
  - These transparent caching proxies cache cleartext content,
  - Therefore, most probably only content on port 80 will be cached.

Do we still need masking?

Clients need to access website using unsecure HTTP, otherwise the poisoned cache will not be hit:

- There is less content that is not using HTTPS
- It is less likely that the client connects to otherwise HTTPS capable site using insecure HTTP because of existing techniques, e.g. HSTS, the https-first efforts.

Proposal - WebSocket extension “no-masking”

- Send by the client only on secure connection (excluding on port 80).

Client:

GET /chat HTTP/1.1
Host: server.example.com
Upgrade: websocket
Connection: Upgrade
Sec-WebSocket-Key: dGhlIHNhbXBsZSBub25jZQ==
Origin: https://example.com
Sec-WebSocket-Protocol: chat, superchat
Sec-WebSocket-Version: 13
Sec-WebSocket-Extensions: no-masking

Server:

HTTP/1.1 101 Switching Protocols
Upgrade: websocket
Connection: Upgrade
Sec-WebSocket-Accept: s3pPLMBiTxaQ9kYGzzhZRbK+xOo=
Sec-WebSocket-Protocol: chat
Sec-WebSocket-Extensions: no-masking

Proposal - WebSocket extension “no-masking”

- Frames send from the client to the server:
  - Without masking_key
  - Without payload masking
- Frames from the server to the client are unchanged.

- Seeking an appropriate Working Group?