Can You Tell Me the Time?

Security Implications of the Server-Timing Header

Vik Vanderlinden, Wouter Joosen, Mathy Vanhoef

KU LEUVEN DistriNet

Overview

- Timing Attacks
  - Goal
  - Data analyses
- *Server-Timing* header
- Web Prevalence
  - Adoption
  - Attribution
  - Other headers
- Attack Techniques
  - Threat model
  - Experimental setup
  - Data analyses
  - Results & defenses

Timing Attacks

The goal: Measure time to leak secrets

Timing Attacks

Calculate `0 * 47 ..`

Calculate `46 * 47 ..`

Calculation time == information leak: e.g. operand

- Remote: over internet
  - Noise (δ) due to network & middleboxes[4], [7]
  - $T = RTT = (t + \delta)_{up} + (t + \delta)_{exec} + (t + \delta)_{down}$

The goal: Measure time to leak secrets

Timing Attacks

Calculate `0 * 47 .. = 0`

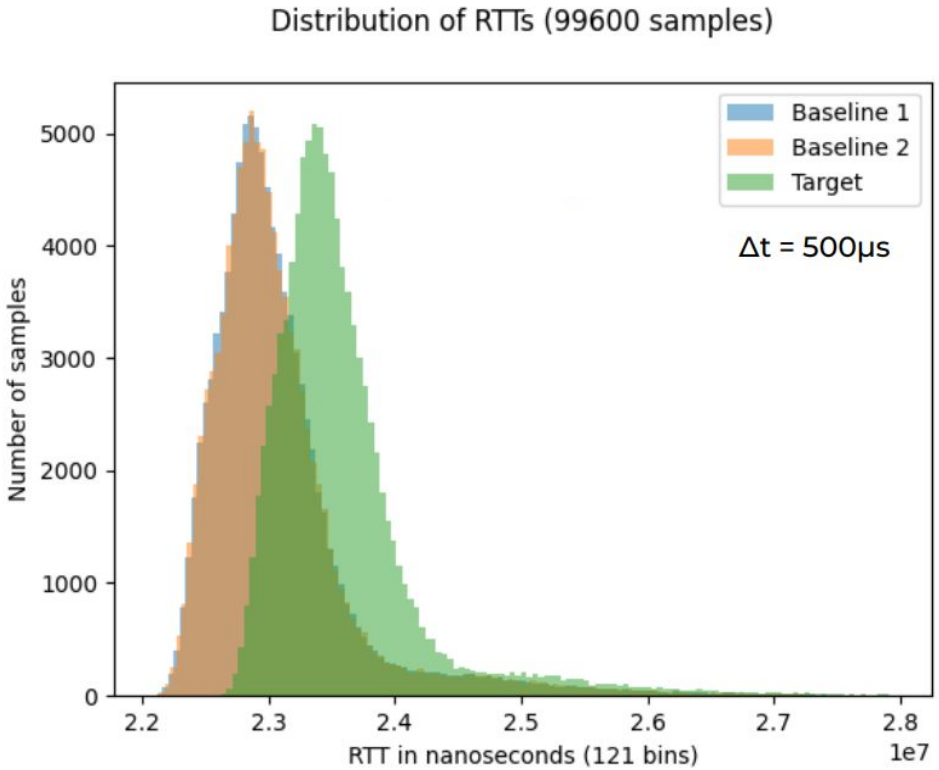
Calculate `46 * 47 ..... = 2 162`

Calculation time == information leak: e.g. operand

- Remote: over internet
  - Noise (δ) due to network & middleboxes[4], [7]
  - $T = RTT = (t + \delta)_{up} + (t + \delta)_{exec} + (t + \delta)_{down}$

How to analyze: Use *box test*[1]

Timing Attacks



Two baselines? Sanity check

(left peak) /

(right peak) /ad → ad iff 20-30 yo

→ 404 otherwise

⇒ Δt reveals user age

KU LEUVEN DistriNet

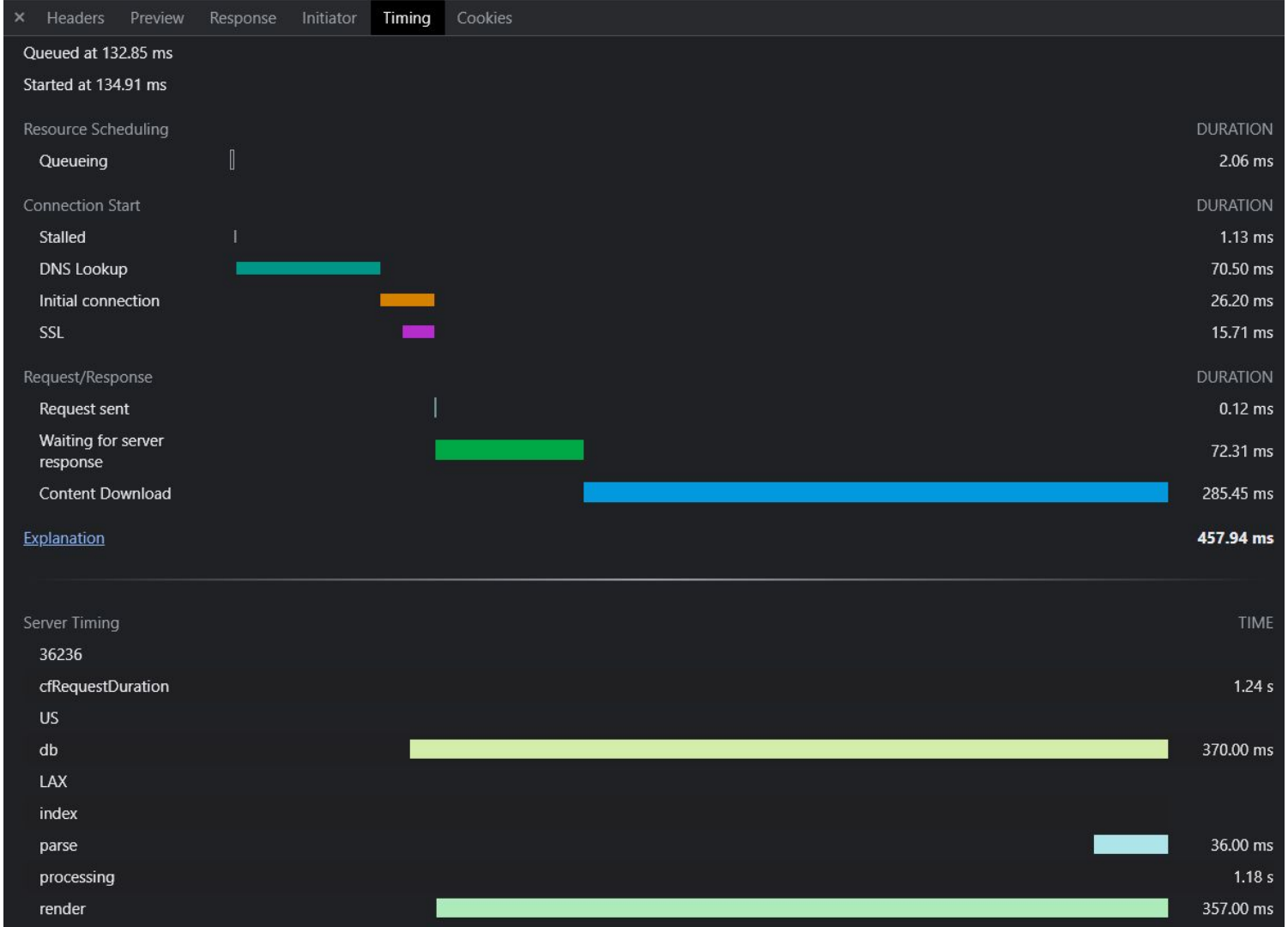
Server-Timing header

Server-Timing header[2] exposes timing info

- W3C working draft
- Performance & debugging data

```
server-timing:
    processing;dur=1175,db;dur=370,parse;dur=36,
    render;dur=357,asn;desc="36236",edge;desc="LAX",country;desc="US",
    theme;desc="Conversion Optimizations Updates 1/12/2023",pageType;desc="index"
server-timing: cfRequestDuration;dur=1241.999865
```

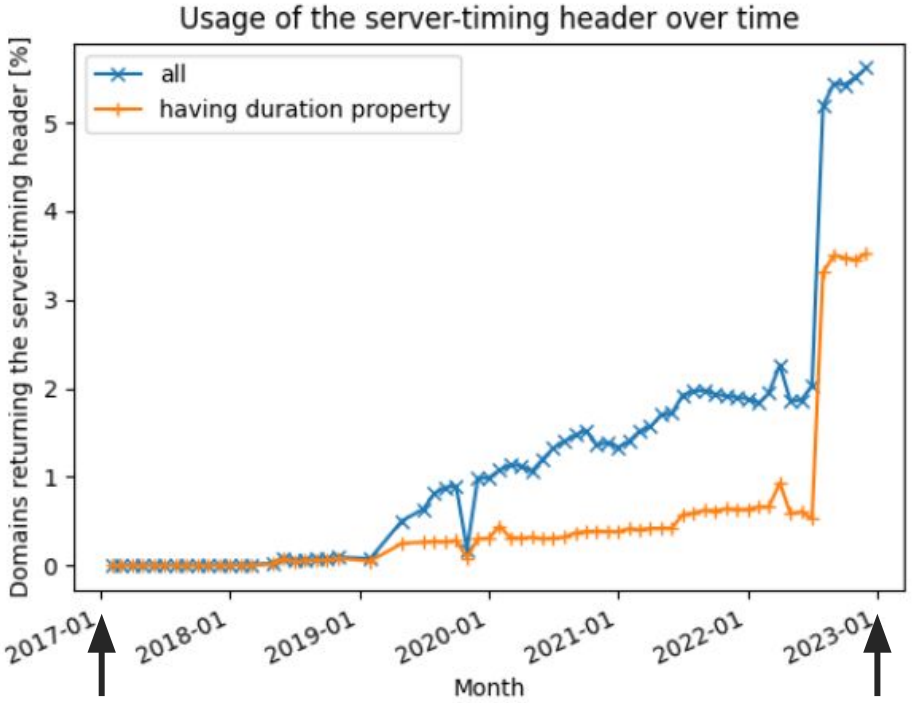
- ‘dur’ property: 1 ms accuracy[3]


Headers
Preview
Response
Initiator
Timing
Cookies
Queued at 132.85 ms
Started at 134.91 ms
Resource Scheduling
DURATION
Queueing
2.06 ms
Connection Start
DURATION
Stalled
1.13 ms
DNS Lookup
70.50 ms
Initial connection
26.20 ms
SSL
15.71 ms
Request/Response
DURATION
Request sent
0.12 ms
Waiting for server response
72.31 ms
Content Download
285.45 ms
Explanation
457.94 ms
Server Timing
TIME
36236
cfRequestDuration
1.24 s
US
db
370.00 ms
LAX
index
parse
36.00 ms
processing
1.18 s
render
357.00 ms

Web Prevalence

Increasing **adoption** in recent years/months

Web Prevalence



2017 **2023**

Queried https://httparchive.org/ data

rank	# scanned	# header (%)	# dur (%)
1k	695	6.04%	5.47%
…	…	…	…
100M	10.192M	5.44%	3.50%

Main **attribution** to one e-commerce vendor

Web Prevalence

- Crawl 100k sites
  - Forms
  - Same-origin action
  - 73.12% dynamic

server	# header (%)	# with ‘dur’ property (%)
cloudflare	3.02%	3.01%
pepyaka	1.81%	**0.00%**
nginx	0.27%	0.25%
apache	0.07%	0.05%
cloudfront	0.02%	0.02%
all	5.44%	3.50%

Other headers also leak timing info

Web Prevalence

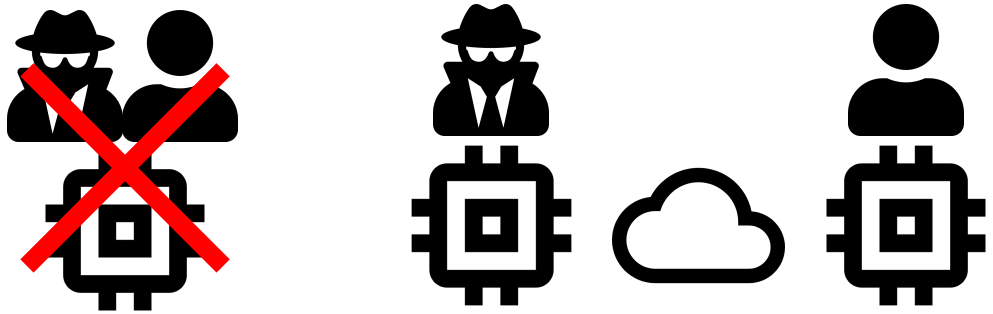
regex	# sites (out of 10 194 945)	%
(run)?-?_?time?(ing)?	894 314	**8.8%**
(run)?-?_?time	341 048	**3.3%**
run-?_?time	195 091	**1.9%**

Attack Techniques

Threat model

Attack Techniques

- Geography



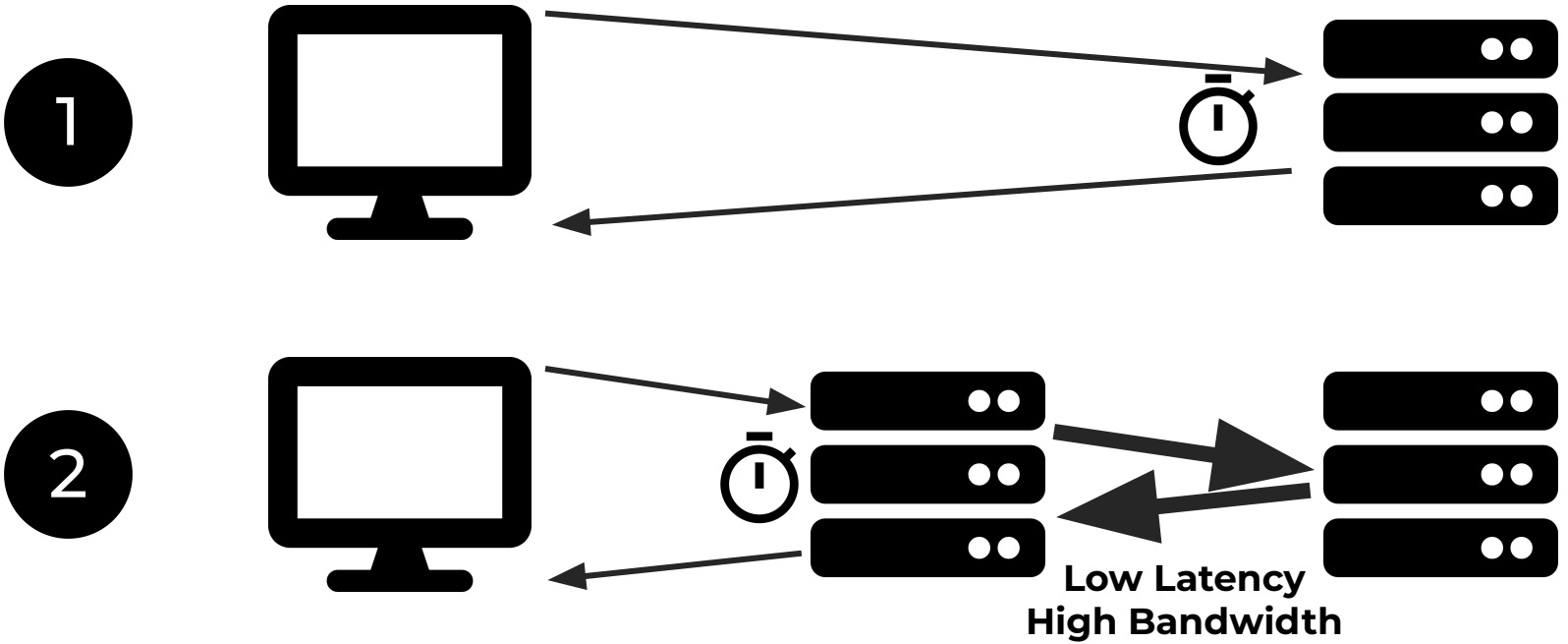
- Random device (e.g. malware/stored XSS)
- No XS attack
  - *timing-allow-origin*[2],[5]
  - CORS *access-control-expose-headers*[6]

Experimental setup

Attack Techniques

$$T = RTT = (t + \delta)_{up} + (t + \delta)_{exec} + (t + \delta)_{down}$$

$$T = \cancel{\quad} \quad (t + \delta)_{exec} \quad \cancel{\quad}$$



Experimental setup

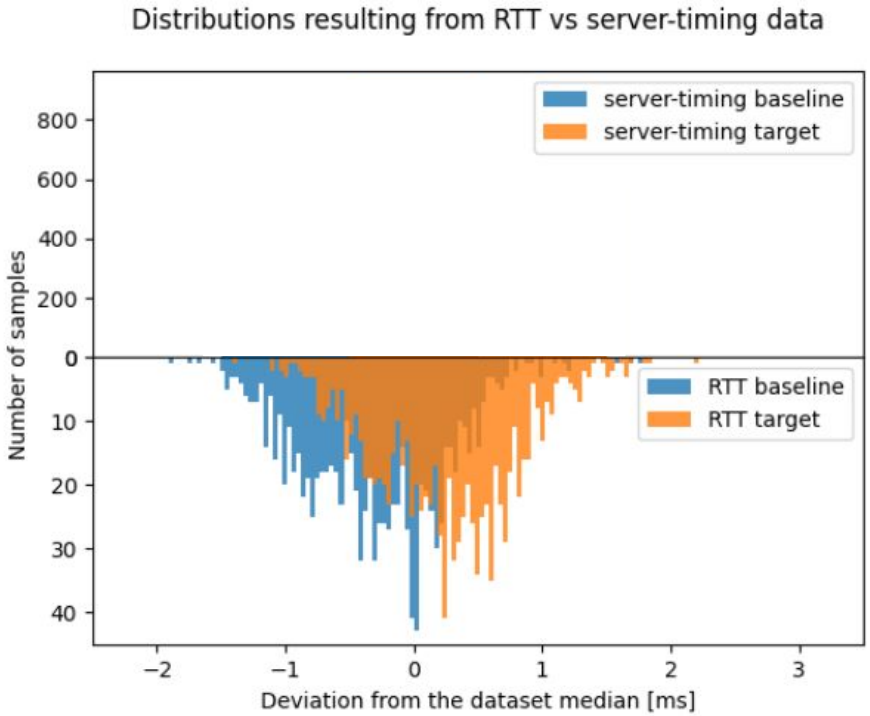
Attack Techniques

- Client
  - University cloud
  - Home network
- *server-timing* header
  - Nginx `$request_time`
  - Full request time
  - No specific ‘dur’ tested
- Server
  - Google Cloud
  - Nginx
- Configurations
  - EU → EU,
    EU → US
  - EU → EU → EU,
    EU → EU → US,
    EU → US → US

Data analyses

Attack Techniques

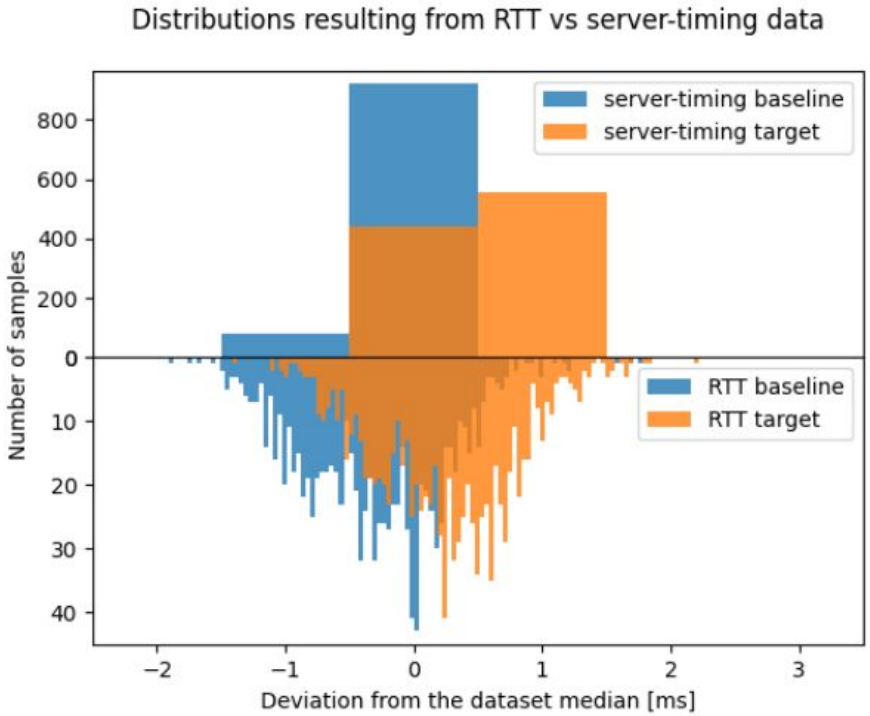
- No box test
- χ^2-contingency test
  - Best of 4 statistics tested
- Classify 95% correctly



Data analyses

Attack Techniques

- No box test
- χ^2-contingency test
  - Best of 4 statistics tested
- Classify 95% correctly



Results

Attack Techniques

- Proxied attack

Attack + Test	Network	Proxy	$5\mu s$	$10\mu s$	$20\mu s$	$50\mu s$	$100\mu s$	$200\mu s$	$500\mu s$	$1ms$	$2ms$	$5ms$
standard RTT box test	university	EU → EU	-	-	-	-	10 000	10 000	500	500	20	20
		EU → US	-	-	-	-	10 000	10 000	2 000	200	10	10
		US → US	-	-	-	-	-	10 000	2 000	500	5 000	50
	residential	EU → EU	-	-	-	-	-	-	5 000	5 000	200	20
		EU → US	-	-	-	-	-	-	-	500	200	20
		US → US	-	-	-	-	-	-	2 000	5 000	1 000	20
server-timing $\mathcal{X}^2$*-contingency*	university	EU → EU	-	-	10 000	10 000	1 000	200	50	10	5	5
		EU → US	-	-	-	10 000	5 000	500	100	20	10	5
		US → US	-	-	-	5 000	5 000	500	50	10	5	5
	residential	EU → EU	-	-	-	-	1 000	-	50	10	5	5
		EU → US	-	-	-	-	5 000	500	100	50	10	5
		US → US	-	-	-	-	-	10 000	50	10	5	5

- Direct attack (similar) → paper

Defenses

Attack Techniques

Ideally: Don’t use *Server-Timing* in production

Alternative: Educated decisions to expose

Full solution: Don’t use sensitive timing values

Partial solutions: Round timing value, pad sensitive operations

Responsible Disclosure

- Contacted authors W3C standard
- Contacted Shopify

No responses yet

Conclusion

- Server-Timing header enables timing attacks
- More sites start to use it, and already expose ‘dur’
- We explored a direct and proxied attack
- Use of the header leads to improved timing attack performance
- Awaiting responsible to our responsible disclosures

Can You Tell Me Your Time?

Vik Vanderlinden, Wouter Joosen, Mathy Vanhoef

imec-DistriNet, KU Leuven
Belgium

vik.vanderlinden@kuleuven.be / @vikvanderlinden

KU LEUVEN DistriNet

References

[1] S. A. Crosby, D. S. Wallach, and R. H. Riedi, “Opportunities and Limits of Remote Timing Attacks,” ACM Transactions on Information and System Security, vol. 12, no. 3, pp. 1–29, jan 2009. [Online]. Available: https://dl.acm.org/doi/10.1145/1455526.1455530

[2] C. Vazac and I. Grigorik, “Server timing: W3c working draft,” https://www.w3.org/TR/server-timing/, 2022.

[3] Y. Weiss, I. Grigorik, J. Simonsen, and J. Mann, “High resolution time: The domhighrestimestamp typedef,” https://www.w3.org/TR/hrtime-3/#dom-domhighrestimestamp, 2022.

[4] D. Brumley and D. Boneh, “Remote timing attacks are practical,” Computer Networks, vol. 48, no. 5, pp. 701–716, aug 2005. [Online]. Available: https://linkinghub.elsevier.com/retrieve/pii/S1389128605000125

[5] Y. Weiss and N. Rosenthal, “Resource timing: Timing-allow-origin response header,” https://www.w3.org/TR/resource-timing/#sec-timingallow-origin, 2022.

[6] whatwg/fetch contributors, “Fetch standard: Cors protocol,” https://fetch.spec.whatwg.org/#http-cors-protocol, 2023.

[7] H. Pucha, Y. Zhang, Z. M. Mao, and Y. C. Hu, “Understanding network delay changes caused by routing events,” ACM SIGMETRICS Performance Evaluation Review, vol. 35, no. 1, pp. 73–84, jun 2007. [Online]. Available: https://dl.acm.org/doi/10.1145/1269899.1254891