Triggering QUIC

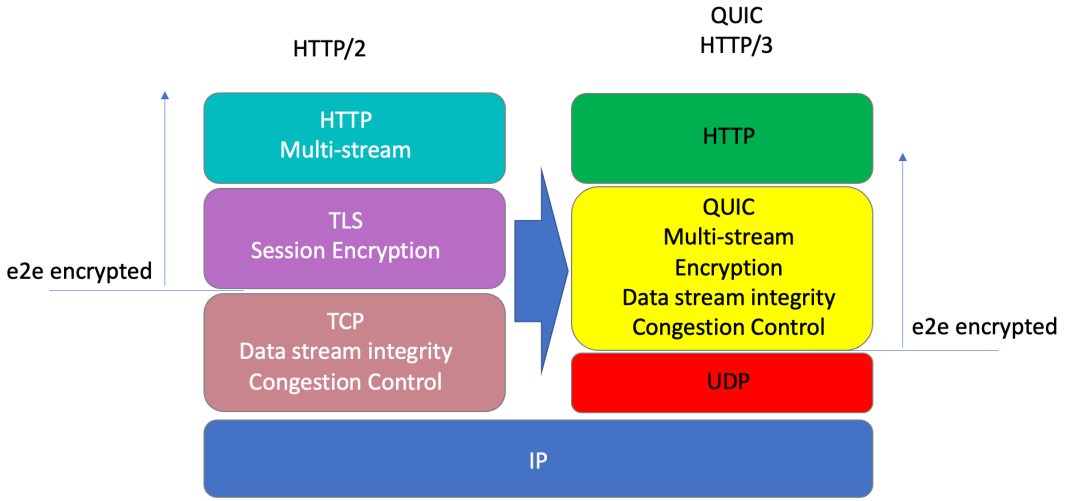
Geoff Huston AM

APNIC

July 2025

What's QUIC?

- An end-to-end encrypted transport protocol, providing more flexibility, faster connection setup, and a larger set of transport services than TCP
- Operates over UDP port 443



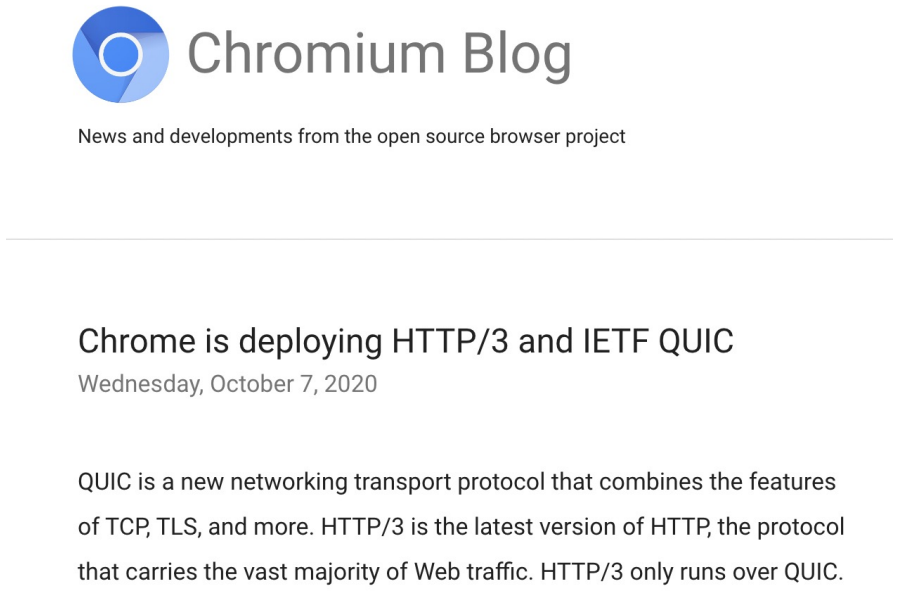
Triggering QUIC in HTTP

Method 1 - Use content-level Alt-Svc controls to trigger the client to use the QUIC transport protocol (if it can):

- Add **`Alt-Svc: h3=":443"`** to the HTML headers

Setting Expectations

- Chrome has a dominant share of browser instances - roughly, some 65%*
- Chrome has been supporting a switch to QUIC via the Alt-Svc directive since 2020



Chromium Blog

News and developments from the open source browser project

Chrome is deploying HTTP/3 and IETF QUIC

Wednesday, October 7, 2020

QUIC is a new networking transport protocol that combines the features of TCP, TLS, and more. HTTP/3 is the latest version of HTTP, the protocol that carries the vast majority of Web traffic. HTTP/3 only runs over QUIC.

* *Oberlo.com*

Setting Expectations

- Chrome has a dominant share of browser instances - roughly, some 65%*
- Chrome has been supporting a switch to QUIC via the Alt-Svc directive since 2020

So, we should expect up to 65÷ of clients will try to connect using QUIC if the server signals it supports QUIC – right?

Chrome is deploying HTTP/3 and IETF QUIC

Wednesday, October 7, 2020

QUIC is a new networking transport protocol that combines the features of TCP, TLS, and more. HTTP/3 is the latest version of HTTP, the protocol that carries the vast majority of Web traffic. HTTP/3 only runs over QUIC.

* *Oberlo.com*

The Alt-Svc Trigger

- This trigger is only effective when the client contacts this server for **the second time**
  - But HTTP/1.1 and HTTP/2.0 use session persistence to keep the original TCP/TLS session open, so the condition where a client needs to open a new connection is less likely to occur
  - The per-server Alt-Svc information is cached by the user for only 24 hours by default

The Alt-Svc Trigger

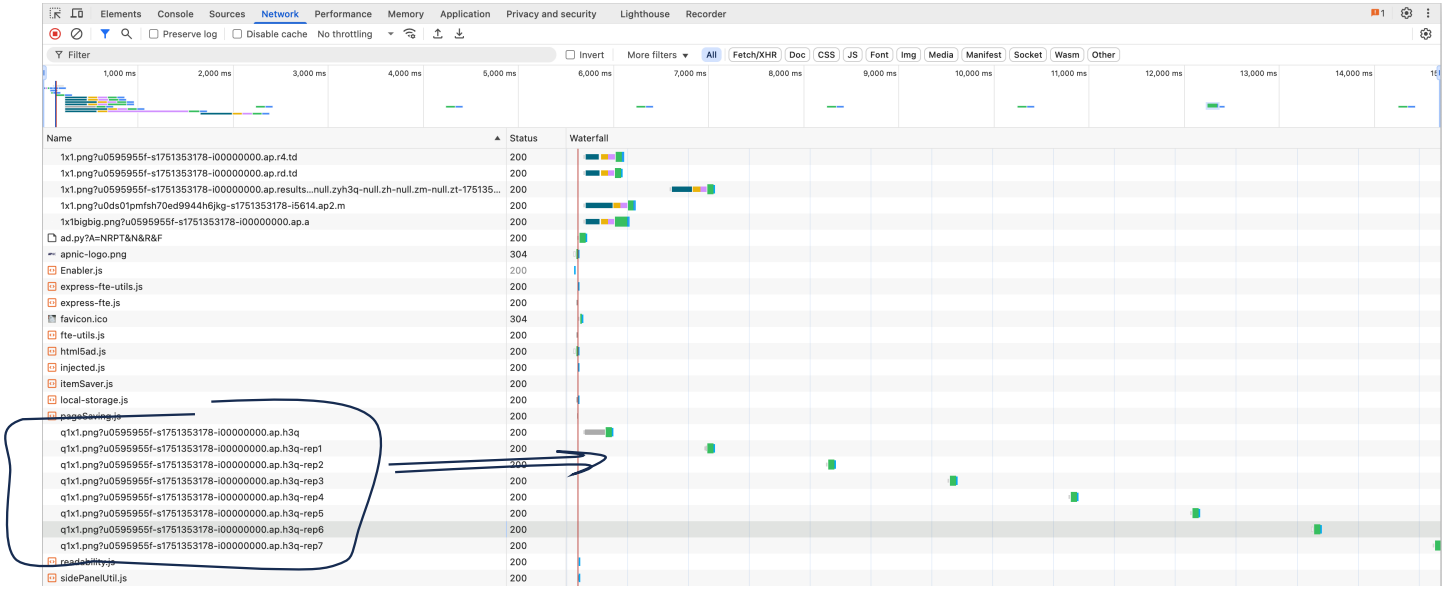
- This trigger is only effective when the client contacts this server for **the second time**
  - But HTTP/1.1 and HTTP/2.0 use session persistence to keep the TCP/TLS session open, so the condition connection is lost

So, QUIC use will only be visible when a server is visited by a client for a second time AFTER the keep-alive expires.

APNIC's measurement

We need to trigger the conditions of a second fetch in the measurement:

- Set the server keepalive time to 1 second
- Request the same web object a total of 8 times using 2 second intervals between requests



Triggering QUIC in HTTP

Method 2 - Use the DNS to trigger QUIC:

- Set up an HTTPS record for the service name, with value: **`alpn="h3"`**
- This allows Safari to use QUIC from the first access

Safari supports QUIC (using Method 2)

- Apple's Safari is now supporting QUIC, using an HTTPS query/response in the DNS, where the **`apln`** directive can specify the use of the HTTP/3 protocol to access this service
- QUIC can be triggered immediately (no wait for the second visit), so presumably, if the client performs a DNS HTTPS query, and the response indicates that the server supports QUIC, then the client should use QUIC for the connection

Setting Expectations

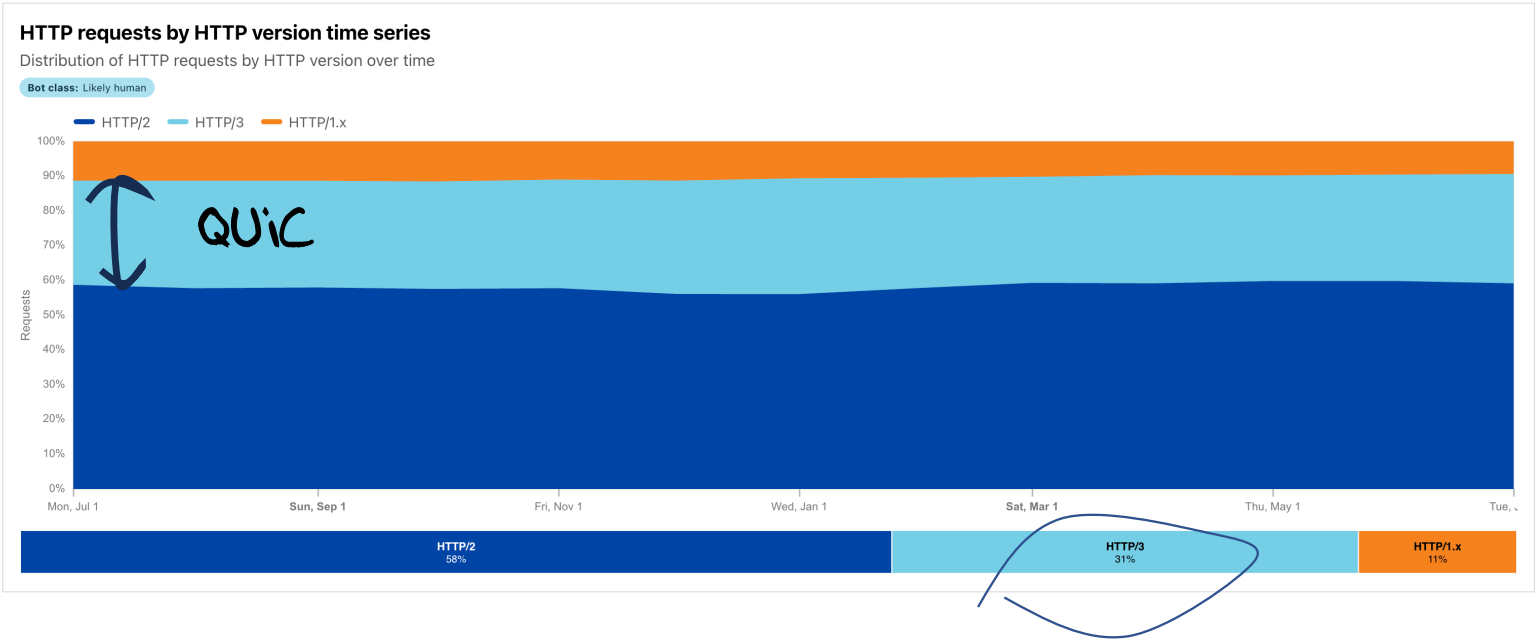
- Chrome has a dominant share of browser instances - roughly, some 65%*
- Apple Safari is now supporting QUIC, using the DNS HTTPS trigger
- So, a QUIC-aware server platform should be seeing **up to** 85% of its sessions using QUIC
  - This figure is probably not achievable as the content level control requires some precise conditions for the “second” visit:
    - long enough between visits for the session keepalive timer to expire
    - Short enough such that the local cache of server capabilities has not expired

** https://gs.statcounter.com/browser-market-share*

Cloudflare's Numbers - 31%

12 Month Time Series

Cloudflare Radar



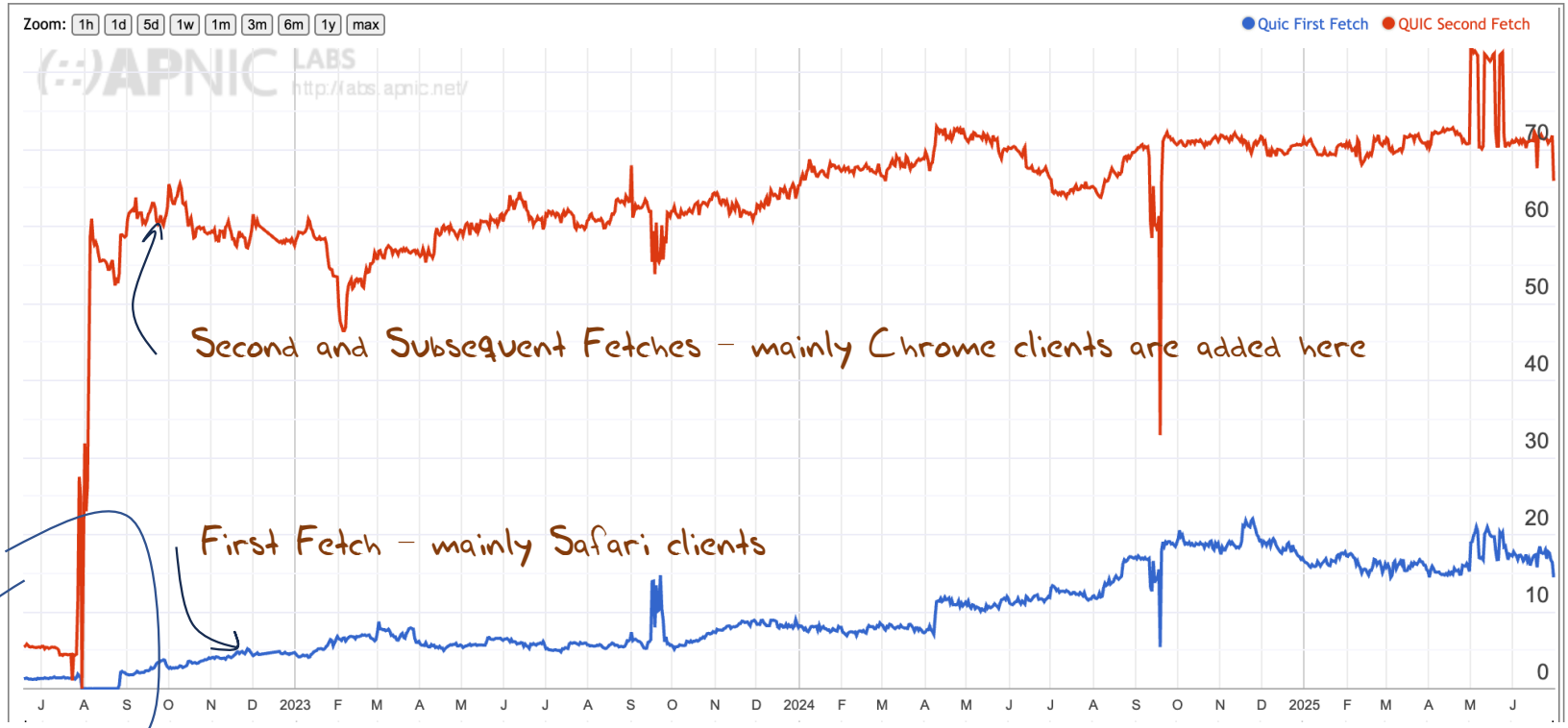
Cloudflare's Numbers - 31%

12 Month Time Series

Cloudflare Radar

That result is less than half of the maximum possible result if this is just alt-svc directive being used to trigger QUIC

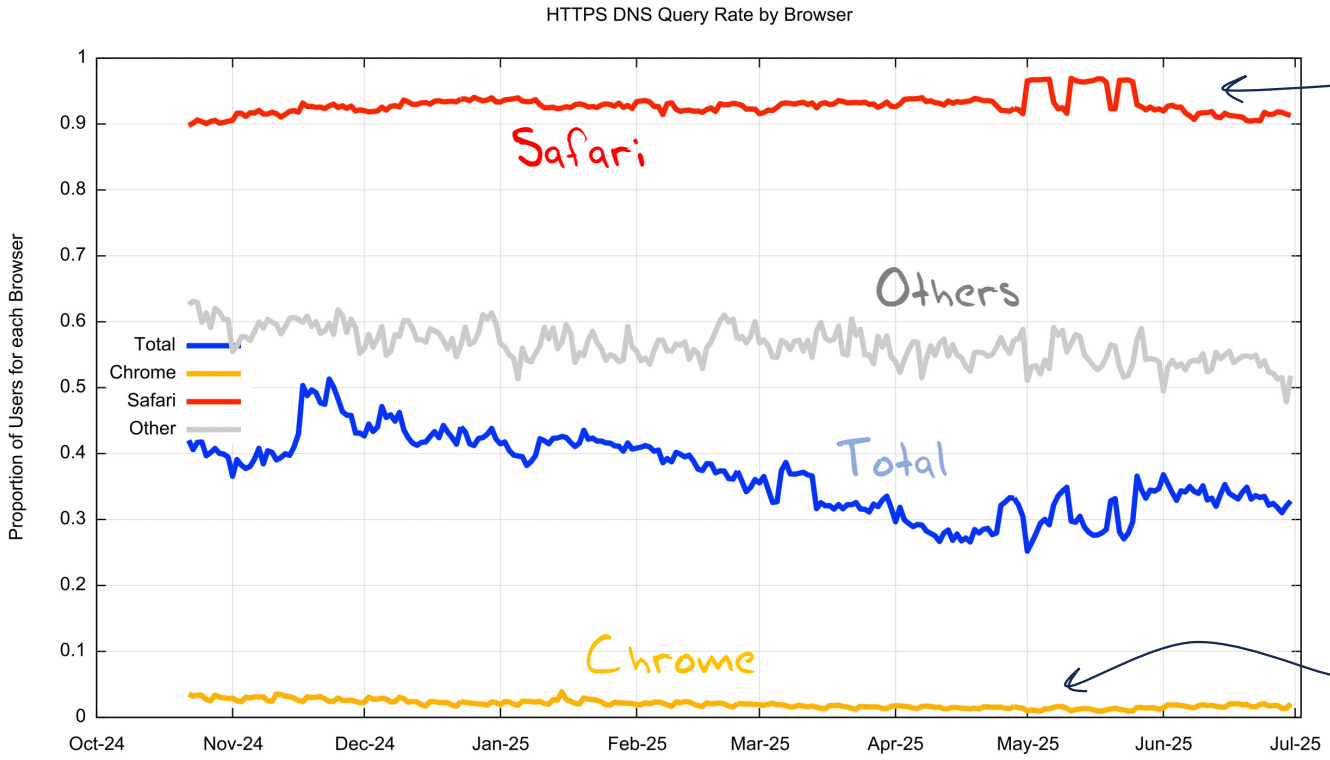
APNIC's Numbers - 70%



Playing with keepalive parameters!

Method 2 - DNS HTTPS Query Rate

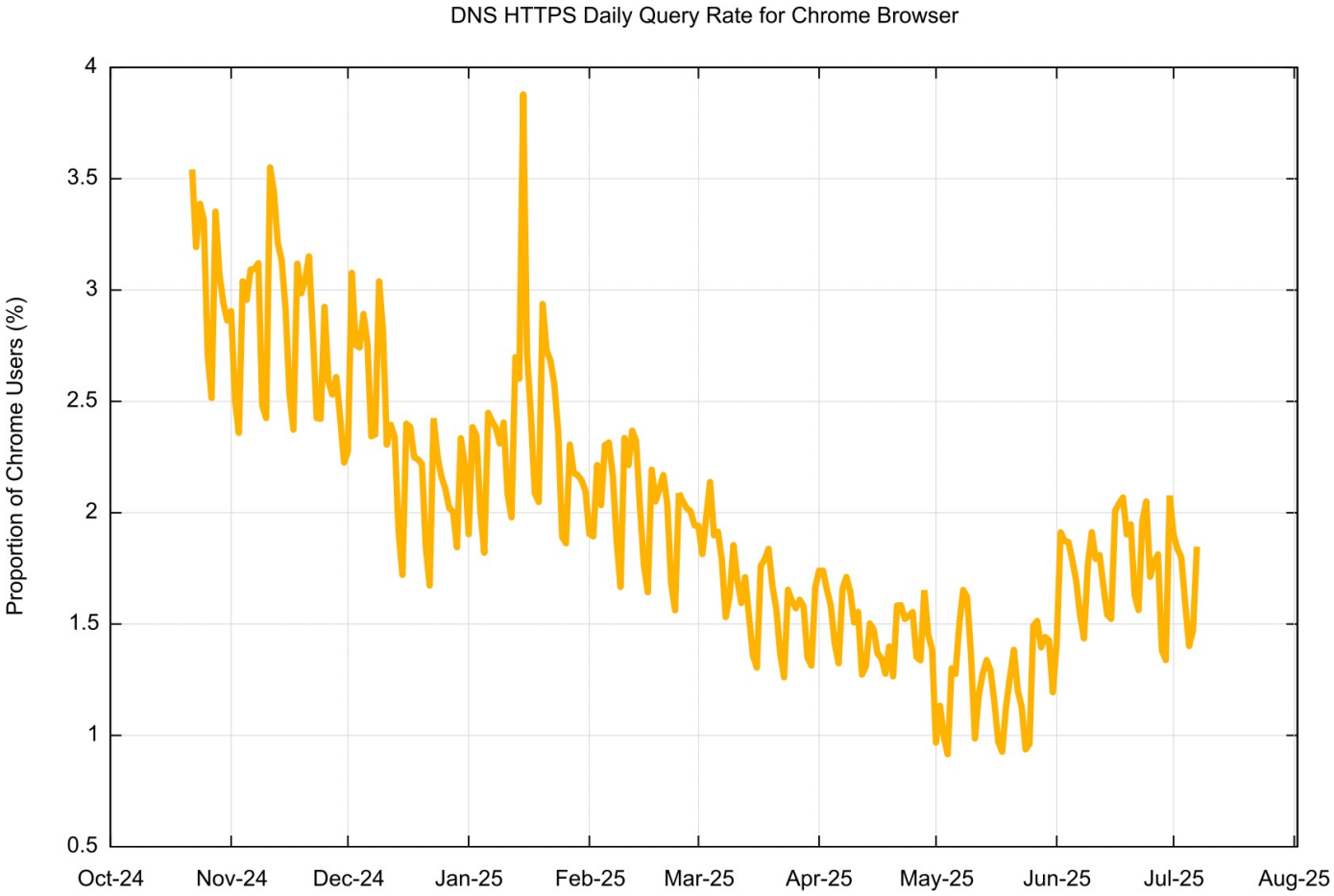
How many users are generating DNS HTTPS Queries?



Most Safari browsers consistently query for a DNS HTTPS record

Around 3% of Chrome browsers query for the HTTPS record

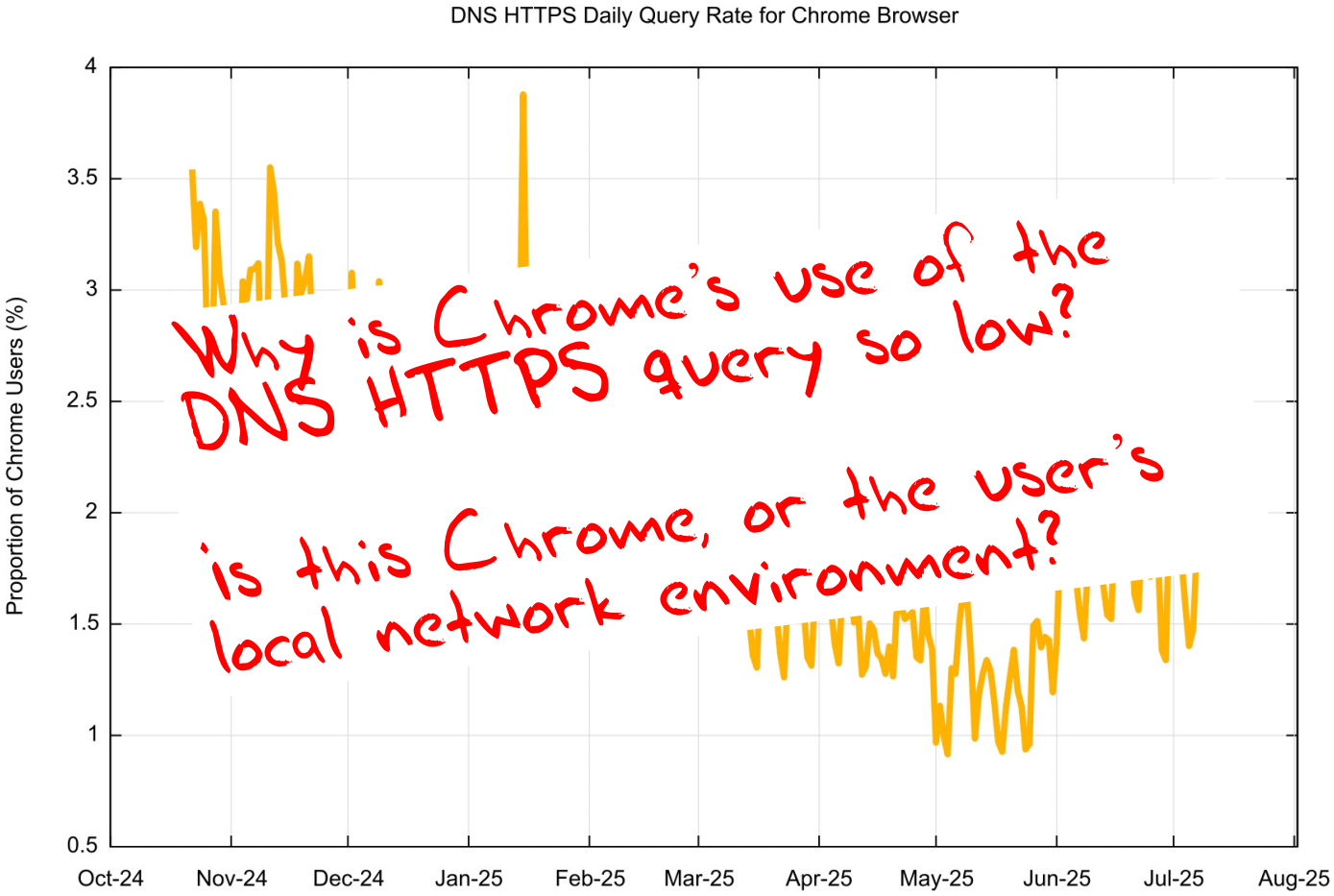
Chrome Browser HTTPS Query Rate



There is a strong weekly pattern in this data where weekend query rates are lower than weekday query rates

Query rates halved between October 2024 and May 2025, then rose across June 2025

Chrome Browser HTTPS Query Rate



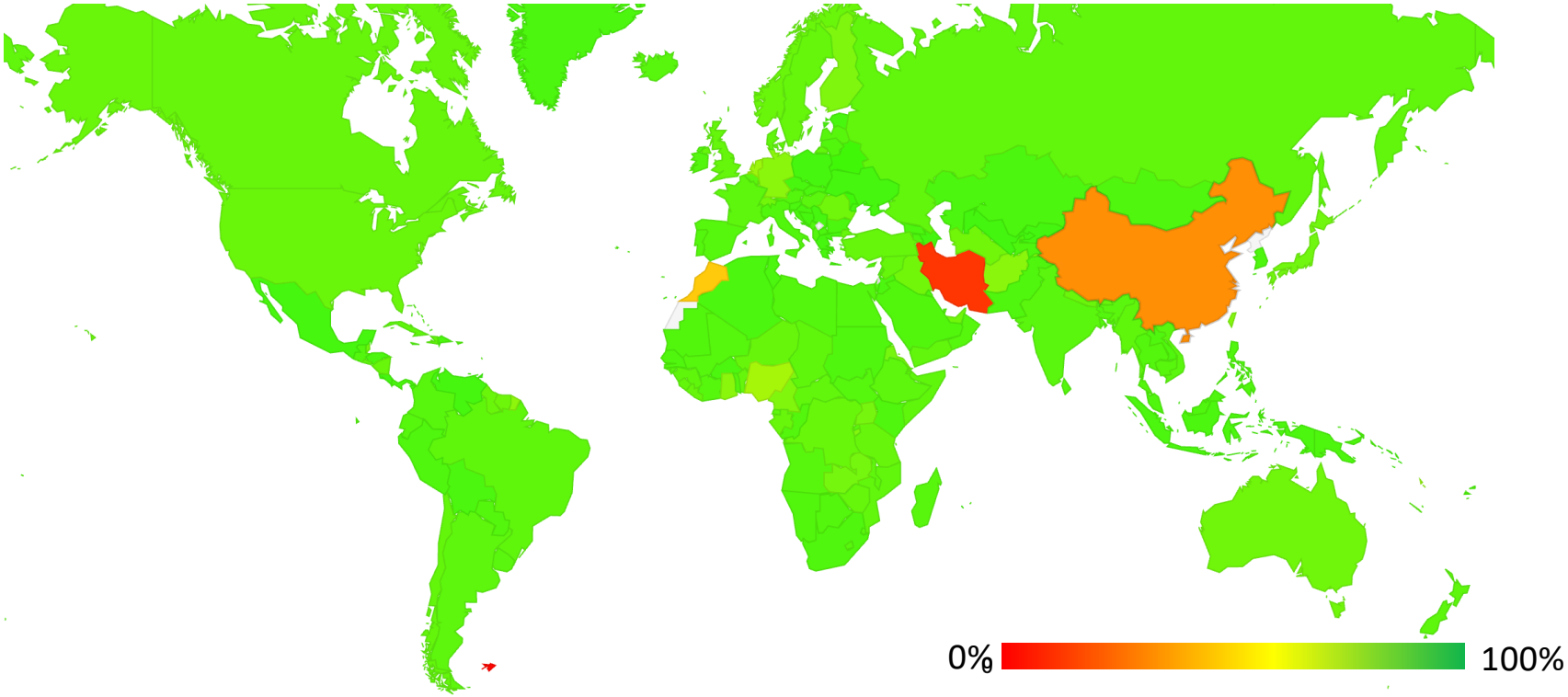
There is a strong weekly pattern in this data where weekend query rates are lower than weekday query rates

Query rates halved between October 2024 and May 2025, then rose across June 2025

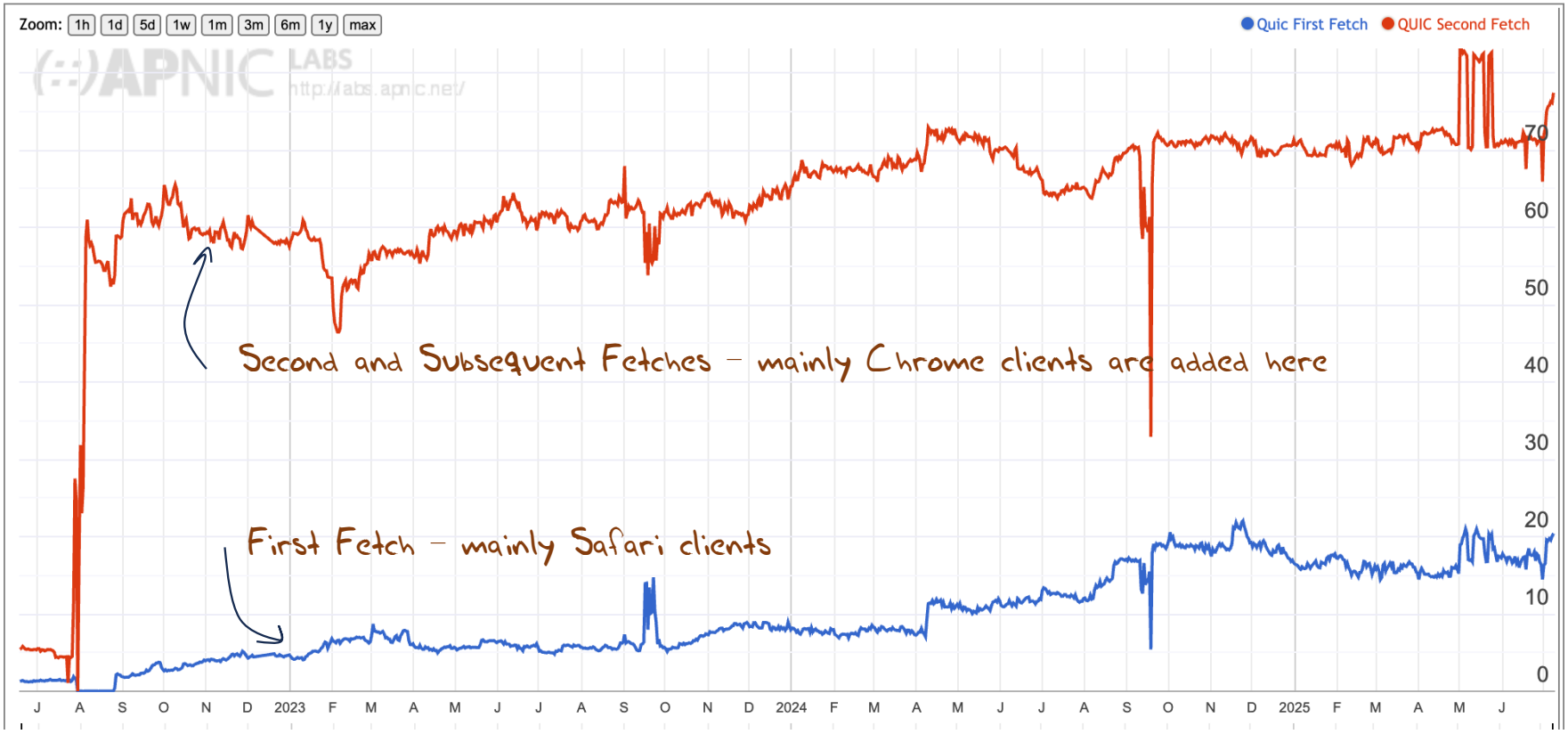
QUIC Use

- If QUIC access is supported by the current releases by both the major browsers, then we should see a high QUIC use rate when the ability to use QUIC is signaled by both methods (alt-svc and DNS HTTPS)
- What do we see?

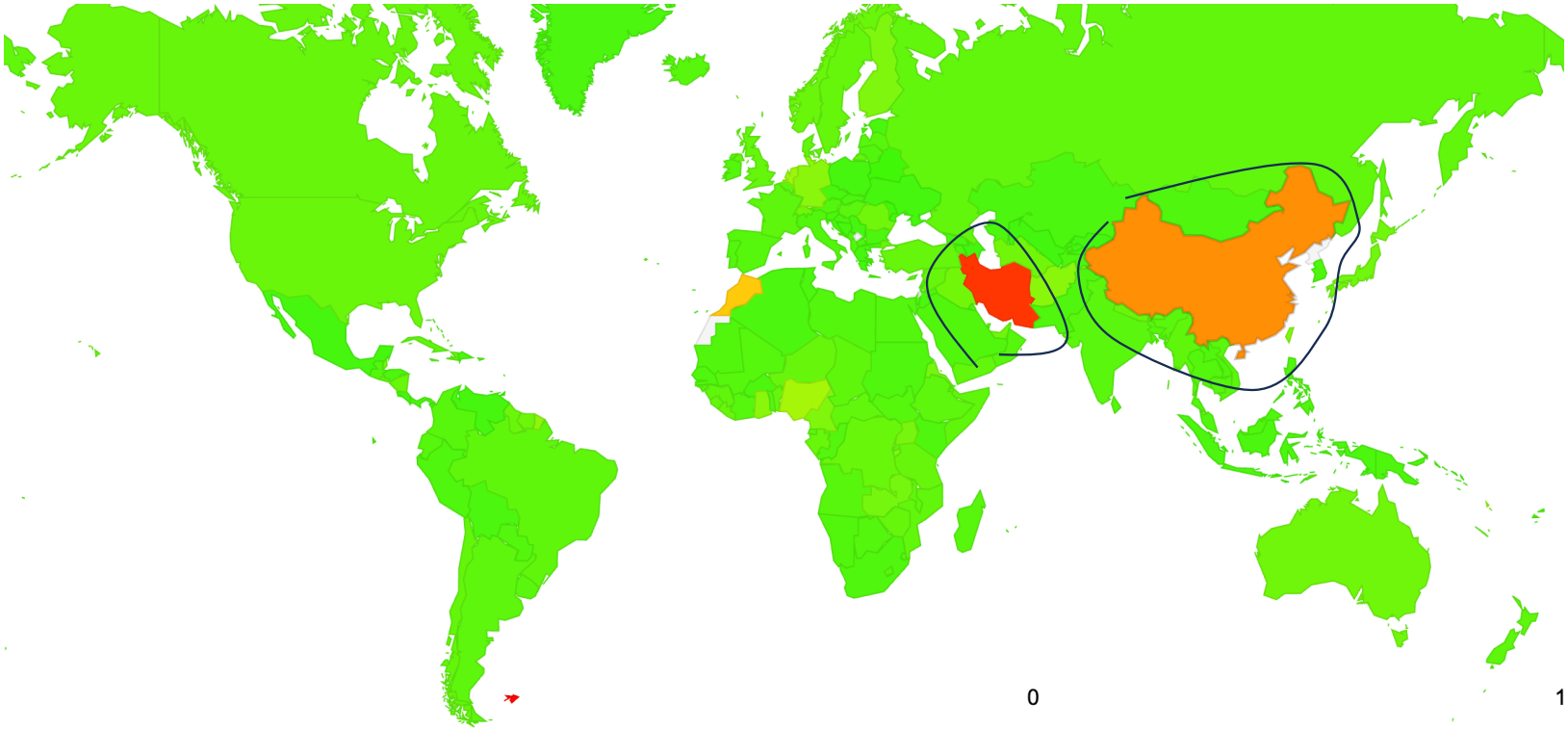
Global QUIC Use



Global QUIC Use



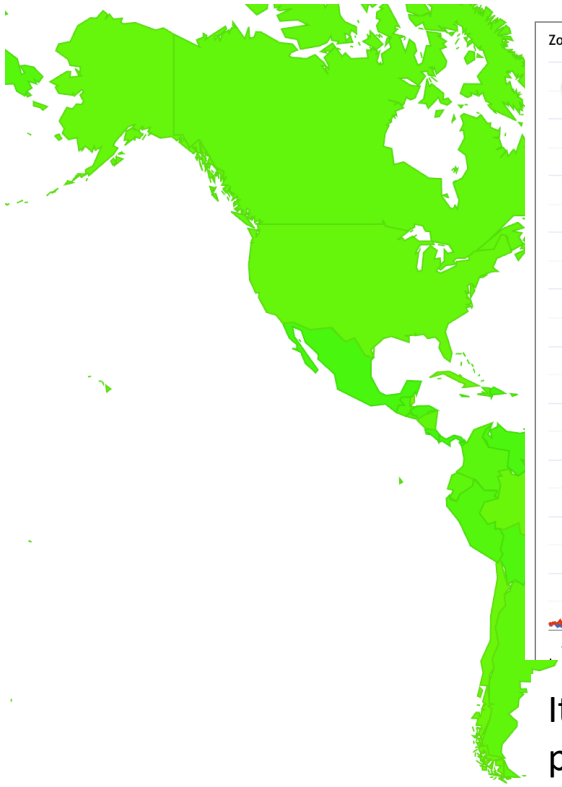
Global QUIC Use

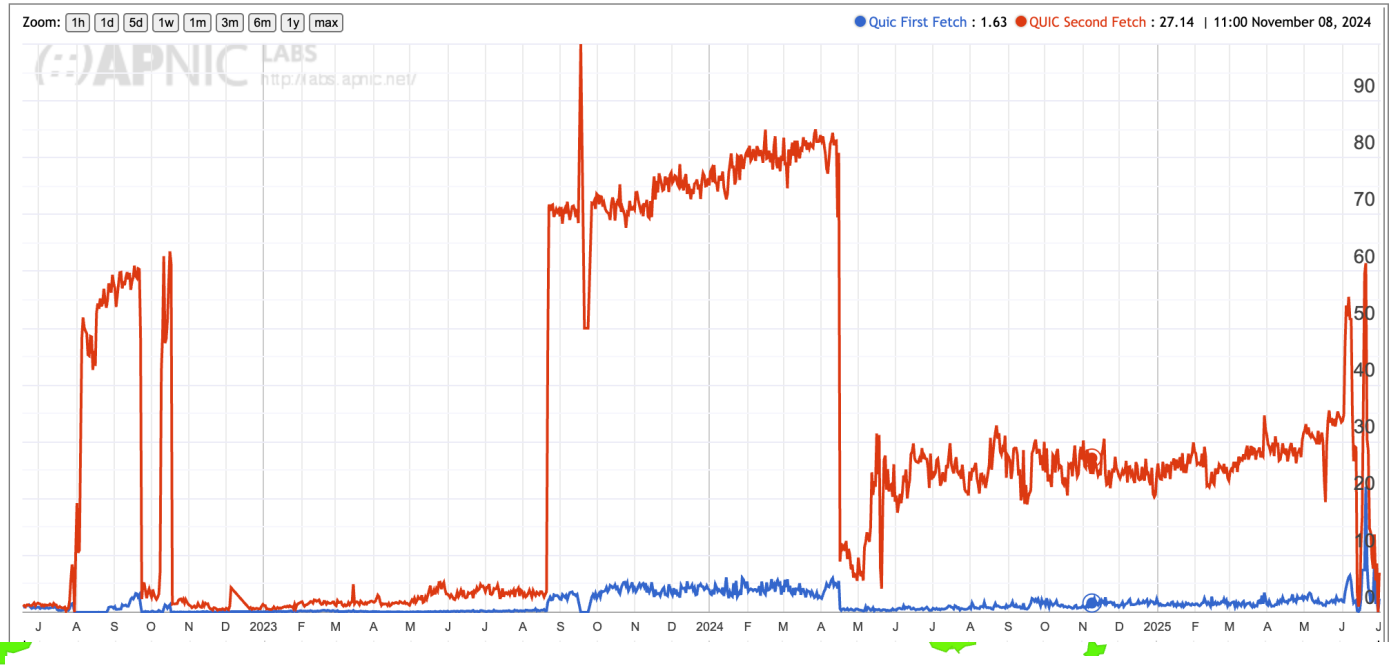


It’s likely that there is some form of national-level block on UDP port 443 traffic in China and Iran

Global QUIC Use

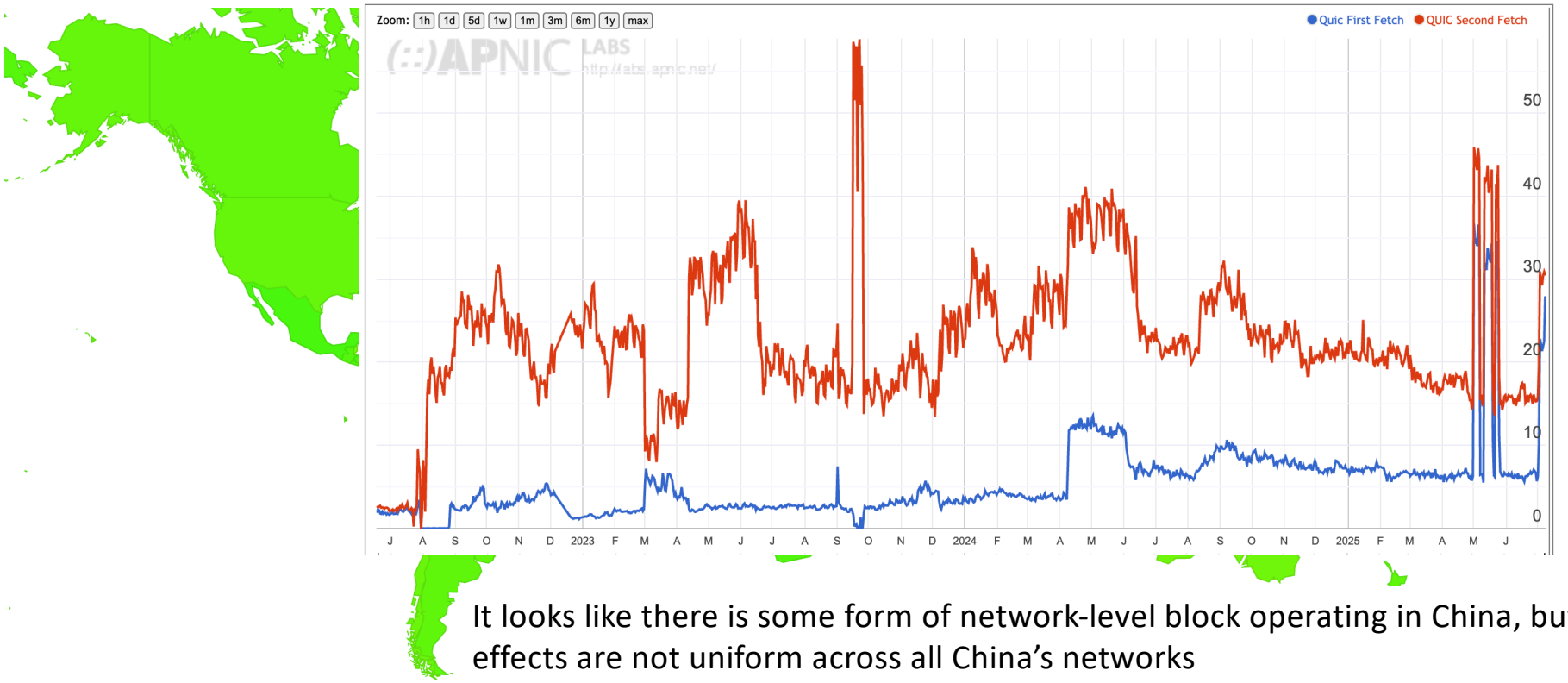
Use of HTTP/3 for Iran (Islamic Republic of) (IR)





It looks like this network-level block has been variously turned on and off over the past few years in Iran

Global QUIC Use



It looks like there is some form of network-level block operating in China, but its effects are not uniform across all China's networks

QUIC Use

- If QUIC access is supported by the current releases by both the major browsers then we should see a high QUIC use rate when the ability to use QUIC is signaled by both methods (alt-svc and DNS HTTPS)
- What do we see?
- In most locales the **alt-svc** method of triggering QUIC is supported by browsers and network infrastructure
- What about the DNS HTTPS method of triggering QUIC?
  - Who uses a DNS HTTPS query?
  - Are HTTPS responses being filtered by DNS infrastructure in some cases?

The DNS HTTPS record

- The HTTPS record can also contain **ipv4hint** and **ipv6hint** attributes
- Any A and AAAA records for a name will be used by a client in preference to these hint attributes
- But if there is no A and no AAAA record in the zone, then a HTTPS-aware client will be forced to use these address hint attributes
- Let's try that, and allow the client to use either HTTP/2 OR HTTP/3:

```
test_name   IN   HTTPS    1   .   alpn="h2,h3" ipv4hint=192.0.2.1 ipv6hint=2001:db8::1
```

DNS HTTPS Use Rate

How many users can use DNS HTTPS responses?

Data collected over a 24-hour period (7/7/2025)

	All		Chrome		Safari		Others	
Samples	13,177,108		9,487,295		3,602,160		87,653	-
DNS HTTPS Query	3,708,895	28.1%	157,695	1.7%	3,506,664	97.3%	44,536	50.8%
Web Fetch (h2/h3)	3,480,873	26.4%	5,957	0.1%	3,469,867	96.3%	5,049	5.8%
Web Fetch (QUIC)	2,710,668	20.6%	4,793	0.1%	2,701,516	75.0%	4,359	5.0%

Few Chrome users (1.7%) perform an HTTPS query, and even fewer (0.1%) followup with a fetch of the web object.

Most Safari users (97.3%) perform an HTTPS query, and most (96.3%) followup with a fetch of the web object. Fewer users (75%) prefer to use QUIC to perform web object retrieval when given the choice.

DNS HTTPS Use Rate

How many users can use DNS HTTPS responses?

Data collected over a 24-hour period (7/7/2025)

	All		Chrome		Safari		Others	
Samples	13,177,108		9,487,295		3,602,160		87,653	-
DNS HTTPS Query	3,708,895	28.1%	157,695	1.7%	3,506,664	97.3%	44,536	50.8%
Web Fetch (h2/h3)	3,480,873	26.4%	5,957	0.1%	3,469,867	96.3%	5,049	5.8%
Web Fetch (QUIC)	2,710,668	20.6%	4,793	0.1%	2,701,516	75.0%	4,359	5.0%

Why is Safari not using QUIC in 25÷ of cases?

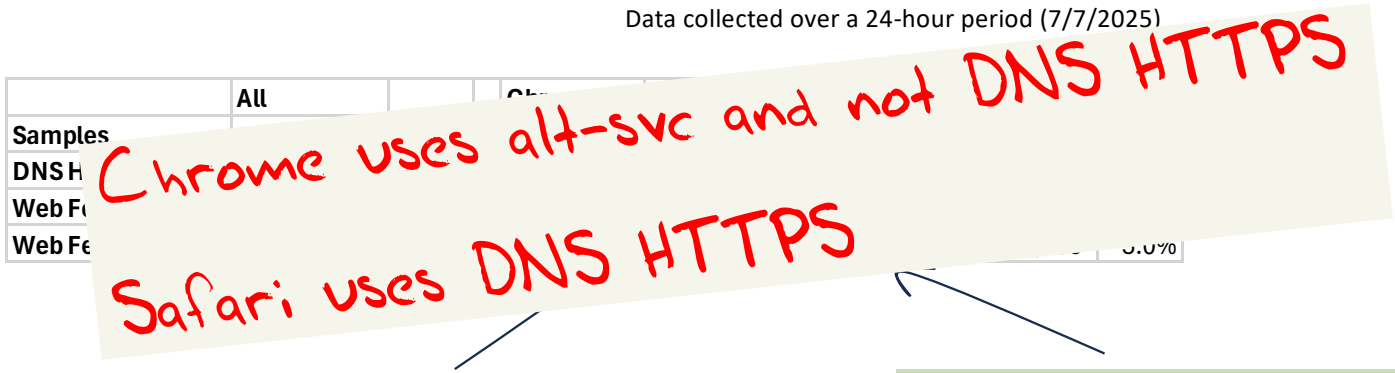
Few Chrome users (1.7%) perform an HTTPS query, and even fewer (0.1%) followup with a fetch of the web object.

Most Safari users (97.3%) perform an HTTPS query, and most (96.3%) followup with a fetch of the web object. Fewer users (75%) prefer to use QUIC to perform web object retrieval when given the choice.

DNS HTTPS Use Rate

How many users can use DNS HTTPS responses?

Data collected over a 24-hour period (7/7/2025)



Few Chrome users (1.7%) perform an HTTPS query, and even fewer (0.1%) followup with a fetch of the web object.

Most Safari users (97.3%) perform an HTTPS query, and most (96.3%) followup with a fetch of the web object. Fewer users (75%) prefer to use QUIC to perform web object retrieval when given the choice.

Does Safari also use alt-svc?

How many users can use QUIC when there is only an **alt-svc** directive and **no** DNS HTTP record?

Data collected over a 24-hour period (10/7/2025)

	All		Chrome		Safari		Others	
Samples	14,163,673		9,788,178		4,251,430		124,065	
TCP First Fetch	14,055,816	99.2%	9,787,962	100.0%	4,151,937	97.7%	115,917	93.4%
QUIC First Fetch	107,857	0.8%	216	0.0%	99,493	2.3%	8,148	6.6%
QUIC 2nd Fetch	9,183,332	64.8%	8,966,915	91.6%	122,086	2.9%	94,331	76.0%

Most Chrome users (91.6%) perform a QUIC retrieval on the subsequent fetch.

Few Safari users (2.9%) perform a QUIC retrieval in the subsequent fetch, indicating that the browsers are NOT following the alt-svc directive

Does Safari also use alt-svc?

How many users can use QUIC when there is only an **alt-svc** directive and **no** DNS HTTP record?

Data collected over a 24-hour period (10/7/

	All		Chrome					
Samples	14,163,673						124,065	
TCP First Fetch	14,055				4,151,937	97.7%	115,917	93.4%
QUIC			216	0.0%	99,493	2.3%	8,148	6.6%
	9,152	64.8%	8,966,915	91.6%	122,086	2.9%	94,331	76.0%

No, Safari does not appear to use the alt-svc directive

Most Chrome users (91.6%) perform a QUIC retrieval on the subsequent fetch.

Few Safari users (2.9%) perform a QUIC retrieval in the subsequent fetch, indicating that the browsers are NOT following the alt-svc directive

Conclusions/Questions

- If you want to serve content over QUIC you have to support BOTH QUIC trigger methods of a DNS HTTP record AND an alt-svc directive to signal QUIC capability to Chrome and Safari clients.
- Why doesn't Chrome also use the HTTPS query?
  - Are they concerned about the greater DNS query load that would result from such a change?
- Why doesn't Safari also use the alt-svc directive?
- Why do 24% of Safari users NOT perform a QUIC fetch despite a HTTPS record being queried?
- Why do 2% of Safari users perform an initial QUIC fetch when there is no DNS HTTPS trigger?

Thanks!