

# ALT-SVC FOR THE PEOPLE

## H.T.T.P

ft. Lucas Pardue



- Motivation? See next slide
- Alternatives == Origins
  - Fully authoritative **iff** reasonable assurance



## Alt-Svc RFC 7838

UABH  
+ 12.0

LE TALK - 9:15 UPDATE - 9:17 ALLEN ABOUT - EXPOSE - MORNING TEAM - RES

“When an origin has resources that are accessible through a different protocol/host/port combination, it is said to have an alternative service



# Shiny HTTP

Shiny HTTP



# Abstract

- Abstract tuple of {protocol, host, port number}
- Protocol != scheme
  - It's the Application Layer Protocol Negotiation (ALPN) Protocol ID
- Host is **optional** (foreshadowing!)
- Port is a number (but what family? More foreshadowing!)

# Concrete

- Concrete instance aka alt-value:
  - “Alt-Svc” HTTP response header field
  - ALTSVC HTTP/2 (and QUIC?) extension frame
    - May be response-specific or connection specific
- Alt-values can be associated with an **OPTIONAL** semicolon-separated list of additional parameters
  - name/value pairs
  - “ma” and “persist” are defined in RFC 7838
  - Unknown parameters **MUST** be ignored

```
http/1.1=":443"
```

```
h2="example.org:443"
```

```
clear
```

```
h2=":4433"; ma=600
```

```
hqm="232.100.1.1:2000";  
source-address=192.0.2.  
1; session-id=1;  
quic=1;  
peak-flow-rate=3000
```

<https://tools.ietf.org/html/draft-pardue-quic-http-mcast-02#appendix-B.1>



# The end of the URL as we know it

“desirable to separate identification and location in HTTP; keeping the same identifier for a resource, but interacting with it at a different location on the network.”

“clients using an alternative service can change the host, port, and protocol that they are using to fetch resources, but these changes **MUST NOT** be propagated to the application that is using HTTP; from that standpoint, the URI being accessed and all information derived from it (scheme, host, and port) are the same as before.”

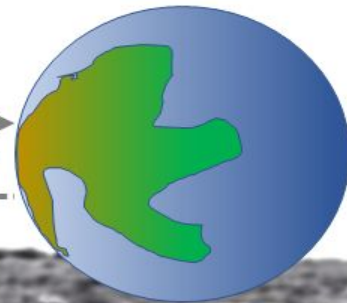


Nobody can see me.



GET earth.example.org/man.html

200 OK + Alt-Svc: hq="moon.example.org:4433"; ma=600



Earth's cert

....  
....  
SAN:  
earth.example.org,  
moon.example.org

GET earth.example.org/man.html

200 OK + Alt-Svc: hq="moon.example.org:4433"; ma=600



Moon's cert

....  
....  
SAN:  
earth.example.org,  
moon.example.org





Consider this, the hint of the century

Happy Alt-Balls

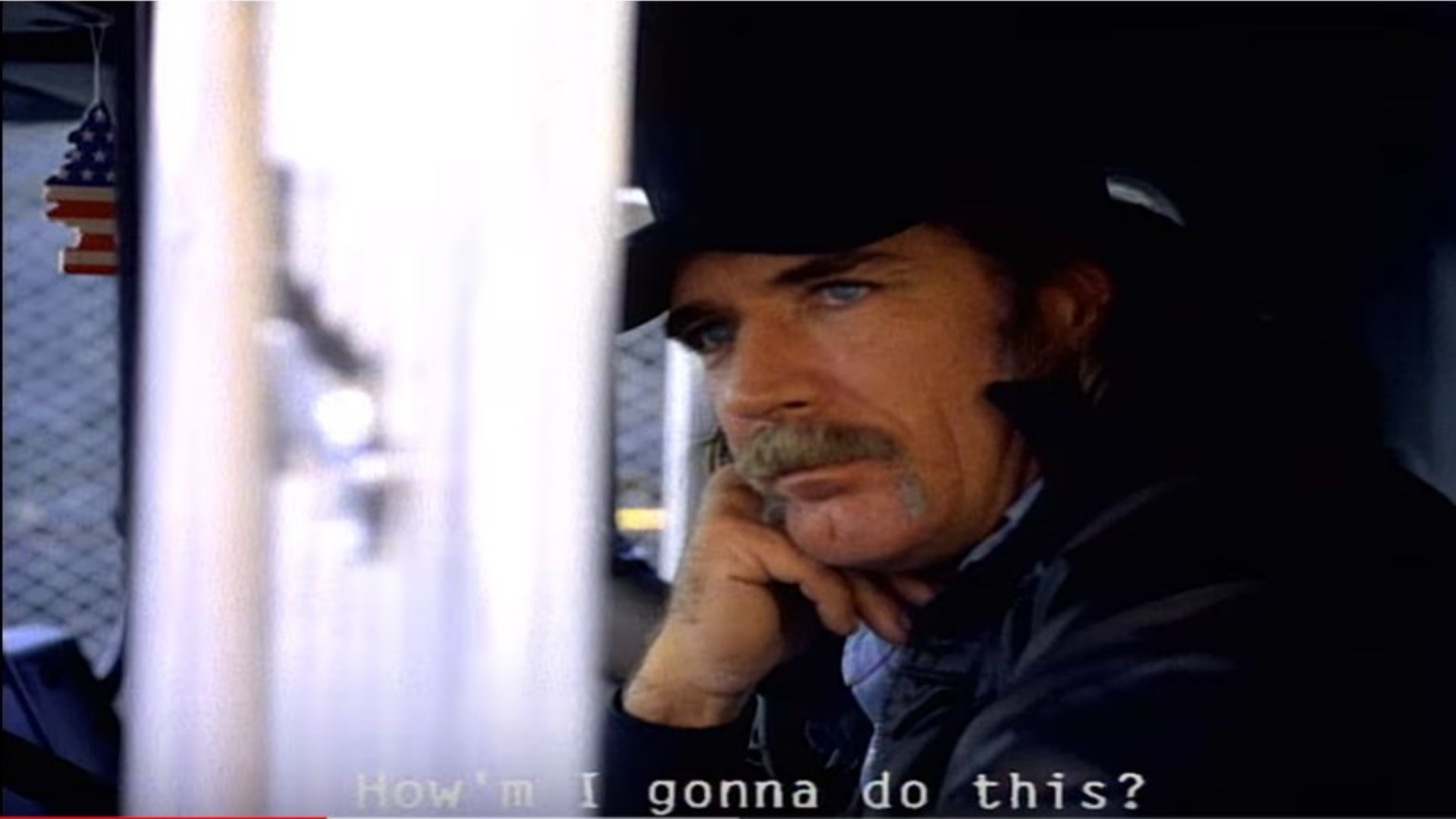




# Caching

- Alternative are default fresh for 24 hours
  - “ma” parameter, measured in seconds
- Alt-Svc adverts update the freshness
  - Alternatives can update their own freshness
- The special alt-value “clear” can clear an entry
- By default, Alt-Svc is cleared on a network change
  - The “persist” parameter can be used to avoid this behaviour
- Connection to the alternative service fails or is unresponsive?
- the client **MAY** fall back to using the origin or another alternative service.
- What happens if connection works but the service (application-layer) has problems?
  - E.g. 500 errors...





How'm I gonna do this?

# Can't we just get by without Alt-Svc?

- Bootstrapping HTTP over QUIC paradox
- Mike Bishop collected thoughts on <https://github.com/quicwg/base-drafts/issues/253>
- Want clients to use “HTTP over UDP”
- https scheme is no good, it implies TCP (or does it?...)
  - RFC 2818, RFC 3986 and RFC 7230 seem to leave gaps that cause arguments
- Port number doesn't define the protocol



# Shiny HTTP

Shiny HTTP





Try QUIC for all [http](#)  
URLs!

# httpq scheme

<https://github.com/quicwg/base-drafts/pull/348>  
Open for > 1 year



ianswett removed the **needs-discussion** label on 15 Mar





TCP

Learn about QUIC  
using Alt-Svc via TCP



Alt-Svc  
via  
HTTP

Alt-Svc  
via DNS

DNS-  
over-  
HTTP

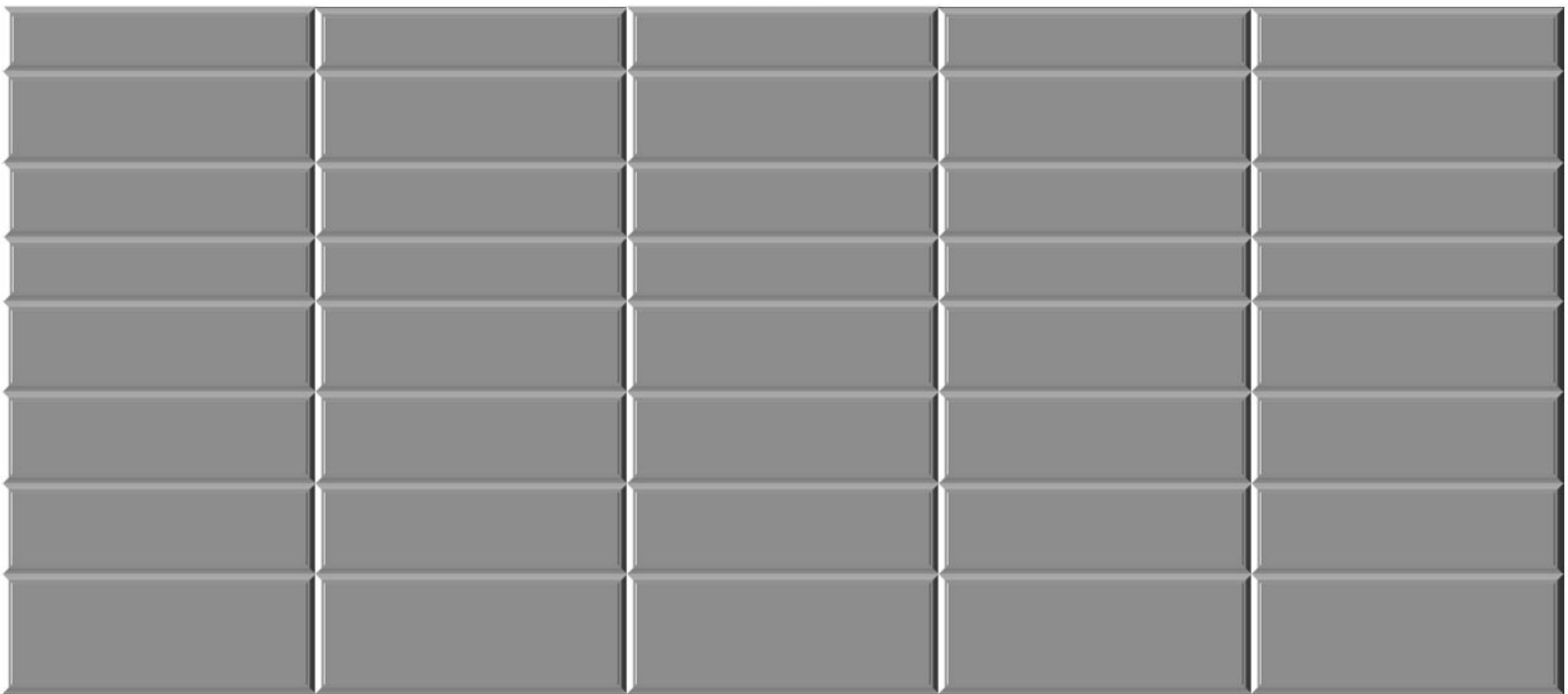


C'MON C'MON













	1	Ignore <i>persist</i> parameter	Take responsibility for selecting Alt-Svc	"Reasonable assurance"
	1	Ignore cache persistence	Couple Alt-Svc too tightly into application/library	Forget to prevent Alt-Svc security downgrade
	2	Forget to time out Alt-Svc according to <i>ma</i>	Forget ALTSVC frame altogether	Forget parameter extensions
1	Scrape Alt-Svc header	Forget to refresh Alt-svc according to <i>ma</i>	Forget ALTSVC frame on connection	Ignore sever errors
1	Expose Alt-Svc to client	Forget misdirect request status 421	Forget to combine Alt-Svc header and ALTSVC frame	Ignore hostname
1	Take responsibility of managing Alt-Svc validity	Forget ORIGIN frame	Forget Alt-Svc DNS record	Forget client may want experimental schemes
1	Ignore network conditions	Forget server push	Forget to combine Alt-Svc header, ALTSVC frame and DNS record	Happy Alt-balls
1	"Connection coalescing"	Forget QUIC out-of-order delivery	Ignore <i>clear</i> value	Alt-Svc loops





<https://bugs.chromium.org/p/chromium/issues/detail?id=392575>

### Implement the Alt-Svc spec

Project Member Reported by [rch@chromium.org](mailto:rch@chromium.org), Jul 9 2014



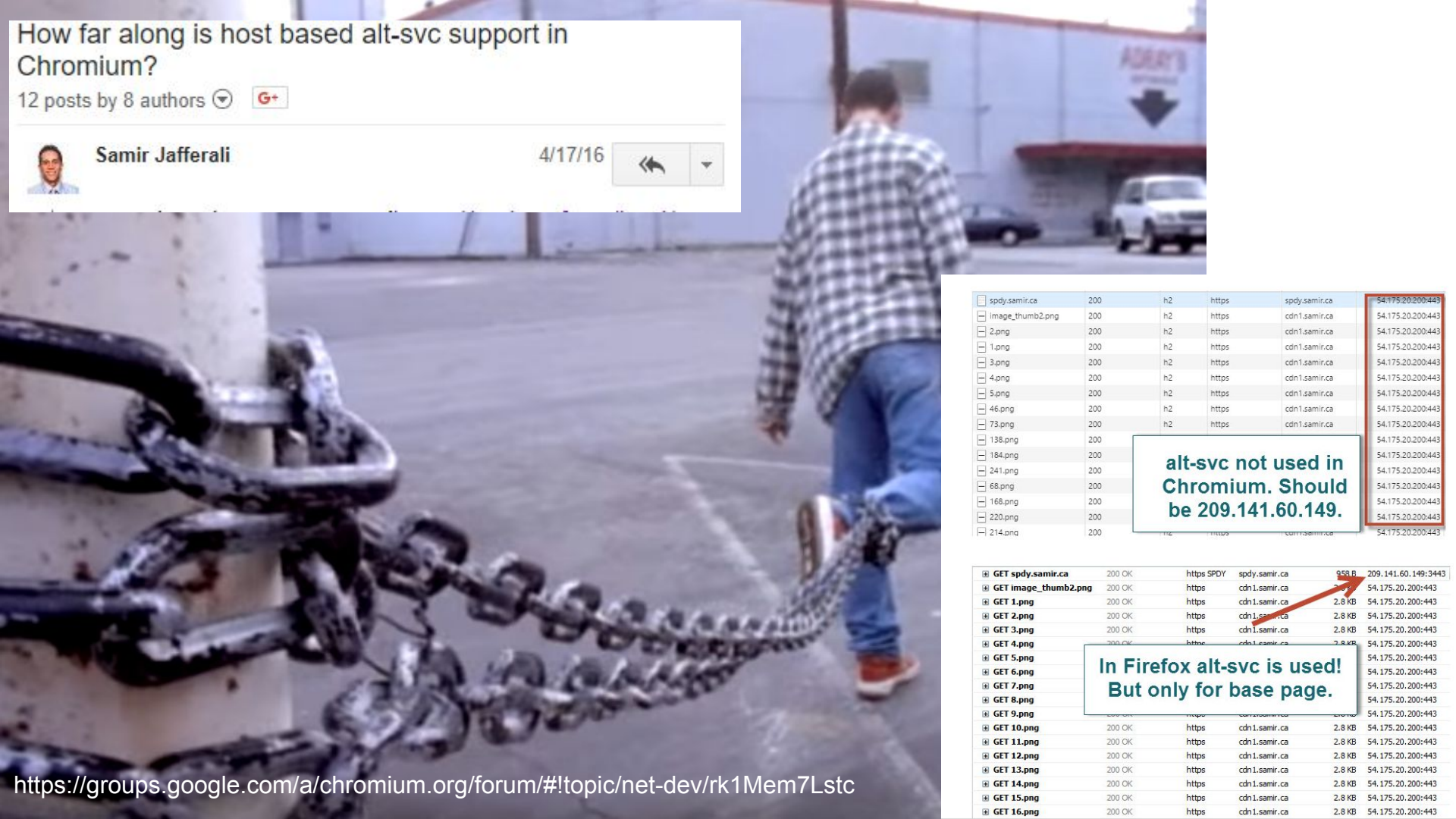
# How far along is host based alt-svc support in Chromium?

12 posts by 8 authors



Samir Jafferli

4/17/16



spdy.samir.ca	200	h2	https	spdy.samir.ca	54.175.20.200:443
image_thumb2.png	200	h2	https	cdn1.samir.ca	54.175.20.200:443
2.png	200	h2	https	cdn1.samir.ca	54.175.20.200:443
1.png	200	h2	https	cdn1.samir.ca	54.175.20.200:443
3.png	200	h2	https	cdn1.samir.ca	54.175.20.200:443
4.png	200	h2	https	cdn1.samir.ca	54.175.20.200:443
5.png	200	h2	https	cdn1.samir.ca	54.175.20.200:443
46.png	200	h2	https	cdn1.samir.ca	54.175.20.200:443
73.png	200	h2	https	cdn1.samir.ca	54.175.20.200:443
138.png	200				54.175.20.200:443
184.png	200				54.175.20.200:443
241.png	200				54.175.20.200:443
68.png	200				54.175.20.200:443
168.png	200				54.175.20.200:443
220.png	200				54.175.20.200:443
214.png	200				54.175.20.200:443

alt-svc not used in Chromium. Should be 209.141.60.149.

GET spdy.samir.ca	200 OK	https SPDY	spdy.samir.ca	958 B	209.141.60.149:3443
GET image_thumb2.png	200 OK	https	cdn1.samir.ca	2.8 KB	54.175.20.200:443
GET 1.png	200 OK	https	cdn1.samir.ca	2.8 KB	54.175.20.200:443
GET 2.png	200 OK	https	cdn1.samir.ca	2.8 KB	54.175.20.200:443
GET 3.png	200 OK	https	cdn1.samir.ca	2.8 KB	54.175.20.200:443
GET 4.png	200 OK	https	cdn1.samir.ca	2.8 KB	54.175.20.200:443
GET 5.png	200 OK	https	cdn1.samir.ca	2.8 KB	54.175.20.200:443
GET 6.png	200 OK	https	cdn1.samir.ca	2.8 KB	54.175.20.200:443
GET 7.png	200 OK	https	cdn1.samir.ca	2.8 KB	54.175.20.200:443
GET 8.png	200 OK	https	cdn1.samir.ca	2.8 KB	54.175.20.200:443
GET 9.png	200 OK	https	cdn1.samir.ca	2.8 KB	54.175.20.200:443
GET 10.png	200 OK	https	cdn1.samir.ca	2.8 KB	54.175.20.200:443
GET 11.png	200 OK	https	cdn1.samir.ca	2.8 KB	54.175.20.200:443
GET 12.png	200 OK	https	cdn1.samir.ca	2.8 KB	54.175.20.200:443
GET 13.png	200 OK	https	cdn1.samir.ca	2.8 KB	54.175.20.200:443
GET 14.png	200 OK	https	cdn1.samir.ca	2.8 KB	54.175.20.200:443
GET 15.png	200 OK	https	cdn1.samir.ca	2.8 KB	54.175.20.200:443
GET 16.png	200 OK	https	cdn1.samir.ca	2.8 KB	54.175.20.200:443

In Firefox alt-svc is used! But only for base page.



### 2.2.1. Draft Version Identification

**\*RFC Editor's Note:** Please remove this section prior to publication of a final version of this document.

Only implementations of the final, published RFC can identify themselves as "hq". Until such an RFC exists, implementations **MUST NOT** identify themselves using this string.

As discussed in-person, the current use of "hq" by Google shouldn't cause long term harm because it's used in conjunction with a quic= list that only includes google QUIC versions.

That being said, I'm happy to change this. We've disabled the experiment in Chrome which thinks in understands "hq" in an Alt-Svc advertisement and we'll also disable the server-side announcement. In addition, I've updated the w3c issue about nextHopProtocol <https://github.com/w3c/navigation-timing/issues/71> and have sent off a Chrome code review <https://chromium-review.googlesource.com/c/chromium/src/+963492> to change the resource timings API output which kicked off this thread.





I've said too much

I haven't said enough

**CUT TO COMMERCIAL**