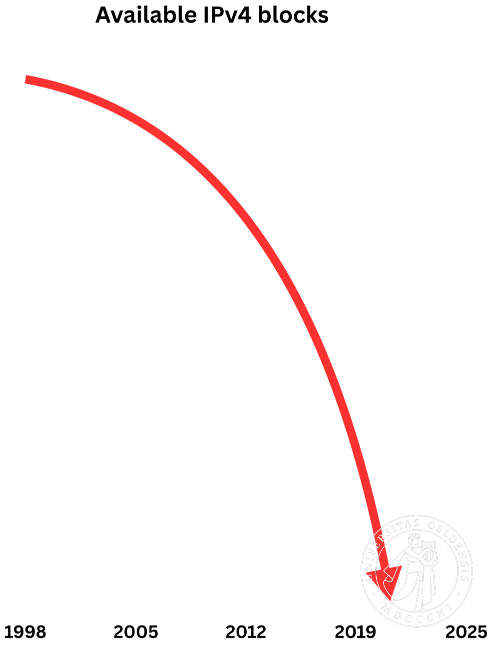
UNIVERSITETET
I OSLO

Protocol Racing

Is it really an advancement?

Joar Heimonen
`contact@joar.me`

October 21, 2025



Agenda

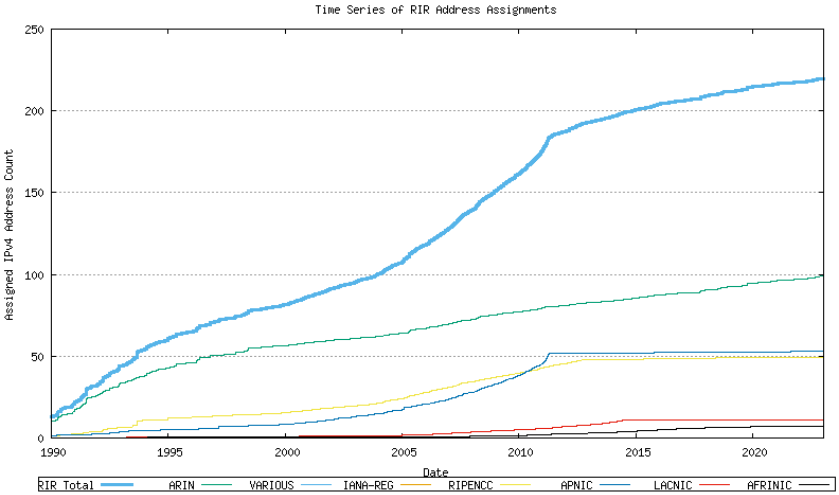
1. A bit of history
2. The issue with Dual-Stacking
3. Happy Eyeballs
4. Issues with Happy Eyeballs
5. Happy Eyeballs version 2
6. Issues with Happy Eyeballs 2
7. The HAPPY WG
8. Transport Services Racing
9. Wrap-up

IPv4

- **RFC 791** – *Internet Protocol*
- Written for DARPA in 1981 (before the IETF existed)
- Designed to interconnect different packet-switched networks (ARPANET, SATNET, university nets)
- Created under the assumption that every device would have its own globally unique, routable address
- 32-bit address space — $2^{32} = 4\,294\,967\,296$ possible addresses
- Sounds like a lot… until you remember that there are 8 billion people alive

Source: (**september** 1981, *Internet Protocol*)

The problem with IPv4



Source: (*IPv4 Address Report*)

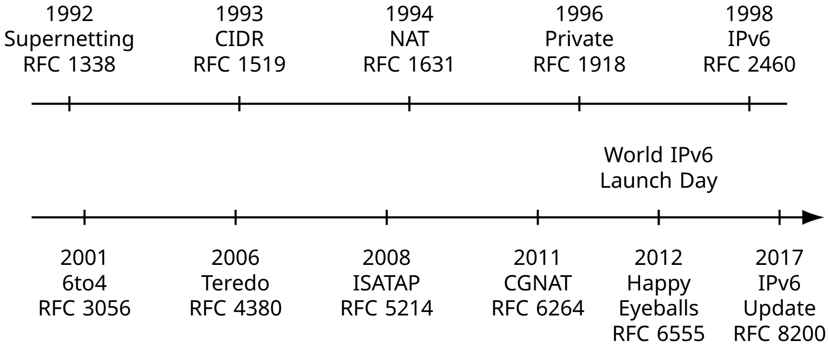
RFC 1338 — Supernetting: an Address Assignment and Aggregation Strategy

It does not attempt to solve the third problem, which is of a more long-term nature, but instead endeavors to ease enough of the short to mid-term difficulties to allow the Internet to continue to function efficiently while progress is made on a longer- term solution.

(The third problem being IPv4 exhaustion)

Source: (Fuller **andothers**, **june** 1992*, Supernetting: An Address Assignment and Aggregation Strategy*)

Timeline of stopgap measures



Timeline of measures from Supernetting to IPv6 'v2'

IPv6

- **RFC 2460** – *Internet Protocol, Version 6*
- Finished in 1998 later updated in 2017
- Designed to fix the issues of IPv4
- Increases the address space from 32-bit to 128-bit
- 340 282 366 920 938 463 463 374 607 431 768 211 456 addresses
- Allows for some cool things like NAT64

Source: (Hinden **and** Deering, **december** 1998*, Internet Protocol, Version 6 (IPv6) Specification*), (Deering **and** Hinden, **july** 2017*, Internet Protocol, Version 6 (IPv6) Specification*)

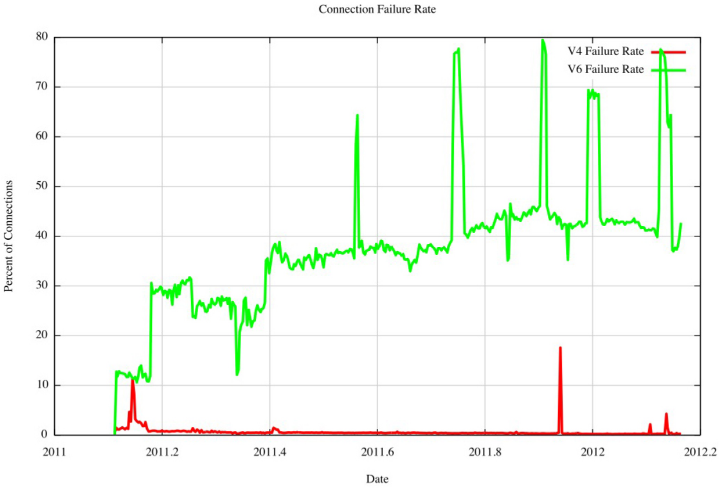
Dual-stack

RFC 1933 — Transition Mechanisms for IPv6 Hosts and Routers

The most straightforward way for IPv6 nodes to remain compatible with IPv4-only nodes is by providing a complete IPv4 implementation. IPv6 nodes that provide a complete IPv4 implementation in addition to their IPv6 implementation are called "IPv6/IPv4 nodes." IPv6/IPv4 nodes have the ability to send and receive both IPv4 and IPv6 packets. They can directly interoperate with IPv4 nodes using IPv4 packets, and also directly interoperate with IPv6 nodes using IPv6 packets.

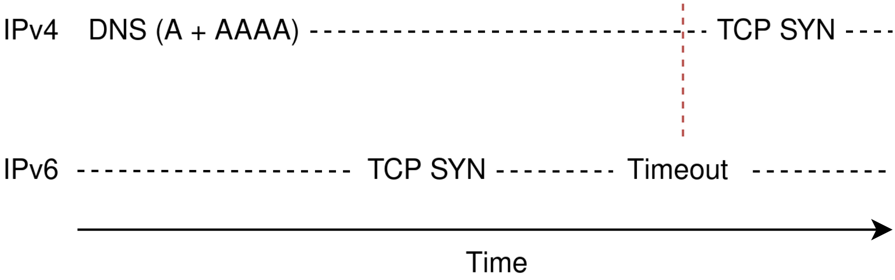
Source: (Nordmark **and** Gilligan, **april** 1996*, Transition Mechanisms for IPv6 Hosts and Routers*)

IPv6 and IPv4 failure rate



Source: (*ISP Column - November 2015*)

Dual-Stacking with IPv6 failure



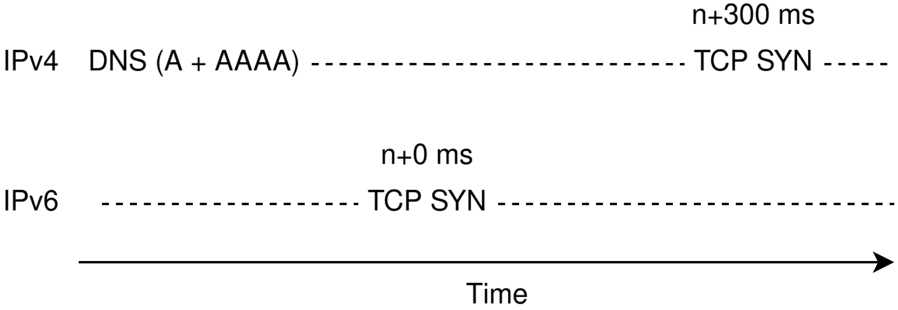
UNIVERSITETET
I OSLO

Happy Eyeballs

- **RFC 6555** – *Happy Eyeballs: Success with Dual-Stack Hosts*
- Created by the V6OPS WG (IPv6 Operations)
- "Races" IPv4 and IPv6 in order to solve blocking behavior
- Applicable to connection oriented transport protocols
- Tries to define a set of standard practices around AAAA DNS records
- No multi-record AAAA responses within the same namespace
- No AAAA specific namespaces e.g. *ipv6.example.com*
- Specifies a head start of 150-250 ms but, all browsers use 300 ms

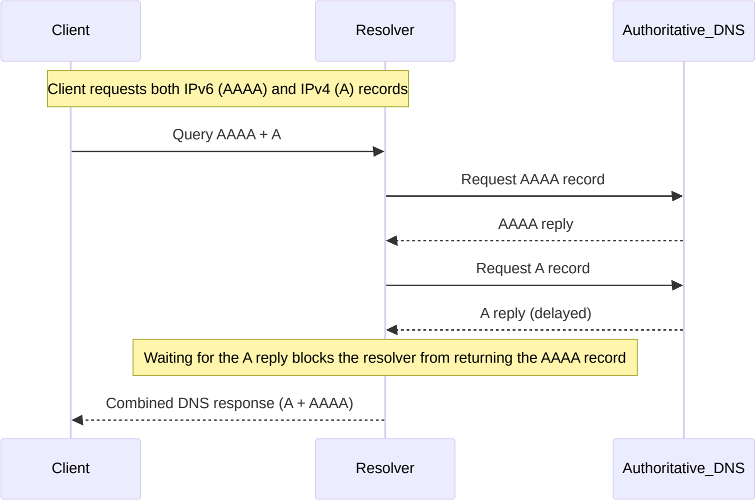
Source: (Wing **and** Yourtchenko, **april** 2012, *Happy Eyeballs: Success with Dual-Stack Hosts*)

Dual-stacking with Happy Eyeballs



UNIVERSITETET
I OSLO

DNS slow family blocking



UNIVERSITETET
I OSLO

Slower Connections

Measuring the Effects of Happy Eyeballs

In 90% of these cases, HE tends to prefer slower IPv6 connection. This shows that the timer value (300 ms) used by the HE algorithm has past its time and is not suitable in today's landscape.

Source: (Bajpai **and** Schönwälder, **july** 2016, *"Measuring the Effects of Happy Eyeballs"*)

Happy Eyeballs version 2

- **RFC 8305** – *Happy Eyeballs Version 2: Better Connectivity Using Concurrency*
- Races DNS families to avoid blocking
- Races multiple records in multi record DNS responses
- Implements a set of 10 rules for sorting the records (Functionally replacing old IPv6 address selection policy defined in **RFC 3484**)
- Prefers DNS over IPv6 but, won’t race them if both IPv4 and IPv6 addresses are available
- Implements a ’resolution delay’ to give AAAA records a head start
- suggests a "Connection Attempt Delay" of 100-250 ms as standard but encourages implementations to calculate delays based on historical RTT

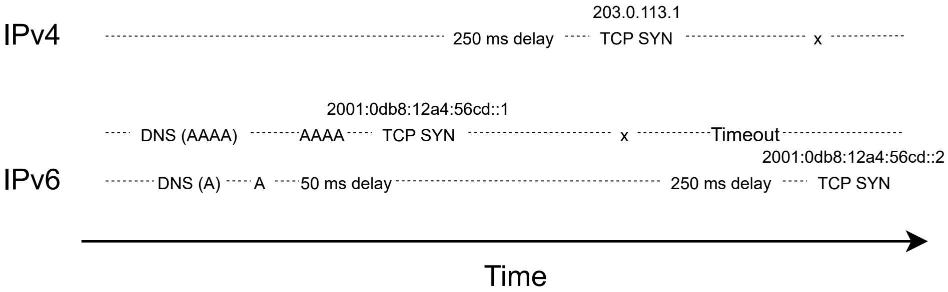
Source: (Schinazi **and** Pauly, **december** 2017, *Happy Eyeballs Version 2: Better Connectivity Using Concurrency*)

Dual-stacking with Happy Eyeballs v2

Sorted addresses after requesting A and AAAA records from *example.com*:

1. 2001:0db8:12a4:56cd::1
2. 203.0.113.1
3. 2001:0db8:12a4:56cd::2

Dual-stacking with Happy Eyeballs v2



Issues with Happy Eyeballs 2

	Prefers IPv6	CAD Impl.	AAAA first[1]	RD Impl.	IPv4 Addrs. Used	IPv6 Addrs. Used	Addr. Selection[2]	Consistency
Chrome *130.0*	●	●	●	○[4]	1	1	○	●
Chromium *130.0*	●	●	●	○[4]	1	1	○	●
Edge *130.0*	●	●	●	○	1	1	○	●
Firefox *132.0*	●	●	○	○	1	1	○	◐
Safari *17.6*	●	●	●	●	10	10	●	○
Mobile Safari	●	●	●	●	-	-	-	-
Chrome Mobile	●	●	●	○	-	-	-	-
`curl` *7.88.1*	●	●	●	○	1	1	○	-
`wget`[3] *1.21.3*	●	○	○	○	-	1	○	-

● Observed as defined; ◐ Observed with RFC deviation; ○ Not observed

Source: (Sattler **andothers**, **october** 2025, *“Lazy Eye Inspection: Capturing the State of Happy Eyeballs Implementations”*)

Issues with Happy Eyeballs 2

Weird behavior with NAT64

If a DNS64 server is available it will race the different records without being aware of which are synthesized and which are native

The HAPPY WG

- Created in early 2025
- First release of the WG charter refers to it as the HAPPIE WG (Heuristics and Algorithms to Prioritize Protocol Initiation and Establishment)
- Becomes HAPPY WG (Heuristics and Algorithms to Prioritize Protocol deploYment)
- Created to develop Happy Eyeballs v3
- Secondary milestone of creating "An informational document that explains the impact of Happy Eyeballs on detecting and measuring broken deployments, with recommendations on how to report errors"

Source: (*Heuristics and Algorithms to Prioritize Protocol deploYment (Happy)*)

draft-ietf-happy-happyeyeballs-v3-02

- Implements all the same features as Happy Eyeballs v2
- Added support for NAT64 networks... The implementation will synthesize IPv6 addresses by embedding the IPv4 to the /96 NAT64 prefix (64:ff9b::/96)
- Queries SVCB and HTTPS records if available, this allows the client to sort records based on discovered transport protocols TCP, QUIC or both

Source: (Pauly **andothers**, **october** 2025, *Happy Eyeballs Version 3: Better Connectivity Using Concurrency*)

Transport Services Racing

- Suggests much of the same racing found in HE 2 and 3
- Introduces racing between different network interfaces
- Does not race DNS
- It has the goal of providing an optimal connection for an applications needs instead of the fastest available one

Source: (Brunstrom **andothers**, **january** 2025, *Implementing Interfaces to Transport Services*)

Is protocol racing an advancement?

Questions?

References I

- Bajpai, Vaibhav **and** Jürgen Schönwälder (**july** 2016). “Measuring the Effects of Happy Eyeballs”. **in***Proceedings of the 2016 Applied Networking Research Workshop*: Berlin Germany: ACM, **pages** 38–44. ISBN: 978-1-4503-4443-2. DOI: 10.1145/2959424.2959429. (**urlseen** 18/10/2025).
- Brunstrom, Anna, Tommy Pauly, Reese Enghardt, Philipp S. Tiesel **and** Michael Welzl (**january** 2025). *Implementing Interfaces to Transport Services*. Request for Comments RFC 9623. Internet Engineering Task Force. DOI: 10.17487/RFC9623. (**urlseen** 21/10/2025).
- Deering, Steve E. **and** Bob Hinden (**july** 2017). *Internet Protocol, Version 6 (IPv6) Specification*. Request for Comments RFC 8200. Internet Engineering Task Force. DOI: 10.17487/RFC8200. (**urlseen** 18/10/2025).

References II

- Fuller, V., T. Li, J. Yu **and** K. Varadhan (**june** 1992). *Supernetting: An Address Assignment and Aggregation Strategy*. techreport RFC1338. RFC Editor, RFC1338. DOI: `10.17487/rfc1338`. (**urlseen** 16/10/2025).
- *Heuristics and Algorithms to Prioritize Protocol deploYment (Happy)* (2025). https://datatracker.ietf.org/wg/happy/about/. (**urlseen** 21/10/2025).
- Hinden, Bob **and** Steve E. Deering (**december** 1998). *Internet Protocol, Version 6 (IPv6) Specification*. Request for Comments RFC 2460. Internet Engineering Task Force. DOI: `10.17487/RFC2460`. (**urlseen** 11/12/2024).
- *Internet Protocol* (**september** 1981). Request for Comments RFC 791. Internet Engineering Task Force. DOI: `10.17487/RFC0791`. (**urlseen** 18/10/2025).
- *IPv4 Address Report* (2025). https://ipv4.potaroo.net/. (**urlseen** 16/10/2025).

References III

- *ISP Column - November 2015* (2025). https://www.potaroo.net/ispcol/2015-11/v6perf.html. (**urlseen** 17/10/2025).
- Nordmark, Erik **and** Robert E. Gilligan (**april** 1996). *Transition Mechanisms for IPv6 Hosts and Routers*. Request for Comments RFC 1933. Internet Engineering Task Force. DOI: 10.17487/RFC1933. (**urlseen** 17/10/2025).
- Pauly, Tommy, David Schinazi, Nidhi Jaju **and** Kenichi Ishibashi (**october** 2025). *Happy Eyeballs Version 3: Better Connectivity Using Concurrency*. Internet Draft draft-ietf-happy-happyeyeballs-v3-02. Internet Engineering Task Force. (**urlseen** 21/10/2025).

References IV

- Sattler, Patrick, Matthias Kirstein, Lars Wüstrich, Johannes Zirngibl **and** Georg Carle (**october** 2025). "Lazy Eye Inspection: Capturing the State of Happy Eyeballs Implementations". **in***Proceedings of the 2025 ACM Internet Measurement Conference*: **pages** 213–221. DOI: 10.1145/3730567.3732925. arXiv: 2412.00263 [cs]. (**urlseen** 21/10/2025).
- Schinazi, David **and** Tommy Pauly (**december** 2017). *Happy Eyeballs Version 2: Better Connectivity Using Concurrency*. Request for Comments RFC 8305. Internet Engineering Task Force. DOI: 10.17487/RFC8305. (**urlseen** 18/10/2025).
- Wing, Dan **and** Anew Yourtchenko (**april** 2012). *Happy Eyeballs: Success with Dual-Stack Hosts*. Request for Comments RFC 6555. Internet Engineering Task Force. DOI: 10.17487/RFC6555. (**urlseen** 18/10/2025).

Joar Heimonen
E-post: `contact@joar.me`
Protocol Racing
Is it really an advancement?