
Public Review for Measuring YouTube over IPv6

Vaibhav Bajpai, Saba Ahsan, Jürgen Schönwälder and Jörg Ott

This paper presents measurements of YouTube performance when streamed over IPv6 vs. IPv4. In particular, the authors (i) use an active measurement tool they have developed specifically for mimicking YouTube clients, (ii) deploy it on 100 different SamKnows nodes (dual-stacked IPv4/IPv6) representing 66 different ASes and over a period of 34 months (from Aug. 2014 until June 2017), and (iii) report a range of metrics assessing both TCP and video performance. Their main finding is that IPv6 is overwhelmingly (97% of the times) preferred over IPv4, due to the shorter TCP connection establishment times. However, the actual performance is found to be worse over IPv6 than IPv4, across all video metrics (startup delay, throughput, and duration of stalls), which can be partially explained by a difference in availability of content caches over IPv6 vs. IPv4.

This paper's main contribution lies in the large scale measurement study. It reports on the status and interaction of two major trends in today's Internet: IPv6 deployment and video (at least YouTube-like) traffic. The measurement study is large in all aspects (w.r.t. spatial distribution of vantage points, time period, multiple performance metrics) and can inform the community. Furthermore, it can inform the design choices made in this space, e.g., in IETF standards, ABR streaming, ISPs use of caches. The measurement tool and datasets are made available to the community.

The paper does also have its limitations, as explicitly stated in Section 10. First, the measurement tool is not a realistic YouTube client: it does take as input YouTube URLs but it does not implement state-of-the-art ABR algorithms, which have been continuously evolving throughout the measurement period. Second, this CCR paper is a continuation of the authors' PAM 2015 paper, which presented the tool and a preliminary study over 20 days. Third, and perhaps more important, much is left to be desired in translating the empirical observations to actual recommendations and actions, as hinted towards the end of the paper. We would like to thank the anonymous reviewers for their constructive feedback that helped significantly improve and clarify several of the aforementioned aspects.

Overall, we hope that this paper can be informative for the community w.r.t. the current state of IPv6 and YouTube-like traffic.

Public review written by
Athina Markopoulou
University of California, Irvine, USA

Artifacts Review for Measuring YouTube over IPv6

Vaibhav Bajpai, Saba Ahsan, Jürgen Schönwälder and Jörg Ott

This paper performs measurements of YouTube through dual-stacked probes distributed across the world. Those are obtained using a YouTube-like client software that hourly downloads video using both IPv4 and IPv6. Outputs of this software are then collected in a SQL database containing a 3-years long dataset.

The YouTube-like client software is available through Github. It takes several parameters such as the URL of the video to fetch, the length of the playout buffer, the maximal test duration and whether the test should be performed over IPv4 or IPv6. Documentation about its output is also provided. The measurement database is made available through two servers. In addition, the authors provide Jupyter notebooks for each of the graphs present in the paper in another Github repository. These notebooks directly operates with the databases hosted on servers, making them easy to run. Overall, I propose to label the paper *Measuring YouTube over IPv6* with the following badge:

- **Artifacts Evaluated – Reusable:** The Youtube-like client can be easily built on Ubuntu 16.04 through a Makefile and README instructions. The tool provides sensible data and is quite easy to use. The code seems readable. Jupyter notebooks are easily accessed and functional. They ease the reproducibility of the results.

Artifacts review written by
Quentin De Coninck
*Universite Catholique de Louvain,
Belgium*

Measuring YouTube Content Delivery over IPv6

Vaibhav Bajpai
TU Munich

Jürgen Schönwälder
Jacobs University Bremen

Saba Ahsan
Aalto University

Jörg Ott
TU Munich

ABSTRACT

We measure YouTube content delivery over IPv6 using ~100 SamKnows probes connected to dual-stacked networks representing 66 different origin ASes. Using a 34-months long (Aug 2014-Jun 2017) dataset, we show that success rates of streaming a stall-free version of a video over IPv6 have improved over time. We show that a Happy Eyeballs (HE) race during initial TCP connection establishment leads to a strong (more than 97%) preference over IPv6. However, even though clients prefer streaming videos over IPv6, we observe worse performance over IPv6 than over IPv4. We witness consistently higher TCP connection establishment times and startup delays (~100 ms or more) over IPv6. We also observe consistently lower achieved throughput both for audio and video over IPv6. We observe less than 1% stall rates over both address families. Due to lower stall rates, bitrates that can be reliably streamed over both address families are comparable. However, in situations, where a stall does occur, 80% of the samples experience higher stall durations that are at least 1s longer over IPv6 and have not reduced over time. The worse performance over IPv6 is due to the disparity in the availability of Google Global Caches (GGC) over IPv6. The measurements performed in this work using the `youtube` test and the entire dataset is made available [5] to the measurement community.

CCS CONCEPTS

• **Networks** → Network monitoring;

KEYWORDS

YouTube, IPv6, SamKnows, Performance

1 INTRODUCTION

The Internet is rapidly exhausting IPv4 address space [33], which has prompted global initiatives (such as the World IPv6 Launch day [35] in 2012) to promote the deployment and adoption of IPv6 [13]. Within a span of 5 years since the initiative, global adoption of IPv6 [12] has increased to ~19% as of Jun 2017 (see Fig. 1) according to Google IPv6 adoption statistics [19] with Belgium (~49%), US (~35%), Germany (~29%) and Switzerland (~27%) leading IPv6 adoption rates. This has largely been possible due to spear-headed IPv6 deployment by service providers both in the fixed-line (such as Telenet, Belgacom, VOO in Belgium, Comcast in US, Deutsche Telekom and Kabel Deutschland in Germany and Swisscom in Switzerland) and cellular (such as AT&T, Verizon Wireless and T-mobile USA) space.

Nadi Sarrar *et al.* in [34] (2012) show that IPv6 traffic after the World IPv6 Day in 2011 is largely dominated by services running over HTTP and that YouTube is the primary service over HTTP that

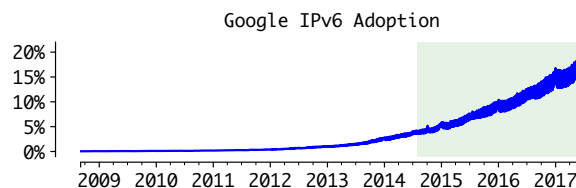
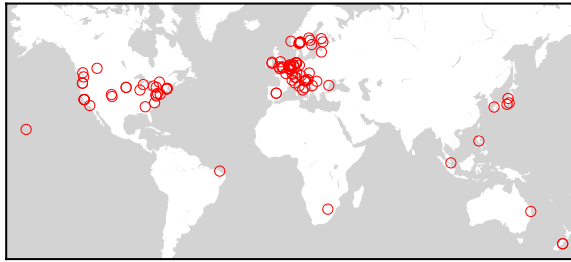


Figure 1: Timeseries of fraction of users reaching Google over IPv6 [19]. The shaded area represents the duration (Aug 2014-Jun 2017) of this study.

contributes heavily to large volumes of IPv6 traffic. Today, AMS-IX on a daily basis witnesses up to 88 Gbps / 5.5 Tbps of IPv6 traffic (as of Jun 2017) with timing of peaks aligned over both address families [16]. Fixed-line service providers such as Comcast and Swisscom estimate IPv6 traffic within their network to be ~25% of the total traffic [31]. In terms of traffic volume this is more than 1 Tbps of native IPv6 traffic (as of Jul 2014) as witnessed by Comcast. Furthermore, Swisscom reports (as of Oct 2014) that 60% of their IPv6 traffic is served by YouTube (with 5% by Facebook) alone [31]. As can be seen, YouTube is the single largest source of IPv6 traffic. This suggests that measuring the performance of YouTube content delivery over IPv6 is necessary today. We want to know – Do users experience benefit (or suffer) from YouTube videos that are delivered over IPv6? Towards this pursuit, we developed an active test (`youtube`) [3] (2015) that measures non-adaptive (see § 10 for limitations of the test) YouTube content delivery over IPv4 and IPv6. We deployed this test on ~100 geographically distributed SamKnows [8] probes (see Fig. 2) to provide diversity of network origins. These probes receive native IPv6 connectivity and belong to different ISPs covering 66 different origin ASes. In this paper, we perform analysis using a 34-months long (Aug 2014-Jun 2017) dataset collected from these dual-stacked probes.

Our contributions – *a*) We show that success rates (see § 3) of streaming a stall-free version of a video over IPv6 have improved over time, *b*) We show that a HE race during initial TCP connection establishment leads to a strong (more than 97%) preference (see § 4) to stream audio and video content over IPv6, *c*) Even though clients prefer streaming videos over IPv6, we observe worse performance over IPv6 than over IPv4. We witness consistently higher TCP connection establishment times and startup delays (100 ms or more) (see § 5) over IPv6. *d*) Furthermore, we observe consistently lower achieved throughput (see § 6) both for audio and video streams over IPv6, although the throughput difference has improved over time. *e*)



RESIDENTIAL	78	RIPE	60
NREN / RESEARCH	10	ARIN	29
BUSINESS / DATACENTER	08	APNIC	10
OPERATOR LAB	04	AFRINIC	01
IXP	01	LACNIC	01

Figure 2: Measurement trial of ~100 dual-stacked SamKnows probes as of Jun 2017. The separate tables represent the number of probes by network type (left) and by regional Internet registries (right). The metadata for each probe is available online: <https://goo.gl/E2m22J>

We observe less than 1% stall rates (see § 7) over both address families. Due to lower stall rates, bitrates that can be reliably streamed over both address families are comparable. However in situations where a stall does occur, 80% of the samples experience stall durations that are at least 1s longer over IPv6 and have not reduced over time, f) We also witness that 97% of our probes receive content delivery through a content cache (see § 8) over IPv4 while only 5% receive it from a content cache over IPv6.

To help with reproducibility [6], the entire dataset and software (see § 11 for details) used in this study is made available to the measurement community.

2 BACKGROUND

2.1 Related Work

A number of studies have focussed on characterization of YouTube videos [11, 18] (2007) to profile workload patterns, observe trends of popular videos, and impact of content duplication on system characteristics. These studies have been followed by a number of passive measurement efforts [2, 17] (2010-2011) to study traffic dynamics, load-balancing strategies and device / location-based user access patterns. We do not discuss them in detail, but we refer the reader to a survey [25] (2016) that discusses these related studies. We instead focus on active measurement studies. For instance, Vijay Kumar Adhikari *et al.* in [1] (2012) use PlanetLab vantage points to crawl a finite subset of YouTube videos to explore the logical organization of the YouTube infrastructure. Parikshit Juluri *et al.* in [24] (2013) use `PyTomo` [23], a Python client to measure YouTube experience from within three ISP networks. They witnessed noticeable difference in experienced quality across ISPs. They reason that the selection mechanisms largely vary depending on the delivery policies and individual ISP agreements. Hyunwoo Nam *et al.* in [30] (2016) introduce `YouSlow`, a browser-based plugin that can detect and report startup

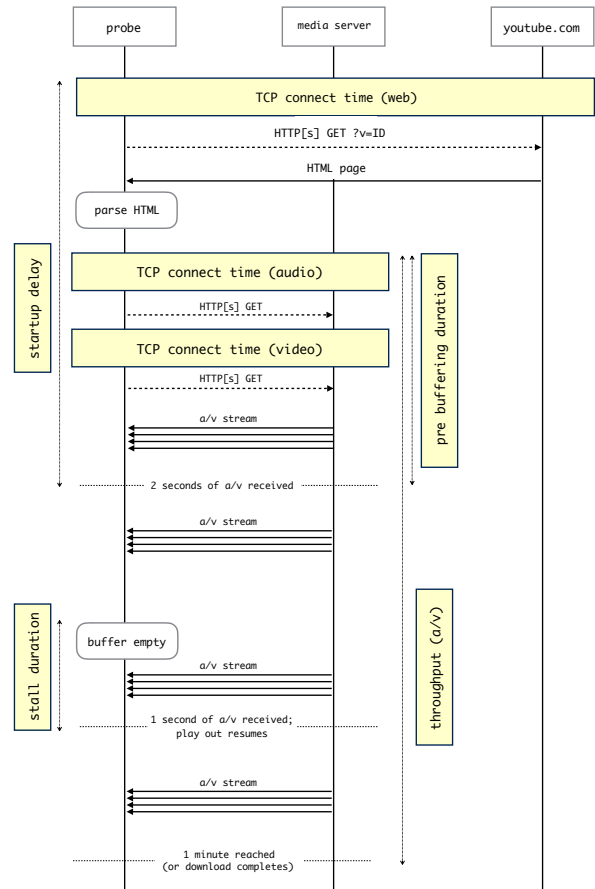


Figure 3: A sequence diagram showing the operation of the youtube test and stages where metrics are collected.

delay, rebuffering and bitrate change events during live playback of a YouTube video. They show that these are good metrics to quantify abandonment rates for short videos on YouTube. These studies however measure YouTube performance over IPv4 only. Studies measuring IPv6 performance [7, 15, 28, 32] (2011-2016) on the other hand have largely focussed on websites.

To the best of our knowledge, this is the first study to measure YouTube performance over IPv6. The study is a continuation of our previous work [3] (2015), where we presented preliminary results from a 20-days (Sep 2014) long dataset collected from a smaller sample of 21 probes deployed within the EU. This paper presents results from probes that cover a much larger geographical area over a longer trial period of 34-months.

2.2 Methodology

We have developed a `youtube test` [3] (2015) that downloads and mimics non-adaptive playout (see § 10 for limitations of the test) of YouTube videos. It measures TCP connect times, startup delay, achievable throughput, bitrate, number of stalls and stall durations as

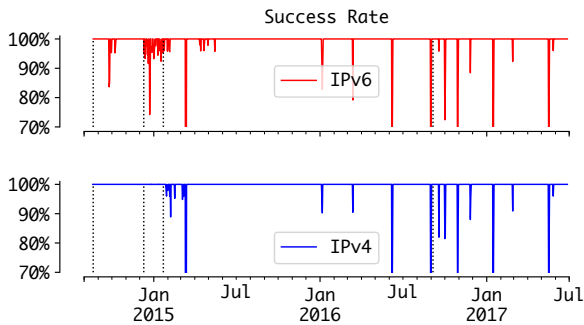


Figure 4: Time series of success rates to YouTube. Success rates over IPv6 have improved over time.

indicators of performance when streaming a YouTube video. Fig. 3 shows the operation of the `youtube` test. The test takes a YouTube URL as input and scrapes the fetched HTML page to extract the list of container formats, available resolutions and URL locations of media servers. The test then establishes two concurrent HTTP sessions to fetch audio and video streams in the desired format and resolution. The test ensures temporal synchronization between the audio and video streams. The test does not at any time render content, but it only reads the container format to extract frame timestamps. The payload is subsequently discarded. Saba Ahsan *et al.* in [4] have shown that active measurement tests towards YouTube should run for a minimum of 1 minute (with a recommended value of 3 minutes). The `youtube` test runs for 1 minute, which is the lower end of the range, however, due to the possibility of interference with user traffic, we find this to be a reasonable compromise.

We deployed the `youtube` test on ~100 SamKnows probes (see Fig. 2) connected in dual-stacked networks representing 66 different origin ASes. As can be seen, most of the probes are connected in residential networks served by the RIPE and ARIN regional registries. To put numbers into perspective, this is more than the number of CAIDA Archipelago (Ark) [26] probes (83 as of Jun 2017) with native IPv6 connectivity. The `youtube` test runs twice, once for IPv4 and subsequently for IPv6 and repeats every hour. We use the YouTube Data API [14] to generate a list of globally popular videos. The popularity list is generated on the SamKnows backend and is refreshed every 12 hours. Probes pull this list on a daily basis. This allows us to measure the same video for the entire day, which enables temporal analysis, while cycling videos on a daily basis allows larger coverage (~871 videos) with different characteristics.

We refer the reader to our previous work [3] (2015) for a more detailed description of our methodology. Since we were limited by the number of probes (21 deployed in EU and JP), our preliminary results [3] were observations from specific vantage points only. In this work, we take this forward and leverage a larger deployment footprint and a longitudinal dataset. This allows us to show that even though lower throughput is observed over IPv6, bitrates that can be reliably streamed are comparable over both address families. It is the startup delay that has been consistently worse over IPv6. We show that the initial web server interaction is responsible for this worse

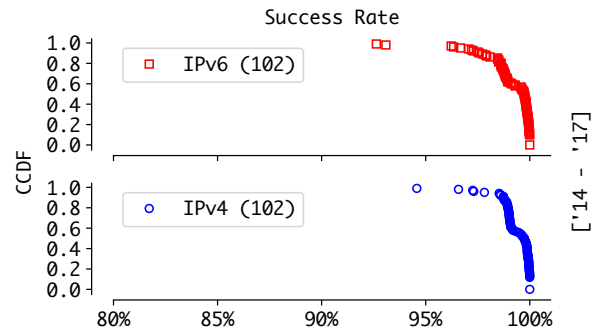


Figure 5: CCDF of success rates over both address families. The probes successfully execute the test slightly more often over IPv4 than over IPv6.

startup delay. The rest of the paper presents this analysis. We also identify areas of improvements within the standards work within the IETF and provide recommendations (see § 9) for ISPs to help improve YouTube content delivery over IPv6.

3 SUCCESS RATE

We start by comparing the success rate of execution of the test over both address families. We define *success rate* as the number of successful iterations to the total number of iterations of the test. The test is deemed successful when it successfully downloads a stall-free version of the video. When a stall occurs, the test reports an error and restarts by stepping down to the same video of a lower bitrate. Fig. 4 shows the timeseries of median success rates over IPv4 and IPv6 across all probes on each day. Vertical markers indicate a rollout (see [5] for a description of changes made in each release) of a test update. We apply a median aggregate, to ensure success rates do not get biased by a specific vantage point. The spikes in the timeseries are not due to outages but an indication that the test experiences a stall and steps down to a lower resolution. It can be seen that success rates in 2014 and 2015 over IPv6 were worse than IPv4. We further investigate the distribution of success rates by removing cases where an error is reported due to a stall event. Fig. 5 shows the distribution of success rate (without stall events) over both address families as seen by all probes. The numbers in the legend represent the number of samples in the distribution. It can be seen that probes relatively achieve a slightly lower success rate over IPv6. For instance, 99% of the probes achieve success rate of more than 94% over IPv4, while 97% of probes achieve the same success rate over IPv6. We investigated the distribution of error codes reported during these failures. The slightly lower success rates over IPv6 are due to issues (such as network error, TCP timeouts or DNS resolution error) encountered closer to the vantage point. Going forward we perform analysis on the subset of results where the test reports success over both address families.

4 IPV6 PREFERENCE

We measure TCP connect times (see Fig. 3) to the YouTube website as well as to media servers hosting audio and video streams. The

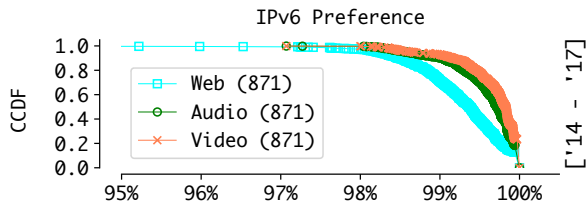


Figure 6: CCDF of TCP connection establishment preference over IPv6. TCP connections over IPv6 to all audio and video streams are preferred at least 97% of the time.

test captures this by recording the time it takes for the `connect()` system call to complete. The DNS resolution time is not taken into account in this measure. This is important to measure because applications running on dual-stacked hosts will prefer connections made over IPv6. This is mandated by the destination address selection policy [37], which makes `getaddrinfo()` resolve DNS names in an order that prefers an IPv6 upgrade path. However, the Happy Eyeballs (HE) algorithm [38] allows these applications to switch to IPv4 in situations where IPv6 connectivity is bad. The connectivity is considered bad when connections made over IPv4 can tolerate the 300 ms advantage imparted to IPv6 and still complete the TCP connection establishment in less time. Fig. 6 shows the effects of the HE algorithm. It can be seen that TCP connections over IPv6 to all audio and video streams are preferred at least 97% of the time.

5 STARTUP DELAY

We have seen that in situations where the test succeeds over both address families, clients strongly prefer streaming videos over IPv6. We now investigate how the observed performance over IPv6 compares to IPv4. We begin by defining a terminology. Let y denote a YouTube video identified by a URL. We call the time taken to establish a TCP connection towards y as $tc(y)$. Since we study the impact of accessing YouTube using different network protocols, we denote the TCP connect time of y accessed over IP version v as $tc_v(y)$. Similarly, we denote prebuffering duration and startup delay of y accessed over IP version v as $pd_v(y)$ and $sd_v(y)$ respectively. We define *prebuffering duration* as the time it takes to fetch 2s of playable video from media servers as shown in Fig. 3. This timer is only triggered once the client has retrieved media server hostnames. As such, prebuffering duration exclusively captures the latency experienced while interacting with the media servers alone. We further define *startup delay* as the time measured from the start of the test until the end of prebuffering as shown in Fig. 3. This also includes the initial time it takes for the test to contact the YouTube web server, scrape the HTML page to extract hostnames of media servers and the aforementioned prebuffering duration. As such, startup delay captures the overall latency experienced for the video to start playing on the screen. DNS resolution times and TCP connect times are accounted in both prebuffering duration and startup delay.

Lower latency achieved using a combined effect of lower TCP connect times and lower startup delay is desirable for a good user experience. We use Eq. 1 to calculate the latency difference over IPv4 and IPv6, where $\Delta t(y)$, $\Delta p(y)$ and $\Delta s(y)$ are the differences

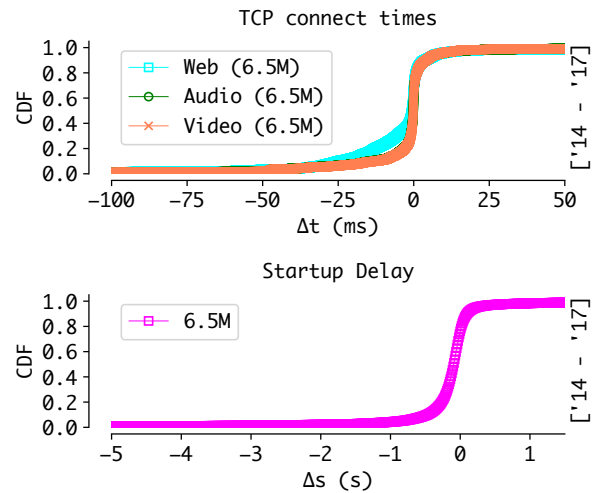


Figure 7: CDF of difference of TCP connect times (above) and startup delay (below) between IPv4 and IPv6. 63% of the audio and video streams (and 72% of web connections) exhibit higher TCP connect times over IPv6 with 14% of them being at least 10 ms slower. 80% of the streams exhibit higher startup delay over IPv6 with 50% being at least 100 ms slower.

between TCP connect times, prebuffering durations and startup delay respectively,

$$\begin{aligned}\Delta t(y) &= tc_4(y) - tc_6(y) \\ \Delta p(y) &= pd_4(y) - pd_6(y) \\ \Delta s(y) &= sd_4(y) - sd_6(y)\end{aligned}\quad (1)$$

Fig. 7 shows the distribution of difference in TCP connect times $\Delta t(y)$ and difference in startup delay $\Delta s(y)$ using the entire 34-months long dataset. Values on the positive scale indicate that IPv6 is faster. The comparison of TCP connect times shows that 63% of the audio and video streams (and 72% of the web connections) are slower over IPv6 with 14% of them being at least 10 ms slower. The comparison of startup delay shows that 80% of the samples are slower over IPv6 with half of the samples being at least 100 ms slower.

We further apply a median aggregate on the TCP connect times, prebuffering duration and startup delay across all probes over each day. Fig. 8 shows the timeseries of median TCP connect times, prebuffering duration and startup delay over IPv4 and IPv6 across all probes. Vertical markers indicate a rollout of a test update. The values on the positive scale indicate that IPv6 is faster. Each of the sub figure is on a different y-scale. It can be seen that TCP connect times tend to be consistently higher over IPv6 and have not improved over time. The TCP connect times towards the webpage appear worse over IPv6 than towards media servers. Even though TCP connect times to fetch audio and video streams are only less than 1 ms slower over IPv6, they play a vital role since it's at this stage where the HE algorithm [38] chooses which address family

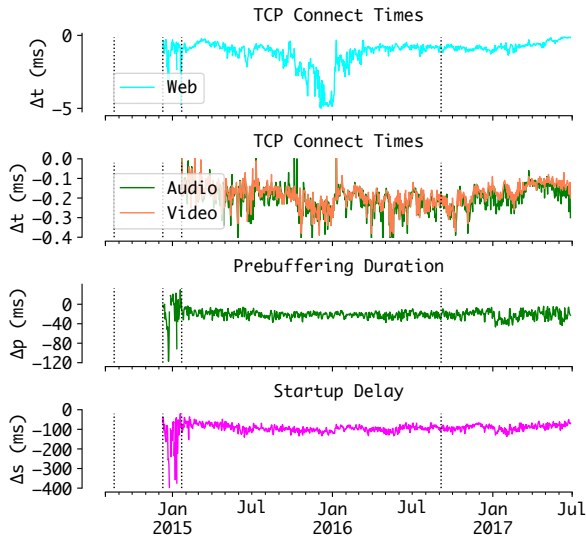


Figure 8: Time series of difference in TCP connect times, prebuffering durations and startup delay over IPv4 and IPv6 to YouTube. The latency is consistently higher over IPv6. Higher prebuffering durations (25 ms or more) and higher startup delays (100 ms or more) are experienced over IPv6.

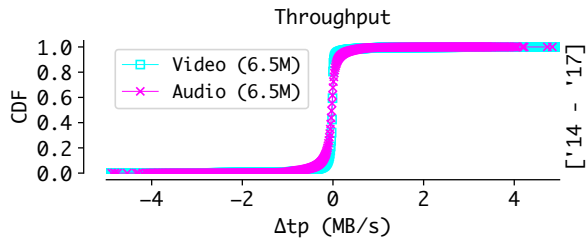


Figure 9: CDF of difference of throughput between IPv4 and IPv6. 80% of the video samples and 60% of the audio samples achieve lower throughput over IPv6.

should be preferred for streaming the video. As a result of a smaller difference in TCP connect times, HE prefers a TCP connection over IPv6. However, once the TCP connection is established, longer startup delays (100 ms or more) are experienced over IPv6. Since the prebuffering durations are not that far off (25 ms or more) over IPv6 compared to that of startup delay, it shows that it is the initial interaction with the web server (see Fig. 3) that makes the startup delay (100 ms or more) worse over IPv6. Our initial observation of TCP connect times also revealed that web connect times over IPv6 are worse than TCP connect times to media servers. As such, even though the media content delivery is almost congruent over both address families, the web server interaction needs to be optimised to reduce the increased startup delay experienced over IPv6.

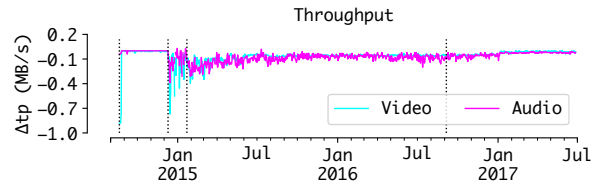


Figure 10: Time series of difference in achieved throughput over IPv4 and IPv6. The achieved throughput is consistently lower over IPv6, but it has improved over time.

6 THROUGHPUT

We have seen that clients strongly prefer streaming videos over IPv6, but they suffer from consistently higher TCP connect times, prebuffering durations (25 ms or more) and startup delays (100 ms or more) when compared to IPv4. We now investigate how the achieved throughput compares over both address families. The test measures throughput over a single TCP connection separately (and combined) over both audio and video streams as shown in Fig. 3. We denote the throughput of y accessed over IP version v as $tp_v(y)$. We use Eq. 2 to calculate the difference in achieved throughput over IPv4 and IPv6.

$$\Delta tp(y) = tp_6(y) - tp_4(y) \quad (2)$$

Fig. 9 shows the distribution of difference in achieved throughput $\Delta tp(y)$ for both audio and video streams using the entire 34-months long dataset. It can be seen that 80% of the video and 60% of the audio samples achieve lower throughput over IPv6. The test steps down to a lower resolution video once a stall event is triggered, which subsequently lowers the achieved throughput, since the test then chooses the next highest bit rate and begins the download from the beginning. This enables the test to produce a more user oriented result in the form of the highest resolution that the client can play out without disruptions over a particular connection. The test is designed to pace the media streams to maintain a playout buffer of 40s (which means, the buffer can only store 40s of playable video) and must wait for the buffer to empty before requesting more frames.

We further apply a median aggregate on the throughput difference across all probes over each day. Fig. 10 shows the timeseries of median throughput difference over IPv4 and IPv6 across all probes. The values on the positive scale indicate that higher throughput is achieved over IPv6. It can be seen that achieved throughput both for audio and video streams tend to be consistently lower over IPv6, although the difference has reduced over time.

7 STALL EVENTS

We have seen that clients prefer streaming videos over IPv6, but the observed performance (both in terms of latency and throughput) over IPv6 is worse. We further compare the number of stall events and stall durations over both address families. We define a *stall* as an event that triggers during playback in situations when a frame is not received before its playout time. Stall events occur due to

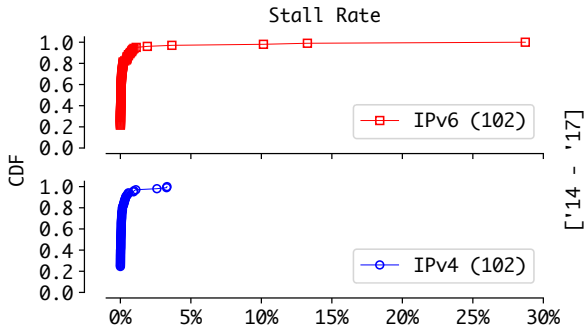


Figure 11: CDF of stall rates over IPv4 and IPv6. 90% of probes experience less than 1% stall rate over IPv4 and IPv6.

throughput constraints caused by a bottleneck at any point on the path between the media server and the client. To avoid unnecessary stalling we use results from SamKnows speed tests [3] to limit the maximum bit rate that the client will attempt to download. The test uses a playout buffer of 40s. In case a stall occurs, 1s of media rebuffering is done before resuming the playout timer as shown in Fig. 3. The media download is paced so as not to exceed capacity of the playout buffer. Fig. 11 shows the distribution of stall rates over IPv4 and IPv6 as seen by all probes. We define *stall rate* as the number of stall events to the total number of iterations of the test. It can be seen that stall rates are comparable over both address families. 90% of the probes witness less than 1% stall rate over both address families. In order to analyse the effects of stalls on achieved bitrate, we utilise a metric, *bitrate reliably streamed* which [27] defines as the highest available bit rate that the test is able to download without experiencing stall events. Since the test cycles through different popular videos each day (which themselves may support different set of available resolutions), we further normalise this metric by taking the ratio of bitrate reliably streamed to the maximum available bit rate of the video. The ratio (*br*) lies between 0 and 1 where 1 is reported in situations when the test can successfully stream the highest available resolution without experiencing any stall events. We observe that 5.7% of the samples over IPv4, while a slightly larger 6.6% of the samples over IPv6 report a *br* value of less than 1. We further observe that 3% of the samples report a higher *br* value over IPv4, while a slightly lower 2% of the samples report a higher *br* value over IPv6. As such, since stall rates are fairly low, the bitrate reliably streamed is comparable over both address families.

In situations where a stall does occur, we measure durations of the stall as shown in Fig. 3. We use Eq. 3 to calculate the difference in stall duration over both address families, where $st_v(y)$ is the stall duration witnessed for video y accessed over IP version v .

$$\Delta st(y) = st_4(y) - st_6(y) \quad (3)$$

Fig. 12 shows the distribution of difference in stall duration $\Delta st(y)$ using the entire 34-months long dataset. The values on the positive scale indicate that stall durations are lower over IPv6. It can be seen that 80% of the samples experience stall durations that are at least 1s longer over IPv6 with half of them being at least 20s longer. We also

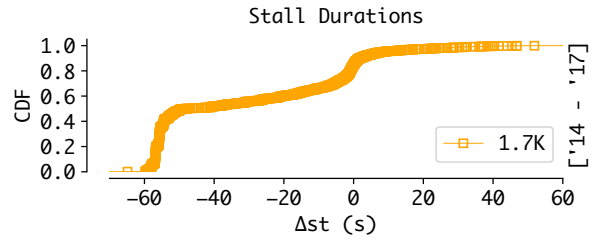


Figure 12: CDF of difference of stall duration between IPv4 and IPv6. 80% of the samples experience stall durations that are at least 1s longer over IPv6.

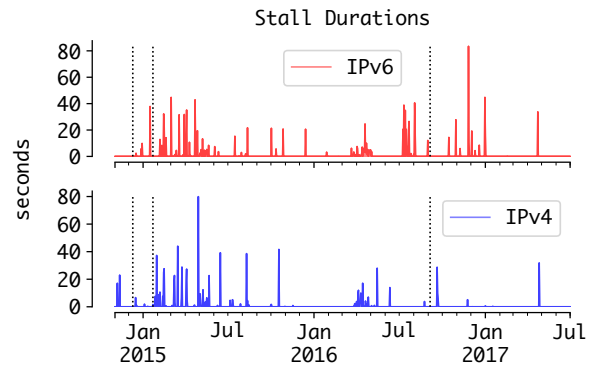


Figure 13: Time series of stall durations over IPv4 and IPv6. Stall durations have not reduced over time.

apply a median aggregate on the stall durations across all probes over each day. Fig. 13 shows the median stall durations over IPv4 and IPv6 across all probes. It can be seen that stall durations do not appear to have reduced over time.

8 CONTENT CACHES

We have seen that clients prefer streaming videos over IPv6, but the observed performance over IPv6 is worse. Furthermore, in situations where a stall occurs, stall durations over IPv6 are also higher. We further investigate the reason for worse performance over IPv6. In order to improve content delivery, operators can deploy servers to host content caches within their networks. These caches form GGC [20] and help bring the content closer to the users, thereby improving performance and minimizing transit bandwidth. In our dataset, we identified GGC by looking up reverse DNS entries of media server IP endpoints. We searched for popular keywords in reverse DNS entries and filtered expressions such as `*-ggc*.sky*` or `*.cache.google*.com` or `ggc*.plus.net` to flag endpoints as GGC nodes. We observe that 97% of probes over IPv4 receive content delivery through a GGC node while only 5% receive it over IPv6. We further flag an IP endpoint as a non-GGC cache (such as an Akamai / Cloudflare cache) if its reverse DNS entry does not match the GGC expressions but the IP endpoint belongs to the

origin AS of the probe. This heuristic provides an indication that the content is served from within the ISP's network. In situations where the content is not served by a content cache, we mapped the IP prefixes to ASNs and used PeeringDB [29] to select ASNs that classify as content providers. This revealed that 96% of the probes do not get content served from a content cache over IPv6 but instead have to reach out to the Google CDN to fetch media streams.

9 RECOMMENDATIONS

We witnessed that HE strongly prefers IPv6 connections for streaming YouTube even though this preference brings worse performance over IPv6. This is because browsers use an HE timer value that has passed its time. The HE timer value (300 ms) was chosen during a time (2012) when broken IPv6 connectivity (attributed largely to failures caused by Teredo [21] and 6to4 [10] relays) was quite prevalent. However, within a span of 5 years, Teredo/6to4 presence has declined to ~0.01% [19] as of Jun 2017. In this changed landscape, we measured the effects of HE [9] and observed that an HE timer value of 150 ms provides a margin benefit of ~10% while retaining same preference levels over IPv6 as is today. Therefore, we recommend browsers to reduce the HE timer value to 150 ms to help reduce the performance penalty in situations where IPv6 is considerably slower. The v6ops working group within the IETF is undergoing rechartering [22] where one of the goals is to update the HE standard with operational experience. We believe our measurements will help inform and improve this standard update.

The performance penalty is attributed to consistently higher TCP connect times and startup delays (~100 ms or more) over IPv6. As such, ISPs should put latency as a first-class citizen when optimizing broadband networks. The higher latency over IPv6 is due to disparity in the availability of content caches over IPv6. Therefore, we recommend ISPs to ensure that their GGC nodes are dual-stacked.

10 LIMITATIONS

The `youtube` test [3] currently supports non-adaptive and step-down playout modes. The test was developed during a time (2014) when most DASH [36] algorithms were proprietary and each of them behaved differently. We made a design decision to not include an adaptive playout mode, since Adaptive Bitrate Streaming (ABR) algorithms were rapidly evolving and it would have been challenging to collect a stable longitudinal dataset if we also kept rolling out test updates during data collection. The step-down mode we use is useful since it helps identify the highest resolution that a client can playout without disruptions over a particular connection and allows us to compare address family differences in identical conditions.

The observations are also biased by our SamKnows probes deployment which largely cover US, EU and JP regions. However, it must be noted that a large fraction of IPv6 deployment today is also centered in these regions, but we concur that the state of IPv6 adoption may change in the future.

11 CONCLUSION

We measured YouTube content delivery over IPv6. Using a 34-months (Aug 2014-Jun 2017) long dataset we showed that success rates of streaming a stall-free version of the video over IPv6 were

lower compared to that of IPv4 but they tend to have improved over time. In situations where the test succeeds over both address families, we observe worse performance over IPv6 with consistently higher TCP connect times and startup delays (100 ms or more) over IPv6. We also observed consistently lower achieved throughput over IPv6 for both audio and video streams. Although we witnessed low stall rates over both address families, in situations where a stall occurred, the stall durations were relatively higher (1s or more) over IPv6. The worse performance is due to disparity in the availability of GGC nodes over IPv6.

Reproducibility Considerations

The `youtube` test is open-sourced and released [5] to the community. The dataset collected by running this test from SamKnows probes is stored as a SQLite database (alongwith the SQL schemas) and also made publicly available [5]. The software used in this study is also released [5]. This includes Jupyter notebooks used in the analysis to generate plots. Guidance on how to reproduce these results is provided [5] and reproducers are encouraged to contact the authors for further questions.

Acknowledgements

We like to thank Sam Crawford and Jamie Mason for providing us support on the SamKnows infrastructure and to all volunteers who host a probe for us. This work was supported by the European Commission Horizon 2020 Programme RIFE Project Grant No. 644663. This work was also supported by the European Community's Seventh Framework Programme in (FP7/2007- 2013) Grant No. 317647 (Leone) and funded by Flamingo, a Network of Excellence project (ICT-318488) supported by the European Commission under its Seventh Framework Programme.

REFERENCES

- [1] Vijay Kumar Adhikari, Sourabh Jain, Yingying Chen, and Zhi-Li Zhang. 2012. Vivisecting YouTube: An active measurement study. In *Proceedings of the IEEE INFOCOM 2012, Orlando, FL, USA, March 25-30, 2012*. 2521–2525. <https://doi.org/10.1109/INFOCOM.2012.6195644>
- [2] Vijay Kumar Adhikari, Sourabh Jain, and Zhi-Li Zhang. 2010. YouTube traffic dynamics and its interplay with a tier-1 ISP: an ISP perspective. In *Proceedings of the 10th ACM SIGCOMM Internet Measurement Conference, IMC 2010, Melbourne, Australia - November 1-3, 2010*. 431–443. <https://doi.org/10.1145/1879141.1879197>
- [3] Saba Ahsan, Vaibhav Bajpai, Jörg Ott, and Jürgen Schönwälder. 2015. Measuring YouTube from Dual-Stacked Hosts. In *Passive and Active Measurement - 16th International Conference, PAM 2015, New York, NY, USA, March 19-20, 2015, Proceedings*. 249–261. https://doi.org/10.1007/978-3-319-15509-8_19
- [4] Saba Ahsan, Varun Singh, and Jörg Ott. 2016. Impact of duration on active video testing. In *Proceedings of the 26th International Workshop on Network and Operating Systems Support for Digital Audio and Video, NOSSDAV 2016, May 13, 2016*. <https://doi.org/10.1145/2910642.2910651>
- [5] Vaibhav Bajpai, Saba Ahsan, Jürgen Schönwälder, and Jörg Ott. 2017. Measuring YouTube Content Delivery over IPv6: Software and Dataset. (2017). <https://github.com/vbajpai/2017-ccr-youtube-analysis>
- [6] Vaibhav Bajpai, Mirja Kühlewind, Jörg Ott, Jürgen Schönwälder, Anna Sperotto, and Brian Trammell. 2017 (to appear). Challenges with Reproducibility. In *ACM SIGCOMM Reproducibility Workshop*.
- [7] Vaibhav Bajpai and Jürgen Schönwälder. 2015. IPv4 versus IPv6 - who connects faster?. In *Proceedings of the 14th IFIP Networking Conference, Networking 2015, Toulouse, France, 20-22 May, 2015*. 1–9. <https://doi.org/10.1109/IFIPNetworking.2015.7145323>

- [8] Vaibhav Bajpai and Jürgen Schönwälder. 2015. A Survey on Internet Performance Measurement Platforms and Related Standardization Efforts. *IEEE Communications Surveys and Tutorials* 17, 3 (2015), 1313–1341. <https://doi.org/10.1109/COMST.2015.2418435>
- [9] Vaibhav Bajpai and Jürgen Schönwälder. 2016. Measuring the Effects of Happy Eyeballs. In *Proceedings of the 2016 Applied Networking Research Workshop, ANRW 2016, Berlin, Germany, July 16, 2016*. 38–44. <https://doi.org/10.1145/2959424.2959429>
- [10] B. Carpenter and K. Moore. 2001. Connection of IPv6 Domains via IPv4 Clouds. RFC 3056. (Feb. 2001). <https://tools.ietf.org/html/rfc3056>
- [11] Meeyoung Cha, Haewoon Kwak, Pablo Rodriguez, Yong-Yeol Ahn, and Sue B. Moon. 2007. I tube, you tube, everybody tubes: analyzing the world's largest user generated content video system. In *Proceedings of the 7th ACM SIGCOMM Internet Measurement Conference, IMC 2007, San Diego, California, USA, October 24-26, 2007*. 1–14. <https://doi.org/10.1145/1298306.1298309>
- [12] Jakub Czyz, Mark Allman, Jing Zhang, Scott Lekel-Johnson, Eric Osterweil, and Michael Bailey. 2014. Measuring IPv6 adoption. In *ACM SIGCOMM 2014 Conference, SIGCOMM'14, Chicago, IL, USA, August 17-22, 2014*. 87–98. <https://doi.org/10.1145/2619239.2626295>
- [13] S. Deering and R. Hinden. 1998. Internet Protocol, Version 6 (IPv6) Specification. RFC 2460. (Dec. 1998). <https://tools.ietf.org/html/rfc2460>
- [14] Google Developers. 2017. YouTube Data API. (2017). Retrieved June 04, 2017 from <https://developers.google.com/youtube/v3/docs/videos/list>
- [15] Amogh Dhamdhere, Matthew J. Luckie, Bradley Huffaker, kc claffy, Ahmed Elmokashfi, and Emile Aben. 2012. Measuring the deployment of IPv6: topology, routing and performance. In *Proceedings of the 12th ACM SIGCOMM Internet Measurement Conference, IMC '12, Boston, MA, USA, November 14-16, 2012*. 537–550. <https://doi.org/10.1145/2398776.2398832>
- [16] Amsterdam Internet Exchange. 2017. IPv6 Traffic. (2017). Retrieved June 04, 2017 from <https://ams-ix.net/technical/statistics/sflow-stats/ipv6-traffic>
- [17] Alessandro Finamore, Marco Mellia, Maurizio M. Munafò, Ruben Torres, and Sanjay G. Rao. 2011. YouTube everywhere: impact of device and infrastructure synergies on user experience. In *Proceedings of the 11th ACM SIGCOMM Internet Measurement Conference, IMC '11, Berlin, Germany, November 2-, 2011*. 345–360. <https://doi.org/10.1145/2068816.2068849>
- [18] Phillipa Gill, Martin F. Arlitt, Zongpeng Li, and Anirban Mahanti. 2007. Youtube traffic characterization: a view from the edge. In *Proceedings of the 7th ACM SIGCOMM Internet Measurement Conference, IMC 2007, San Diego, California, USA, October 24-26, 2007*. <https://doi.org/10.1145/1298306.1298310>
- [19] Google. 2017. IPv6 Adoption Statistics. (2017). Retrieved June 04, 2017 from <https://www.google.com/intl/en/ipv6/statistics.html>
- [20] Google. 2017. Peering and Content Delivery. (2017). Retrieved June 04, 2017 from <https://peering.google.com/about/ggc.html>
- [21] C. Huitema. 2006. Teredo: Tunneling IPv6 over UDP through NATs. RFC 4380. (Feb. 2006). <https://tools.ietf.org/html/rfc4380>
- [22] IETF. 2017. IPv6 Operations (v6ops) Charter. (2017). Retrieved June 04, 2017 from <https://datatracker.ietf.org/wg/v6ops/charter>
- [23] Parikshit Juluri, Louis Plissonneau, and Deep Medhi. 2011. Pytomo: A tool for analyzing playback quality of YouTube videos. In *23rd International Teletraffic Congress, ITC 2011, San Francisco, CA, USA, September 6-9, 2011*. 304–305. <http://ieeexplore.ieee.org/document/6038496/>
- [24] Parikshit Juluri, Louis Plissonneau, Yong Zeng, and Deep Medhi. 2013. Viewing YouTube from a metropolitan area: What do users accessing from residential ISPs experience?. In *2013 IFIP/IEEE International Symposium on Integrated Network Management (IM 2013), Ghent, Belgium, May 27-31, 2013*. 589–595. <http://ieeexplore.ieee.org/document/6573037/>
- [25] Parikshit Juluri, Venkatesh Tamarapalli, and Deep Medhi. 2016. Measurement of Quality of Experience of Video-on-Demand Services: A Survey. *IEEE Communications Surveys and Tutorials* 18, 1 (2016), 401–418. <https://doi.org/10.1109/COMST.2015.2401424>
- [26] kc claffy. 2016. The 7th Workshop on Active Internet Measurements (AIMS7) Report. *Computer Communication Review* 46, 1 (2016), 50–57. <https://doi.org/10.1145/2875951.2875960>
- [27] Leone. 2015. From global measurements to local management. (2015). Retrieved June 04, 2017 from http://leone-project.eu/drupal/Leone_Final_Report_Part_A-Publishable_Summary.pdf
- [28] Ioana Livadariu, Ahmed Elmokashfi, and Amogh Dhamdhere. 2016. Characterizing IPv6 control and data plane stability. In *35th Annual IEEE International Conference on Computer Communications, INFOCOM 2016, San Francisco, CA, USA, April 10-14, 2016*. <https://doi.org/10.1109/INFOCOM.2016.7524465>
- [29] Aemen Lodhi, Natalie Larson, Amogh Dhamdhere, Constantine Dovrolis, and kc claffy. 2014. Using peeringDB to understand the peering ecosystem. *Computer Communication Review* 44, 2 (2014), 20–27. <https://doi.org/10.1145/2602204.2602208>
- [30] Hyunwoo Nam, Kyung-Hwa Kim, and Henning Schulzrinne. 2016. QoE matters more than QoS: Why people stop watching cat videos. In *35th Annual IEEE International Conference on Computer Communications, INFOCOM 2016, San Francisco, CA, USA, April 10-14, 2016*. 1–9. <https://doi.org/10.1109/INFOCOM.2016.7524426>
- [31] NANOG. 2016. IPv6 traffic percentages? (2016). Retrieved June 04, 2017 from <https://mailman.nanog.org/pipermail/nanog/2016-January/083624.html>
- [32] Mehdi Nikkhab, Roch Guérin, Yiu Lee, and Richard Woundy. 2011. Assessing IPv6 through web access a measurement study and its findings. In *Proceedings of the 2011 Conference on Emerging Networking Experiments and Technologies, Co-NEXT '11, Tokyo, Japan, December 6-9, 2011*. 26. <https://doi.org/10.1145/2079296.2079322>
- [33] Philipp Richter, Mark Allman, Randy Bush, and Vern Paxson. 2015. A Primer on IPv4 Scarcity. *Computer Communication Review* 45, 2 (2015), 21–31. <https://doi.org/10.1145/2766330.2766335>
- [34] Nadi Sarrar, Gregor Maier, Bernhard Ager, Robin Sommer, and Steve Uhlig. 2012. Investigating IPv6 Traffic - What Happened at the World IPv6 Day?. In *Passive and Active Measurement - 13th International Conference, PAM 2012, Vienna, Austria, March 12-14th, 2012. Proceedings*. 11–20. https://doi.org/10.1007/978-3-642-28537-0_2
- [35] Internet Society. 2012. World IPv6 Launch. (2012). Retrieved June 04, 2017 from <http://www.worldipv6launch.org>
- [36] Thomas Stockhammer. 2011. Dynamic adaptive streaming over HTTP -: standards and design principles. In *Proceedings of the Second Annual ACM SIGMM Conference on Multimedia Systems, MMSys 2011, Santa Clara, CA, USA, February 23-25, 2011*. 133–144. <https://doi.org/10.1145/1943552.1943572>
- [37] D. Thaler, R. Draves, A. Matsumoto, and T. Chown. 2012. Default Address Selection for Internet Protocol Version 6 (IPv6). RFC 6724. (Sept. 2012). <https://tools.ietf.org/html/rfc6724>
- [38] D. Wing and A. Youtchenko. 2012. Happy Eyeballs: Success with Dual-Stack Hosts. RFC 6555. (2012). <https://tools.ietf.org/html/rfc6555>