

TAPS
Internet-Draft
Intended status: Informational
Expires: January 9, 2017

K-J. Grinnemo
A. Brunstrom
P. Hurtig
Karlstad University
N. Khademi
University of Oslo
July 8, 2016

Happy Eyeballs for Transport Selection
draft-grinnemo-taps-he-00

Abstract

It is widely recognized that it has become very difficult to introduce new transport protocols on the Internet. One reason for this is that there are no agreed-upon way for a source end host to find out whether there is support for a particular transport protocol along the network path from itself to a destination end host or not. This document describes a Happy Eyeballs framework that generalizes previously proposed Happy Eyeballs mechanisms to include a transport protocol, the configuration of this transport protocol, as well as a transport address, i.e., to include complete transport solutions. The described Happy Eyeballs framework is not only useful in the design and implementation of mechanisms that are to determine the support of particular transport solutions, but also for the design of mechanisms that are to select the transport solution, which, according to some criteria, is the most appropriate one to use.

Status of This Memo

This Internet-Draft is submitted in full conformance with the provisions of [BCP 78](#) and [BCP 79](#).

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at <http://datatracker.ietf.org/drafts/current/>.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on January 9, 2017.

Internet-Draft Happy Eyeballs for Transport Selection

July 2016

Copyright Notice

Copyright (c) 2016 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to [BCP 78](#) and the IETF Trust's Legal Provisions Relating to IETF Documents (<http://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

1.	Definitions	2
2.	Introduction	2
3.	Problem Statement	3
4.	The Happy Eyeballs Framework	4
5.	Design and Implementation Considerations	5
5.1.	Caching	5
5.2.	Non-Winning Transport Solutions	5
6.	Example Happy Eyeballs Mechanism	5
7.	IANA Considerations	6
8.	Security Considerations	6
9.	Acknowledgements	6
10.	References	6
10.1.	Normative References	6
10.2.	Informative References	6
	Authors' Addresses	7

[1.](#) Definitions

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in [RFC 2119](#) [[RFC2119](#)].

[2.](#) Introduction

Concerns have been raised in the past several years that introducing new transport protocols on the Internet has become increasingly

difficult, not least because there is no agreed-upon way for a source end host to find out if a transport protocol is supported all the way to a destination peer. Of course, testing a set of candidate protocols can be done serially, e.g., try first with the preferred choice, X, and, if the connection attempt fails after a suitable

timeout, fall back to a default alternative, Y. However, since a connection timeout can introduce a delay of up to tens of seconds, serializing attempts can incur a large delay penalty when the new protocol, X, is not supported, stalling the application until the subsequent connection attempt succeeds. A solution to a similar problem, finding out support for IPv6, has been proposed and is currently being deployed, the Happy Eyeballs (HE) mechanism [[RFC6555](#)]: The HE mechanism was introduced as a means to facilitate IPv6 adoption. Dual-stack client applications should be encouraged to try setting up connections over IPv6 first, and fall back to using IPv4 if IPv6 connection attempts fail. However, serializing tests for IPv6 and IPv4 connectivity can result in large connection latencies. HE for IPv6 minimizes the cost in delay by parallelizing attempts over IPv6 and IPv4. HE has also been proposed as an efficient way to find out the optimal combination of IPv4/IPv6 and TCP/SCTP to use to connect to a server [[I-D.wing-v6ops-happy-eyeballs-ipv6](#)]. This document suggests a further generalization of the transport HE mechanism proposed in Wing et al. [[I-D.wing-v6ops-happy-eyeballs-ipv6](#)] which addresses the selection of arbitrary transport solutions, i.e., arbitrary combinations of transport protocol, transport address, and pertaining configurations, and which lend itself to arbitrary transport selection criterias, not only support for a particular transport solution. Particularly, this document provides a HE framework from which a transport selection mechanism could be realized that selects, from supported transport solutions, the one that according to some criteria is the most appropriate one.

[3.](#) Problem Statement

There is, as mentioned in [Section 2](#), no agreed-upon way for a source end host to find out if a transport protocol is supported along a network path between itself and a destination end host. As a consequence, it has become increasingly difficult to introduce new transport protocols. This is further aggravated by the fact that several applications, including many web applications, run over TCP

although there are other transport solutions that better meet the requirements of these applications. Another, related problem, addressed by our proposed HE framework is how to select the transport solution that is, according to some criteria, the most appropriate one for a given application. In fact, our HE framework provides guidelines on how to design and implement a HE transport selection mechanism that combines these two problems by selecting, from among supported transport solutions, the one that best meets a given criteria.

4. The Happy Eyeballs Framework

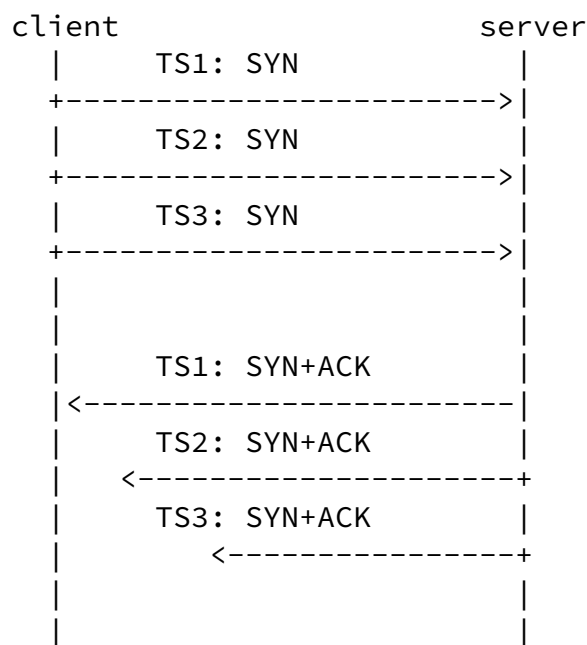


Figure 1: Illustration of Happy Eyeballing between three TCP transport solutions.

The HE framework takes as input a list of candidate transport solutions, *L*, sorted in decreasing priority order. The exact scale used for the priority is beyond the scope of the framework, however, since the difference between priorities should be meaningful, it must

at least be an interval scale. The HE framework consists of the following two steps (cf. Figure 1):

1. Spawn in priority order connection attempts for each candidate transport solution in L. The difference in priority between two consecutive candidates, C1 and C2, is translated according to some criteria to a delay, D. D then governs the delay between the spawn of C1 and C2. The way connection attempts are spawned is not within the scope of the HE framework, and could, for example, be realized through threads or asynchronous I/O.
2. Wait for the first connection, W, to be established. W becomes the winning transport solution. The action taken with the remaining, non-winning, transport solutions is beyond the scope of the HE framework.

[5.](#) Design and Implementation Considerations

This section discusses implementation issues that should be considered when a HE mechanism is designed and implemented on the basis of the HE framework proposed in this document.

[5.1.](#) Caching

As pointed out in [RFC 6555](#) [[RFC6555](#)], a HE algorithm should not waste networking resources by routinely making simultaneous connection attempts. To this end, the HE algorithm should cache the outcome of previous connection attempts. Cached connection attempts should be valid for a configurable time after which they should become invalid and have to be repeated. The impact and efficiency of the HE algorithm has been evaluated in a publication by us [[Papastergiou16](#)]. The paper suggests that caching significantly reduces the CPU load imposed by a HE mechanism.

[5.2.](#) Non-Winning Transport Solutions

As mentioned in [Section 4](#), the management of the non-winning transport solutions is not within the scope of the HE framework.

Still, it is recommended that all non-winning transport solutions are abandoned. Moreover, if caching is used, we recommend that the outcome of the connection attempt of the non-winning transport solution is cached.

6. Example Happy Eyeballs Mechanism

Consider the scenario of a dual-stack IPv4/IPv6 client trying to setup a connection to a dual-stack IPv4/IPv6 server. Assume the client and server support both TCP and SCTP. A list of candidate transport solutions is put together that contains TCP and SCTP over IPv6 (with equal priority) followed by TCP and SCTP over IPv4 (with equal priority), but the latter have a lower priority than their IPv6 counterparts. The HE mechanism is invoked; traverses the candidate list, and spawns a separate thread for each candidate transport solution. In our example, two threads are created for TCP and SCTP over IPv4, and two threads for TCP and SCTP over IPv6. Since the IPv4 transport solutions have a lower priority than the IPv6 ones, the creation of the threads for the IPv4 transport solutions are delayed with respect to the IPv6 transport solution threads. The length of the delay depends on the size of the difference in priority. Within each thread, a connection attempt is made, and if the connection attempt is successful, the thread returns a connection handle. Otherwise, it signals a failure by returning an invalid connection handle. In both cases the outcome (connection attempt success or failure) is cached. Assume that in our example all

connection attempts succeed and therefore will be cached as successful connection attempts. Since the IPv6 connection attempts were started earlier than IPv4 counterparts, one of these attempts will be selected as the winning transport solution.

7. IANA Considerations

XX RFC ED - PLEASE REMOVE THIS SECTION XXX

This memo includes no request to IANA.

8. Security Considerations

Security will be considered in future versions of this document.

9. Acknowledgements

This work has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 644334 (NEAT). The views expressed are solely those of the author(s).

10. References

10.1. Normative References

- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", [BCP 14](#), [RFC 2119](#), DOI 10.17487/RFC2119, March 1997, <<http://www.rfc-editor.org/info/rfc2119>>.
- [RFC6555] Wing, D. and A. Yourtchenko, "Happy Eyeballs: Success with Dual-Stack Hosts", [RFC 6555](#), DOI 10.17487/RFC6555, April 2012, <<http://www.rfc-editor.org/info/rfc6555>>.

10.2. Informative References

- [I-D.wing-v6ops-happy-eyeballs-ipv6]
Wing, D. and A. Yourtchenko, "Happy Eyeballs: Trending Towards Success with Dual-Stack Hosts", [draft-wing-v6ops-happy-eyeballs-ipv6-01](#) (work in progress), October 2010.
- [Papastergiou16]
Papastergiou, G., Grinnemo, K-J., Brunstrom, A., Ros, D., Tuexen, M., Khademi, N., and P. Hurtig, "On the Cost of Using Happy Eyeballs for Transport Protocol Selection", July 2016.

Authors' Addresses

Karl-Johan Grinnemo
Karlstad University
Universitetsgatan 2
Karlstad 651 88
Sweden

Phone: +46 54 700 24 40
Email: karl-johan.grinnemo@kau.se

Anna Brunstrom
Karlstad University
Universitetsgatan 2
Karlstad 651 88
Sweden

Phone: +46 54 700 17 95
Email: anna.brunstrom@kau.se

Per Hurtig
Karlstad University
Universitetsgatan 2
Karlstad 651 88
Sweden

Phone: +46 54 700 23 35
Email: per.hurtig@kau.se

Naeem Khademi
University of Oslo
PO Box 1080 Blindern
Oslo N-0316
Norway

Phone: +47 2285 24 93
Email: naeemk@ifi.uio.no