

IPv6 Operations Working Group (v6ops)  
Internet-Draft  
Intended status: Informational  
Expires: October 23, 2015

F. Gont  
SI6 Networks / UTN-FRH  
J. Linkova  
Google  
T. Chown  
University of Southampton  
W. Liu  
Huawei Technologies  
April 21, 2015

**Observations on IPv6 EH Filtering in the Real World**  
**draft-ietf-v6ops-ipv6-ehs-in-real-world-00**

Abstract

This document presents real-world data regarding the extent to which packets with IPv6 extension headers are filtered in the Internet (as measured in August 2014), and where in the network such filtering occurs. The aforementioned results serve as a problem statement that is expected to trigger operational advice on the filtering of IPv6 packets carrying IPv6 Extension Headers, so that the situation improves over time. This document also explains how the aforementioned results were obtained, such that the corresponding measurements can be reproduced by other members of the community.

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 October 23, 2015.

Copyright Notice

Copyright (c) 2015 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.](#) Introduction . . . . . [2](#)
- [2.](#) Support of IPv6 Extension Headers in the Internet . . . . . [3](#)
- [3.](#) IANA Considerations . . . . . [7](#)
- [4.](#) Security Considerations . . . . . [7](#)
- [5.](#) Acknowledgements . . . . . [7](#)
- [6.](#) References . . . . . [7](#)
  - [6.1.](#) Normative References . . . . . [7](#)
  - [6.2.](#) Informative References . . . . . [8](#)
- [Appendix A.](#) Reproducing Our Experiment . . . . . [9](#)
  - [A.1.](#) Obtaining the List of Domain Names . . . . . [9](#)
  - [A.2.](#) Obtaining AAAA Resource Records . . . . . [10](#)
  - [A.3.](#) Filtering the IPv6 Address Datasets . . . . . [10](#)
  - A.4. Performing Measurements with Each IPv6 Address Dataset . 11
  - [A.5.](#) Obtaining Statistics from our Measurements . . . . . [12](#)
- [Appendix B.](#) Measurements Caveats . . . . . [13](#)
  - [B.1.](#) Isolating the Dropping Node . . . . . [13](#)
  - B.2. Obtaining the Responsible Organization for the Packet Drops . . . . . [14](#)
- [Appendix C.](#) Troubleshooting Packet Drops due to IPv6 Extension Headers . . . . . [15](#)
- Authors' Addresses . . . . . [15](#)

**1. Introduction**

IPv6 Extension Headers (EHs) allow for the extension of the IPv6 protocol, and provide support for core functionality such as IPv6 fragmentation. While packets employing IPv6 Extension Headers have been suspected to be dropped in some IPv6 deployments, there was not much concrete data on the topic. Some preliminary measurements have been presented in [[PMTUD-Blackholes](#)], [[Gont-IEPG88](#)] and [[Gont-Chown-IEPG89](#)], whereas [[Linkova-Gont-IEPG90](#)] presents more comprehensive results on which this document is based.

This document presents real-world data regarding the extent to which packets containing IPv6 Extension Headers are filtered in the Internet, as measured in August 2014 (pending operational advice in



this area). The results presented in this document indicate that in the scenarios where the corresponding measurements were performed, the use of IPv6 extension headers can lead to packet drops. We note that, in particular, packet drops occurring at transit networks are undesirable, and it is hoped and expected that this situation will improve over time.

## 2. Support of IPv6 Extension Headers in the Internet

This section summarizes the results obtained when measuring the support of IPv6 Extension Headers on the path towards different types of public IPv6 servers. Two sources of information were employed for the list of public IPv6 servers: the "World IPv6 Launch Day" site (<http://www.worldipv6launch.org/>) and Alexa's top 1M web sites (<http://www.alexa.com>). For each list of domain names, the following datasets were obtained:

- o Web servers (AAAA records of the aforementioned list)
- o Mail servers (MX -> AAAA of the aforementioned list)
- o Name servers (NS -> AAAA of the aforementioned list)

Duplicate addresses and IPv6 addresses other than global unicast addresses were eliminated from each of those lists prior to obtaining the results included in this document. Additionally, addresses that were found to be unreachable were discarded from the dataset (please see [Appendix B](#) for further details).

For each of the aforementioned address sets, three different types of probes were employed:

- o IPv6 packets with a Destination Options header of 8 bytes
- o IPv6 packets resulting in two IPv6 fragments of 512 bytes each (approximately)
- o IPv6 packets with a Hop-by-Hop Options header of 8 bytes

In the case of packets with a Destination Options Header and the case of packets with a Hop-by-Hop Options header, the desired EH size was achieved by means of PadN options [[RFC2460](#)]. The upper-layer protocol of the probe packets was, in all cases, TCP [[RFC0793](#)] segments with the Destination Port set to the service port [[IANA-PORT-NUMBERS](#)] of the corresponding dataset. For example, the probe packets for all the measurements involving web servers were TCP segments with the destination port set to 80.



Besides obtaining the packet drop rate when employing the aforementioned IPv6 extension headers, we tried to identify whether the Autonomous System (AS) dropping the packets was the same as the Autonomous System of the destination/target address. This is of particular interest since it essentially reveals whether the packet drops are under the control of the intended destination of the packets. Packets dropped by the destination AS are less of a concern, since the device dropping the packets is under the control of the same organization as that to which the packets are destined (hence, it is probably easier to update the filtering policy if deemed necessary). On the other hand, packets dropped by transit ASes are more of a concern, since they affect the deployability and usability of IPv6 extension headers (including IPv6 fragmentation) by a third-party (the destination AS). In any case, we note that it is impossible to tell whether, in those cases where IPv6 packets with extension headers get dropped, the packet drops are the result of an explicit and intended policy, or the result of improper device configuration defaults, buggy devices, etc. Thus, packet drops that occur at the destination AS might still prove to be problematic.

Since there is some ambiguity when identifying the autonomous system to which a specific router belongs (see [Appendix B.2](#)), each of our measurements results in two different values: one corresponding to the "best-case scenario", and one corresponding to the "worst-case scenario". The "best-case scenario" is that in which, when in doubt, the packets are assumed to be dropped by the destination AS, whereas the "worst-case scenario" is that in which, when in doubt, the packets are assumed to be dropped by a transit AS (please see [Appendix B.2](#) for details). In the following tables, the values shown within parentheses represent the possibility that, when a packet is dropped, the packet drop occurs in an AS other than the destination AS (considering both the best-case scenario and the worst-case scenario).



Dataset	D08	HBH8	FH512
Webservers	11.88% (17.60%/20.80%)	40.70% (31.43%/40.00%)	30.51% (5.08%/6.78%)
Mailservers	17.07% (6.35%/26.98%)	48.86% (40.50%/65.42%)	39.17% (2.91%/12.73%)
Nameservers	15.37% (14.29%/33.46%)	43.25% (42.49%/72.07%)	38.55% (3.90%/13.96%)

Table 1: WIPv6LD dataset: Packet drop rate for different destination types, and estimated percentage of dropped packets that were deemed to be dropped in a different AS (lower, in parentheses)

NOTE: As an example, we note that the cell describing the support of IPv6 packets with D08 for web servers (containing the value "11.88% (17.60%/20.80%)") should be read as: "when sending IPv6 packets with D08 to public web servers, 11.88% of such packets get dropped. Among those packets that get dropped, 17.60%/20.80% (best case / worst case) of them get dropped at an AS other than the destination AS".

EH Type	Webservers	Mailservers	Nameservers
D08	11.88% (17.60%/20.80%)	17.07% (6.35%/26.98%)	15.37% (14.29%/33.46%)
HBH8	40.70% (31.43%/40.00%)	48.86% (40.50%/65.42%)	43.25% (42.49%/72.07%)
FH512	30.51% (5.08%/6.78%)	39.17% (2.91%/12.73%)	38.55% (3.90%/13.96%)

Table 2: WIPv6LD dataset: Packet drop rate for different EH types, and estimated percentage of dropped packets that were deemed to be dropped in a different AS (lower, in parentheses)

NOTE: This table contains the same information as Table 1, but makes it easier to obtain the drop rates for each EH type. Each cell should be read in exactly the same way as each cell in Table 1.





Dataset	D08	HBH8	FH512
Webservers	10.91% (46.52%/53.23%)	39.03% (36.90%/46.35%)	28.26% (53.64%/61.43%)
Mailservers	11.54% (2.41%/21.08%)	45.45% (41.27%/61.13%)	35.68% (3.15%/10.92%)
Nameservers	21.33% (10.27%/56.80%)	54.12% (50.64%/81.00%)	55.23% (5.66%/32.23%)

Table 3: Alexa's top 1M sites dataset: Packet drop rate for different destination types, and estimated percentage of dropped packets that were deemed to be dropped in a different AS (lower, in parentheses)

EH Type	Webservers	Mailservers	Nameservers
D08	10.91% (46.52%/53.23%)	11.54% (2.41%/21.08%)	21.33% (10.27%/56.80%)
HBH8	39.03% (36.90%/46.35%)	45.45% (41.27%/61.13%)	54.12% (50.64%/81.00%)
FH512	28.26% (53.64%/61.43%)	35.68% (3.15%/10.92%)	55.23% (5.66%/32.23%)

Table 4: Alexa's top 1M sites dataset: Packet drop rate for different EH types, and estimated percentage of dropped packets that were deemed to be dropped in a different AS (lower, in parentheses)

NOTE: This table contains the same information as Table 3, but makes it easier to obtain the drop rates for each EH type. Each cell should be read in exactly the same way as each cell in Table 3.

There are a number of observations to be made based on the results presented above. Firstly, while it has been generally assumed that it is IPv6 fragments that are dropped by operators, our results indicate that it is IPv6 extension headers in general that result in packet drops. Secondly, our results indicate that a significant percentage of such packet drops occurs in transit Autonomous Systems; that is, the packet drops are not under the control of the same organization as the final destination.



### **3. IANA Considerations**

There are no IANA registries within this document. The RFC-Editor can remove this section before publication of this document as an RFC.

### **4. Security Considerations**

This document presents real-world data regarding the extent to which IPv6 packets employing extension headers are filtered in the Internet. As such, this document does not introduce any new security issues.

### **5. Acknowledgements**

The authors would like to thank (in alphabetical order) Mikael Abrahamsson, Mark Andrews, Fred Baker, Brian Carpenter, Gert Doering, C. M. Heard, Nick Hilliard, Joel Jaeggli, Tatuya Jinmei, Merike Kaeo, Warren Kumari, Mark Smith, Ole Troan, and Eric Vyncke, for providing valuable comments on earlier versions of this document. Additionally, the authors would like to thank participants of the v6ops and opsec working groups for their valuable input on the topics discussed in this document.

The authors would like to thank Fred Baker for his guidance in improving this document.

Fernando Gont would like to thank Jan Zorz / Go6 Lab <<http://go6lab.si/>>, and Jared Mauch / NTT America, for providing access to systems and networks that were employed to produce some of the measurement results presented in this document. Additionally, he would like to thank SixXS <<https://www.sixxs.net>> for providing IPv6 connectivity.

### **6. References**

#### **6.1. Normative References**

- [RFC0793] Postel, J., "Transmission Control Protocol", STD 7, [RFC 793](#), September 1981.
- [RFC1034] Mockapetris, P., "Domain names - concepts and facilities", STD 13, [RFC 1034](#), November 1987.
- [RFC2460] Deering, S. and R. Hinden, "Internet Protocol, Version 6 (IPv6) Specification", [RFC 2460](#), December 1998.



- [RFC4443] Conta, A., Deering, S., and M. Gupta, "Internet Control Message Protocol (ICMPv6) for the Internet Protocol Version 6 (IPv6) Specification", [RFC 4443](#), March 2006.
- [RFC4861] Narten, T., Nordmark, E., Simpson, W., and H. Soliman, "Neighbor Discovery for IP version 6 (IPv6)", [RFC 4861](#), September 2007.
- [RFC6145] Li, X., Bao, C., and F. Baker, "IP/ICMP Translation Algorithm", [RFC 6145](#), April 2011.
- [RFC6946] Gont, F., "Processing of IPv6 "Atomic" Fragments", [RFC 6946](#), May 2013.

## **[6.2.](#) Informative References**

- [Gont-Chown-IEPG89]  
Gont, F. and T. Chown, "A Small Update on the Use of IPv6 Extension Headers", IEPG 89. London, UK. March 2, 2014, <<http://www.iepg.org/2014-03-02-ietf89/fgont-iepg-ietf89-eh-update.pdf>>.
- [Gont-IEPG88]  
Gont, F., "Fragmentation and Extension header Support in the IPv6 Internet", IEPG 88. Vancouver, BC, Canada. November 13, 2013, <<http://www.iepg.org/2013-11-ietf88/fgont-iepg-ietf88-ipv6-frag-and-eh.pdf>>.
- [IANA-PORT-NUMBERS]  
IANA, "Service Name and Transport Protocol Port Number Registry", <<http://www.iana.org/assignments/service-names-port-numbers/service-names-port-numbers.txt>>.
- [IPv6-Toolkit]  
"SI6 Networks' IPv6 Toolkit", <<http://www.si6networks.com/tools/ipv6toolkit>>.
- [Linkova-Gont-IEPG90]  
Linkova, J. and F. Gont, "IPv6 Extension Headers in the Real World v2.0", IEPG 90. Toronto, ON, Canada. July 20, 2014, <<http://www.iepg.org/2014-07-20-ietf90/iepg-ietf90-ipv6-ehs-in-the-real-world-v2.0.pdf>>.



**[PMTUD-Blackholes]**

De Boer, M. and J. Bosma, "Discovering Path MTU black holes on the Internet using RIPE Atlas", July 2012, <<http://www.nlnetlabs.nl/downloads/publications/pmtu-black-holes-msc-thesis.pdf>>.

[RFC5927] Gont, F., "ICMP Attacks against TCP", [RFC 5927](#), July 2010.

[RFC6980] Gont, F., "Security Implications of IPv6 Fragmentation with IPv6 Neighbor Discovery", [RFC 6980](#), August 2013.

[RFC7045] Carpenter, B. and S. Jiang, "Transmission and Processing of IPv6 Extension Headers", [RFC 7045](#), December 2013.

[RFC7113] Gont, F., "Implementation Advice for IPv6 Router Advertisement Guard (RA-Guard)", [RFC 7113](#), February 2014.

[RFC7123] Gont, F. and W. Liu, "Security Implications of IPv6 on IPv4 Networks", [RFC 7123](#), February 2014.

**[blackhole6]**

blackhole6, , "blackhole6 tool manual page", <<http://www.si6networks.com/tools/ipv6toolkit>>, 2014.

[path6] path6, , "path6 tool manual page", <<http://www.si6networks.com/tools/ipv6toolkit>>, 2014.

**Appendix A. Reproducing Our Experiment**

This section describes, step by step, how to reproduce the experiment with which we obtained the results presented in this document. Each subsection represents one step in the experiment. The tools employed for the experiment are traditional UNIX-like tools (such as gunzip), and the SI6 Networks' IPv6 Toolkit [[IPv6-Toolkit](#)].

**A.1. Obtaining the List of Domain Names**

The primary data source employed was Alexa's Top 1M web sites, available at: <<http://s3.amazonaws.com/alexa-static/top-1m.csv.zip>>. The file is a zipped file containing the list of the most popular web sites, in CSV format. The aforementioned file can be extracted with "gunzip < top-1m.csv.zip > top-1m.csv".

A list of domain names (i.e., other data stripped) can be obtained with the following command of [[IPv6-Toolkit](#)]: "cat top-1m.csv | script6 get-alexa-domains > top-1m.txt". This command will create a "top-1m.txt" file, containing one domain name per line.





NOTE: The domain names corresponding to the WIPv6LD dataset is available at: <<http://www.sixnetworks.com/datasets/wipv6day-domains.txt>>. Since the corresponding file is a text file containing one domain name per line, the steps produced in this subsection need not be performed. The WIPv6LD data set should be processed in the same way as the Alexa Dataset, starting from [Appendix A.2](#).

### **A.2. Obtaining AAAA Resource Records**

The file obtained in the previous subsection contains a list of domain names that correspond to web sites. The AAAA records for such domain names can be obtained with:

```
$ cat top-1m.txt | script6 get-aaaa > top-1m-web-aaaa.txt
```

The AAAA records corresponding to the mailservers of each of the aforementioned domain names can be obtained with:

```
$ cat top-1m.txt | script6 get-mx | script6 get-aaaa > top-1m-mail-aaaa.txt
```

The AAAA records corresponding to the nameservers of each of the aforementioned domain names can be obtained with:

```
$ cat top-1m.txt | script6 get-ns | script6 get-aaaa > top-1m-dns-aaaa.txt
```

### **A.3. Filtering the IPv6 Address Datasets**

The lists of IPv6 addresses obtained in the previous step could possibly contain undesired addresses (i.e., non-global unicast addresses) and/or duplicate addresses. In order to remove both undesired and duplicate addresses, each of the three files from the previous section should be filtered accordingly:

```
$ cat top-1m-web-aaaa.txt | addr6 -i -q -B multicast -B unspec -k global > top-1m-web-aaaa-unique.txt
```

```
$ cat top-1m-mail-aaaa.txt | addr6 -i -q -B multicast -B unspec -k global > top-1m-mail-aaaa-unique.txt
```

```
$ cat top-1m-dns-aaaa.txt | addr6 -i -q -B multicast -B unspec -k global > top-1m-dns-aaaa-unique.txt
```



#### **[A.4.](#) Performing Measurements with Each IPv6 Address Dataset**

##### **[A.4.1.](#) Measurements with web servers**

In order to measure D08 with the list of webservers:

```
# cat top-1m-web-aaaa-unique.txt | script6 trace6 do8 tcp 80 > top-1m-web-aaaa-do8-m.txt
```

In order to measure HBH8 with the list of webservers:

```
# cat top-1m-web-aaaa-unique.txt | script6 trace6 hbh8 tcp 80 > top-1m-web-aaaa-hbh8-m.txt
```

In order to measure FH512 with the list of webservers:

```
# cat top-1m-web-aaaa-unique.txt | script6 trace6 fh512 tcp 80 > top-1m-web-aaaa-fh512-m.txt
```

##### **[A.4.2.](#) Measurements with mail servers**

In order to measure D08 with the list of mailservers:

```
# cat top-1m-mail-aaaa-unique.txt | script6 trace6 do8 tcp 25 > top-1m-mail-aaaa-do8-m.txt
```

In order to measure HBH8 with the list of webservers:

```
# cat top-1m-mail-aaaa-unique.txt | script6 trace6 hbh8 tcp 25 > top-1m-mail-aaaa-hbh8-m.txt
```

In order to measure FH512 with the list of webservers:

```
# cat top-1m-mail-aaaa-unique.txt | script6 trace6 fh512 tcp 25 > top-1m-mail-aaaa-fh512-m.txt
```

##### **[A.4.3.](#) Measurements with DNS servers**

In order to measure D08 with the list of nameservers:

```
# cat top-1m-dns-aaaa-unique.txt | script6 trace6 do8 tcp 53 > top-1m-dns-aaaa-do8-m.txt
```

In order to measure HBH8 with the list of webservers:

```
# cat top-1m-dns-aaaa-unique.txt | script6 trace6 hbh8 tcp 53 > top-1m-dns-aaaa-hbh8-m.txt
```



In order to measure FH512 with the list of webservers:

```
# cat top-1m-dns-aaaa-unique.txt | script6 trace6 fh512 tcp 53 > top-1m-dns-aaaa-fh512-m.txt
```

## **[A.5.](#) Obtaining Statistics from our Measurements**

### **[A.5.1.](#) Statistics for Web Servers**

In order to compute the statistics corresponding to our measurements of D08 with the list of webservers:

```
$ cat top-1m-web-aaaa-do8-m.txt | script6 get-trace6-stats > top-1m-web-aaaa-do8-stats.txt
```

In order to compute the statistics corresponding to our measurements of HBH8 with the list of webservers:

```
$ cat top-1m-web-aaaa-hbh8-m.txt | script6 get-trace6-stats > top-1m-web-aaaa-hbh8-stats.txt
```

In order to compute the statistics corresponding to our measurements of FH512 with the list of webservers:

```
$ cat top-1m-web-aaaa-fh512-m.txt | script6 get-trace6-stats > top-1m-web-aaaa-fh512-stats.txt
```

### **[A.5.2.](#) Statistics for Mail Servers**

In order to compute the statistics corresponding to our measurements of D08 with the list of mailservers:

```
$ cat top-1m-mail-aaaa-do8-m.txt | script6 get-trace6-stats > top-1m-mail-aaaa-do8-stats.txt
```

In order to compute the statistics corresponding to our measurements of HBH8 with the list of mailservers:

```
$ cat top-1m-mail-aaaa-hbh8-m.txt | script6 get-trace6-stats > top-1m-mail-aaaa-hbh8-stats.txt
```

In order to compute the statistics corresponding to our measurements of FH512 with the list of mailservers:

```
$ cat top-1m-mail-aaaa-fh512-m.txt | script6 get-trace6-stats > top-1m-mail-aaaa-fh512-stats.txt
```



### **[A.5.3.](#) Statistics for Name Servers**

In order to compute the statistics corresponding to our measurements of D08 with the list of nameservers:

```
$ cat top-1m-dns-aaaa-do8-m.txt | script6 get-trace6-stats > top-1m-dns-aaaa-do8-stats.txt
```

In order to compute the statistics corresponding to our measurements of HBH8 with the list of mailservers:

```
$ cat top-1m-dns-aaaa-hbh8-m.txt | script6 get-trace6-stats > top-1m-dns-aaaa-hbh8-stats.txt
```

In order to compute the statistics corresponding to our measurements of FH512 with the list of mailservers:

```
$ cat top-1m-dns-aaaa-fh512-m.txt | script6 get-trace6-stats > top-1m-dns-aaaa-fh512-stats.txt
```

## **[Appendix B.](#) Measurements Caveats**

A number of issues have needed some consideration when producing the results presented in this document. These same issues should be considered when troubleshooting connectivity problems resulting from the use of IPv6 Extension headers.

### **[B.1.](#) Isolating the Dropping Node**

Let us assume that we find that IPv6 packets with EHs are being dropped on their way to the destination system 2001:db8:d::1, and that the output of running traceroute towards such destination is:

1. 2001:db8:1:1000::1
2. 2001:db8:2:4000::1
3. 2001:db8:3:4000::1
4. 2001:db8:3:1000::1
5. 2001:db8:4:4000::1
6. 2001:db8:4:1000::1
7. 2001:db8:5:5000::1
8. 2001:db8:5:6000::1
9. 2001:db8:d::1

Additionally, let us assume that the output of EH-enabled traceroute to the same destination is:





1. 2001:db8:1:1000::1
2. 2001:db8:2:4000::1
3. 2001:db8:3:4000::1
4. 2001:db8:3:1000::1
5. 2001:db8:4:4000::1

For the sake of brevity, let us refer to the last-responding node in the EH-enabled traceroute ("2001:db8:4:4000::1" in this case) as "M". Assuming that packets in both traceroutes employ the same path, we'll refer to "the node following the last responding node in the EH-enabled traceroute" ("2001:db8:4:1000::1" in our case), as "M+1", etc.

Based on traceroute information above, which node is the one actually dropping the EH-enabled packets will depend on whether the dropping node filters packets before making the forwarding decision, or after making the forwarding decision. If the former, the dropping node will be M+1. If the latter, the dropping node will be "M".

Throughout this document (and our measurements), we assume that those nodes filtering packets that carry IPv6 EHs apply their filtering policy, and only then, if necessary, forward the packets. Thus, in our example above the last responding node to the EH-enabled traceroute ("M") is "2001:db8:4:4000::1", and therefore we assume the dropping node to be "2001:db8:4:1000::1" ("M+1").

Additionally, we note that when isolating the dropping node we assume that both the EH-enabled and the EH-free traceroutes result in the same paths. However, this might not be the case.

## **B.2. Obtaining the Responsible Organization for the Packet Drops**

In order to identify the organization operating the dropping node, one would be tempted to lookup the ASN corresponding to the dropping node. However, assuming that M and M+1 are two peering routers, any of these two organizations could be providing the address space employed for such peering. Or, in the case of an Internet eXchange Point (IXP), the address space could correspond to the IXP AS, rather than to any of the participating ASes. Thus, the organization operating the dropping node (M+1) could be the AS for M+1, but it might as well be the AS for M+2. Only when the ASN for M+1 is the same as the ASN for M+2 we have certainty about who the responsible organization for the packet drops is (see slides 21-23 of [[Linkova-Gont-IEPG90](#)]).

In the measurement results presented in [Section 2](#), the aforementioned ambiguity results in a "best-case" and a "worst-case" scenario (rather than a single value): the lowest percentage value means that,



when in doubt, we assume the packet drops occur in the same AS as the destination; on the other hand, the highest percentage value means that, when in doubt, we assume the packet drops occur at different AS than the destination AS.

We note that the aforementioned ambiguity should also be considered when troubleshooting and reporting IPv6 packet drops, since identifying the organization responsible for the packet drops might prove to be a non-trivial task.

Finally, we note that a specific organization might be operating more than one Autonomous System. However, our measurements assume that different Autonomous System Numbers imply different organizations.

### **Appendix C. Troubleshooting Packet Drops due to IPv6 Extension Headers**

Isolating IPv6 blackholes essentially involves performing IPv6 traceroute for a destination system with and without IPv6 extension headers. The EH-free traceroute would provide the full working path towards a destination, while the EH-enabled traceroute would provide the address of the last-responding node for EH-enabled packets (say, "M"). In principle, one could isolate the dropping node by looking-up "M" in the EH-free traceroute, with the dropping node being "M+1" (see [Appendix B.1](#) for caveats).

At the time of this writing, most traceroute implementations do not support IPv6 extension headers. However, the path6 tool [[path6](#)] of [[IPv6-Toolkit](#)] provides such support. Additionally, the blackhole6 tool [[blackhole6](#)] of [[IPv6-Toolkit](#)] automates the troubleshooting process and can readily provide information such as: dropping node's IPv6 address, dropping node's Autonomous System, etc.

#### Authors' Addresses

Fernando Gont  
SI6 Networks / UTN-FRH  
Evaristo Carriego 2644  
Haedo, Provincia de Buenos Aires 1706  
Argentina

Phone: +54 11 4650 8472  
Email: [fgont@si6networks.com](mailto:fgont@si6networks.com)  
URI: <http://www.si6networks.com>



J. Linkova  
Google  
1600 Amphitheatre Parkway  
Mountain View, CA 94043  
USA

Email: [furry@google.com](mailto:furry@google.com)

Tim Chown  
University of Southampton  
Highfield  
Southampton , Hampshire S017 1BJ  
United Kingdom

Email: [tjc@ecs.soton.ac.uk](mailto:tjc@ecs.soton.ac.uk)

Will(Shucheng) Liu  
Huawei Technologies  
Bantian, Longgang District  
Shenzhen 518129  
P.R. China

Email: [liushucheng@huawei.com](mailto:liushucheng@huawei.com)

