

IPv6 maintenance Working Group (6man)
Internet-Draft
Intended status: Best Current Practice
Expires: September 1, 2017

F. Gont
SI6 Networks / UTN-FRH
G. Gont
SI6 Networks
M. Garcia Corbo
SITRANS
February 28, 2017

IPv6 Address Usage Recommendations
draft-gont-6man-address-usage-recommendations-01

Abstract

This document analyzes the security and privacy implications of IPv6 addresses based on a number of properties such as address scope, stability, and usage type. It analyzes what properties are desirable for some popular scenarios, and provides advice regarding the configuration and usage of such addresses.

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 September 1, 2017.

Copyright Notice

Copyright (c) 2017 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

Internet-Draft

IPv6 Address Recommendations

February 2017

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.	Terminology	3
3.	Predictability Considerations	3
4.	Address Scope Considerations	3
5.	Address Stability Considerations	4
6.	Usage Type Considerations	5
7.	Advice on IPv6 Address Configuration	6
8.	Advice on IPv6 Address Usage	7
9.	IANA Considerations	7
10.	Security Considerations	7
11.	Acknowledgements	7
12.	References	7
12.1.	Normative References	7
12.2.	Informative References	8
	Authors' Addresses	8

[1.](#) Introduction

IPv6 hosts typically configure a number of IPv6 addresses, which may differ in multiple aspects, such as address scope and stability (e.g. stable addresses vs. temporary addresses). For example, a host may configure one stable and one temporary address per each autoconfiguration prefix advertised on the local network. The addresses to be configured typically depend on local system configuration, with the aforementioned configuration being static and irrespective of the network the host attaches to.

There are three parameters that affect the security and privacy properties of an address:

- o Scope
- o Stability
- o Usage type (client-like "outgoing connections" vs. server-like "incoming connections")

[Section 3](#), [Section 4](#), [Section 5](#), and [Section 6](#) discuss the security and privacy implications (and associated tradeoffs) of the scope, stability and usage type properties of IPv6 addresses, respectively.

[2.](#) Terminology

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](#)].

[3.](#) Predictability Considerations

Predictable IPv6 addresses result in a number of security and privacy implications. For example, [[Barnes2012](#)] discusses how patterns in network prefixes can be leveraged for IPv6 address scanning. On the other hand, [[RFC7707](#)], [[RFC7721](#)] and [[RFC7217](#)] discuss the security and privacy implications of predictable IPv6 Interface Identifiers (IIDs).

Given the aforementioned previous work in this area, and the formal specification update produced by [[RFC8064](#)], we expect (and assume in the rest of this document) that implementations have replaced any schemes that produce predictable addresses with alternative schemes that avoid such patterns (e.g., [RFC7217](#) in replacement of the traditional SLAAC addresses that embed link-layer addresses).

[4.](#) Address Scope Considerations

The IPv6 address scope can, in some scenarios, limit the attack exposure of a node as a result of the implicit isolation provided by a non-global address scope. For example, a node that only employs link-local addresses may, in principle, only be exposed to attack from other nodes in the local link. Hosts employing only Unique Local Addresses (ULAs) may be more isolated from attack than those employing Global Unicast Addresses (GUAs), assuming that proper packet filtering is enforced at the network edge.

The potential protection provided by a non-global addresses should not be regarded as a complete security strategy, but rather as a form of "prophylactic" security (see

[\[I-D.gont-opsawg-firewalls-analysis\]](#))).

We note that the use of non-global addresses is usually limited to a reduced type of applications/protocols that e.g. are only meant to operate on a reduced scope, and hence their applicability may be limited.

A discussion of ULA usage considerations can be found in [\[I-D.ietf-v6ops-ula-usage-considerations\]](#).

[5.](#) Address Stability Considerations

The stability of an address has two associated security/privacy implications:

- o Ability of an attacker to correlate network activity
- o Exposure to attack

For obvious reasons, an address that is employed for multiple communication instances allows the aforementioned network activities to be correlated. The longer an address is employed (i.e., the more stable it is), the longer such correlation will be possible. In the worst-case scenario, a stable address that is employed for multiple communication instances over time will allow all such activities to be correlated. On the other hand, if a host were to generate (and eventually "throw away") one new address for each communication instance (e.g., TCP connection), network activity correlation would be mitigated.

NOTE: the use of constant IIDs (as in traditional SLAAC) result in addresses that, while not constant as a whole (since the prefix changes), contain a globally-unique value that leaks out the node "identity". Such addresses result in the worst possible security and privacy implications, and their use has been deprecated by [\[RFC8064\]](#).

Typically, when it comes to attack exposure, the longer an address is employed the longer an attacker is exposed to attacks (e.g. an

attacker has more time to find the address in the first place [[RFC7707](#)]). While such exposure is traditionally associated with the stability of the address, the usage type of the address (see [Section 6](#)) may also have an impact on attack exposure.

A popular approach to mitigate network activity correlation is the use of "temporary addresses" [[RFC4941](#)]. Temporary addresses are typically configured and employed along with stable addresses, with the temporary addresses employed for outgoing communications, and the stable addresses employed for incoming communications.

NOTE: Ongoing work [[I-D.gont-6man-non-stable-iids](#)] aims at updating [[RFC4941](#)] such that temporary addresses can be employed without the need to configure stable addresses.

We note that the extent to which temporary addresses provide improved mitigation of network activity correlation and/or reduced attack exposure may be questionable and/or limited in some scenarios. For example, a temporary address that is reachable for, say, a few hours

has a questionable "reduced exposure" (particularly when automated attack tools do not typically require such a long period of time to complete their task). Similarly, if network activity can be correlated for the life of such address (e.g., on the order of several hours), such period of time might be long enough for the attacker to correlate all the network activity he is meaning to correlate.

In order to better mitigate network activity correlation and/or possibly reduce host exposure, an implementation might want to either reduce the preferred lifetime of a temporary address, or even better, generate one new temporary address for each new transport protocol instance. However, the associated lifetime/stability of an address may have a negative impact on the network. For example, if a node were to employ "throw away" IPv6 addresses, or employ temporary addresses [[RFC4941](#)] with a short preferred lifetime, local nodes might need to maintain too many entries in their Neighbor Cache, and a number of devices (possibly enforcing security policies) might also need to keep such additional state.

Additionally, enforcing a maximum lifetime on IPv6 addresses may cause long-lived TCP connections to fail. For example, an address

becoming "Invalid" (after transiting through the "Preferred" and "Deprecated" states) would cause the TCP connections employing them to break. This, in turn, would cause e.g. long-lived SSH sessions to break/fail.

In some scenarios, attack exposure may be reduced by limiting the usage of temporary addresses to outbound connections, and prevent such addresses from being used for inbound connections (please see [Section 6](#)).

[6](#). Usage Type Considerations

A node that employs one of its addresses to communicate with an external server (i.e., to perform an "outgoing connection") may cause such address to become exposed to attack. For example, once the external server receives an incoming connection, the corresponding server might launch an attack against the aforementioned address. A real-world instance of this type of scenario has been documented in [\[Hein\]](#).

However, we note that employing an IPv6 address for an outgoing communications need not increase the exposure of local services to other parties. For example, nodes could employ temporary addresses only for outgoing connections, but not for incoming connections. Thus, external nodes that learn about client's addresses could not really leverage such addresses for actively contacting the clients.

There are multiple ways in which this could possibly be achieved, with different implications. Namely:

- Run a host-based or network-based firewall

- Bind services to specific (explicit) addresses

- Bind services only to stable addresses

A client could simply run a host-based firewall that only allows incoming connections on the stable addresses. This is clearly more of an operational way of achieving the desired functionality, and may require good firewall/host integration (e.g., the firewall should be able to tell stable vs. temporary addresses), may require the client to run additional firewall software for this specific purpose, etc.

In other scenarios, a network-based firewalls could be configured to allow outgoing communications from all internal addresses, but only allow incoming communications to stable addresses. For obvious reasons, this is generally only applicable to networks where incoming communications are allowed to a limited number of hosts/servers.

Services could be bound to specific (explicit) addresses, rather than to all locally-configured addresses. However, there are a number of short-comings associated with this approach. Firstly, an application would need to be able to learn all of its addresses and associated stability properties, something that tends to be non-trivial and non-portable, and that also makes applications protocol-dependent, unnecessarily. Secondly, the Sockets API does not really allow a socket to be bound to a subset of the node's addresses. That is, sockets can be bound to a single address or to all available addresses (wildcard), but not to a subset of all the configured addresses.

Binding services only to stable addresses provides a clean separation between addresses employed for client-like outgoing connections and server-like incoming connections. However, we currently lack an appropriate API for nodes to be able to specify that a socket should only be bound to stable addresses. Development of such an API should be considered for future work.

[7.](#) Advice on IPv6 Address Configuration

[TBD]

[8.](#) Advice on IPv6 Address Usage

[TBD]

[9.](#) 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.

10. Security Considerations

This document discusses address usage considerations, and also describes possible future standards-track work to allow for greater flexibility in IPv6 address usage.

11. Acknowledgements

The authors would like to thank [TBD] for providing valuable comments on earlier versions of this document.

Fernando Gont would like to thank Nelida Garcia and Jorge Oscar Gont for their love and support, and Ivan Arce and Diego Armando Maradona for their inspiration.

12. References

12.1. Normative References

- [RFC2460] Deering, S. and R. Hinden, "Internet Protocol, Version 6 (IPv6) Specification", [RFC 2460](#), DOI 10.17487/RFC2460, December 1998, <<http://www.rfc-editor.org/info/rfc2460>>.
- [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>>.
- [RFC4941] Narten, T., Draves, R., and S. Krishnan, "Privacy Extensions for Stateless Address Autoconfiguration in IPv6", [RFC 4941](#), DOI 10.17487/RFC4941, September 2007, <<http://www.rfc-editor.org/info/rfc4941>>.
- [RFC7217] Gont, F., "A Method for Generating Semantically Opaque Interface Identifiers with IPv6 Stateless Address Autoconfiguration (SLAAC)", [RFC 7217](#), DOI 10.17487/RFC7217, April 2014, <<http://www.rfc-editor.org/info/rfc7217>>.

"Recommendation on Stable IPv6 Interface Identifiers",
February 2017, <<https://tools.ietf.org/rfc/rfc8064.txt>>.

12.2. Informative References

- [RFC7707] Gont, F. and T. Chown, "Network Reconnaissance in IPv6 Networks", [RFC 7707](#), DOI 10.17487/RFC7707, March 2016, <<http://www.rfc-editor.org/info/rfc7707>>.
- [RFC7721] Cooper, A., Gont, F., and D. Thaler, "Security and Privacy Considerations for IPv6 Address Generation Mechanisms", [RFC 7721](#), DOI 10.17487/RFC7721, March 2016, <<http://www.rfc-editor.org/info/rfc7721>>.
- [I-D.ietf-v6ops-ula-usage-considerations]
Liu, B. and S. Jiang, "Considerations For Using Unique Local Addresses", [draft-ietf-v6ops-ula-usage-considerations-01](#) (work in progress), August 2016.
- [I-D.gont-6man-non-stable-iids]
Gont, F. and S. LIU, "Recommendation on Non-Stable IPv6 Interface Identifiers", [draft-gont-6man-non-stable-iids-00](#) (work in progress), May 2016.
- [I-D.gont-opsawg-firewalls-analysis]
Gont, F. and F. Baker, "On Firewalls in Network Security", [draft-gont-opsawg-firewalls-analysis-02](#) (work in progress), February 2016.
- [Barnes2012]
Barnes, R., Altmann, R., and D. Kerr, "Mapping the Great Void Smarter scanning for IPv6", ISMA 2012 AIMS-4 - Workshop on Active Internet Measurements, February 2012, <https://www.caida.org/workshops/isma/1202/slides/aims1202_rbarnes.pdf>.
- [Hein] Hein, B., "The Rising Sophistication of Network Scanning", January 2016, <<http://netpatterns.blogspot.be/2016/01/the-rising-sophistication-of-network.html>>.

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
URI: <http://www.si6networks.com>

Guillermo Gont
SI6 Networks
Evaristo Carriego 2644
Haedo, Provincia de Buenos Aires 1706
Argentina

Phone: +54 11 4650 8472
Email: ggont@si6networks.com
URI: <https://www.si6networks.com>

Madeleine Garcia Corbo
Servicios de Informacion del Transporte
Neptuno 358
Havana City 10400
Cuba

Email: madelen.garcia16@gmail.com

