Internet Engineering Task Force (IETF)
Request for Comments: 7721
Category: Informational
ISSN: 2070-1721

A. Cooper
Cisco
F. Gont
Huawei Technologies
D. Thaler
Microsoft

March 2016

Security and Privacy Considerations for IPv6 Address Generation Mechanisms

Abstract

This document discusses privacy and security considerations for several IPv6 address generation mechanisms, both standardized and non-standardized. It evaluates how different mechanisms mitigate different threats and the trade-offs that implementors, developers, and users face in choosing different addresses or address generation mechanisms.

Status of This Memo

This document is not an Internet Standards Track specification; it is published for informational purposes.

This document is a product of the Internet Engineering Task Force (IETF). It represents the consensus of the IETF community. It has received public review and has been approved for publication by the Internet Engineering Steering Group (IESG). Not all documents approved by the IESG are a candidate for any level of Internet Standard; see Section 2 of RFC 5741.

Information about the current status of this document, any errata, and how to provide feedback on it may be obtained at http://www.rfc-editor.org/info/rfc7721.

Table of Contents

1. Introduction

IPv6 was designed to improve upon IPv4 in many respects, and mechanisms for address assignment were one such area for improvement. In addition to static address assignment and DHCP, stateless autoconfiguration was developed as a less intensive, fate-shared means of performing address assignment. With stateless autoconfiguration, routers advertise on-link prefixes and hosts generate their own Interface Identifiers (IIDs) to complete their addresses. [RFC7136] clarifies that the IID should be treated as an opaque value, while [RFC7421] provides an analysis of the 64-bit boundary in IPv6 addressing (e.g., the implications of the IID length on security and privacy). Over the years, many IID generation techniques have been defined, both standardized and non-standardized:

- Manual configuration [RFC7707]
  - IPv4 address
  - Service port
  - Wordy
  - Low-byte
- Stateless Address Autoconfiguration (SLAAC)
  - IEEE 802 48-bit Media Access Control (MAC) or IEEE 64-bit Extended Unique Identifier (EUI-64) [RFC2464]
  - Cryptographically generated [RFC3972]
  - Temporary (also known as "privacy addresses") [RFC4941]
  - Constant, semantically opaque (also known as "random") [Microsoft]
  - Stable, semantically opaque [RFC7217]
- DHCPv6 based [RFC3315]
- Specified by transition/co-existence technologies
  - Derived from an IPv4 address (e.g., [RFC5214], [RFC6052])
  - Derived from an IPv4 address and port set ID (e.g., [RFC7596], [RFC7597], [RFC7599])

* Derived from an IPv4 address and port (e.g., [RFC4380])

Deriving the IID from a globally unique IEEE identifier [RFC2464] [RFC4862] was one of the earliest mechanisms developed (and originally specified in [RFC1971] and [RFC1972]). A number of privacy and security issues related to the IIDs derived from IEEE identifiers were discovered after their standardization, and many of the mechanisms developed later aimed to mitigate some or all of these weaknesses. This document identifies four types of attacks against IEEE-identifier-based IIDs and discusses how other existing techniques for generating IIDs do or do not mitigate those attacks.

2. Terminology

This section clarifies the terminology used throughout this document.

Public address:
: An address that has been published in a directory or other public location, such as the DNS, a SIP proxy [RFC3261], an application-specific Distributed Hash Table (DHT), or a publicly available URI. A host's public addresses are intended to be discoverable by third parties.

Stable address:
: An address that does not vary over time within the same IPv6 link. Note that [RFC4941] refers to these as "public" addresses, but "stable" is used here for reasons explained in Section 4.

Temporary address:
: An address that varies over time within the same IPv6 link.

Constant IID:
: An IPv6 interface identifier that is globally stable. That is, the Interface ID will remain constant even if the node moves from one IPv6 link to another.

Stable IID:
: An IPv6 interface identifier that is stable within some specified context. For example, an Interface ID can be globally stable (constant) or could be stable per IPv6 link (meaning that the Interface ID will remain unchanged as long as the node stays on the same IPv6 link but may change when the node moves from one IPv6 link to another).

Temporary IID:
: An IPv6 interface identifier that varies over time.

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "NOT RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in [RFC2119]. These words take their normative meanings only when they are presented in ALL UPPERCASE.

3. Weaknesses in IEEE-Identifier-Based IIDs

There are a number of privacy and security implications that exist for hosts that use IEEE-identifier-based IIDs. This section discusses four generic attack types: correlation of activities over time, location tracking, address scanning, and device-specific vulnerability exploitation. The first three of these rely on the attacker first gaining knowledge of the IID of the target host. This could be achieved by a number of different entities: the operator of a server to which the host connects, such as a web server or a peer-to-peer server; an entity that connects to the same IPv6 link as the target (such as a conference network or any public network); a passive observer of traffic that the host broadcasts; or an entity that is on path to the destinations with which the host communicates, such as a network operator.

3.1. Correlation of Activities over Time

As with other identifiers, an IPv6 address can be used to correlate the activities of a host for at least as long as the lifetime of the address. The correlation made possible by IEEE-identifier-based IIDs is of particular concern since they last roughly for the lifetime of a device's network interface, allowing correlation on the order of years.

As [RFC4941] explains,

> [t]he use of a non-changing interface identifier to form addresses is a specific instance of the more general case where a constant identifier is reused over an extended period of time and in multiple independent activities. Anytime the same identifier is used in multiple contexts, it becomes possible for that identifier to be used to correlate seemingly unrelated activity. ... The use of a constant identifier within an address is of special concern because addresses are a fundamental requirement of communication and cannot easily be hidden from eavesdroppers and other parties. Even when higher layers encrypt their payloads, addresses in packet headers appear in the clear.

IP addresses are just one example of information that can be used to correlate activities over time. DNS names, cookies [RFC6265], browser fingerprints [Panopticlick], and application-layer usernames

can all be used to link a host's activities together. Although IEEE-identifier-based IIDs are likely to last at least as long or longer than these other identifiers, IIDs generated in other ways may have shorter or longer lifetimes than these identifiers depending on how they are generated. Therefore, the extent to which a host's activities can be correlated depends on whether the host uses multiple identifiers together and the lifetimes of all of those identifiers. Frequently refreshing an IPv6 address may not mitigate correlation if an attacker has access to other longer-lived identifiers for a particular host. This is an important caveat to keep in mind throughout the discussion of correlation in this document. For further discussion of correlation, see Section 5.2.1 of [RFC6973].

As noted in [RFC4941], in some cases correlation is just as feasible for a host using an IPv4 address as for a host using an IEEE identifier to generate its IID in its IPv6 address. Hosts that use static IPv4 addressing or who are consistently allocated the same address via DHCPv4 can be tracked as described above. However, the widespread use of both NAT and DHCPv4 implementations that assign the same host a different address upon lease expiration mitigates this threat in the IPv4 case as compared to the IEEE identifier case in IPv6.

3.2. Location Tracking

Because the IPv6 address structure is divided between a topological portion and an interface identifier portion, an interface identifier that remains constant when a host connects to different IPv6 links (as an IEEE-identifier-based IID does) provides a way for observers to track the movements of that host. In a passive attack on a mobile host, a server that receives connections from the same host over time would be able to determine the host's movements as its prefix changes.

Active attacks are also possible. An attacker that first learns the host's interface identifier by being connected to the same IPv6 link, running a server that the host connects to, or being on path to the host's communications could subsequently probe other networks for the presence of the same interface identifier by sending a probe packet (e.g., ICMPv6 Echo Request, or any other probe packet). Even if the host does not respond, the first-hop router will usually respond with an ICMP Destination Unreachable/Address Unreachable (type 1, code 3) when the host is not present and be silent when the host is present.

Location tracking based on IP address is generally not possible in IPv4 since hosts get assigned wholly new addresses when they change networks.

3.3. Address Scanning

The structure of IEEE-based identifiers used for address generation can be leveraged by an attacker to reduce the target search space [RFC7707]. The 24-bit Organizationally Unique Identifier (OUI) of MAC addresses, together with the fixed value (0xff, 0xfe) used to form a Modified EUI-64 interface identifier, greatly help to reduce the search space, making it easier for an attacker to scan for individual addresses using widely known popular OUIs. This erases much of the protection against address scanning that the larger IPv6 address space could provide as compared to IPv4.

3.4. Device-Specific Vulnerability Exploitation

IPv6 addresses that embed IEEE identifiers leak information about the device (e.g., Network Interface Card vendor, or even Operating System and/or software type), which could be leveraged by an attacker with knowledge of device- or software-specific vulnerabilities to quickly find possible targets. Attackers can exploit vulnerabilities in hosts whose IIDs they have previously obtained or scan an address space to find potential targets.

4. Privacy and Security Properties of Address Generation Mechanisms

Analysis of the extent to which a particular host is protected against the attacks described in Section 3 depends on how each of a host's addresses is generated and used. In some scenarios, a host configures a single global address and uses it for all communications. In other scenarios, a host configures multiple addresses using different mechanisms and may use any or all of them.

[RFC3041] (later obsoleted by [RFC4941]) sought to address some of the problems described in Section 3 by defining "temporary addresses" for outbound connections. Temporary addresses are meant to supplement the other addresses that a device might use, not to replace them. They use IIDs that are randomly generated and change daily by default. The idea was for temporary addresses to be used for outgoing connections (e.g., web browsing) while maintaining the ability to use a stable address when more address stability is desired (e.g., for IPv6 addresses published in the DNS).

[RFC3484] originally specified that stable addresses be used for outbound connections unless an application explicitly prefers temporary addresses. The default preference for stable addresses was established to avoid applications potentially failing due to the short lifetime of temporary addresses or the possibility of a reverse look-up failure or error. However, [RFC3484] allowed that "implementations for which privacy considerations outweigh these

application-compatibility concerns MAY reverse the sense of this rule" and instead prefer by default temporary addresses rather than stable addresses. Indeed, most implementations (notably including Windows) chose to default to temporary addresses for outbound connections since privacy was considered more important (and few applications supported IPv6 at the time, so application compatibility concerns were minimal). [RFC6724] then obsoleted [RFC3484] and changed the default to match what implementations actually did.

The envisioned relationship in [RFC3484] between stability of an address and its use in "public" can be misleading when conducting privacy analysis. The stability of an address and the extent to which it is linkable to some other public identifier are independent of one another. For example, there is nothing that prevents a host from publishing a temporary address in a public place, such as the DNS. Publishing both a stable address and a temporary address in the DNS or elsewhere where they can be linked together by a public identifier allows the host's activities when using either address to be correlated together.

Moreover, because temporary addresses were designed to supplement other addresses generated by a host, the host may still configure a more stable address even if it only ever intentionally uses temporary addresses (as source addresses) for communication to off-link destinations. An attacker can probe for the stable address even if it is never used as such a source address or advertised outside the link (e.g., in DNS or SIP).

This section compares the privacy and security properties of a variety of IID generation mechanisms and their possible usage scenarios, including scenarios in which a single mechanism is used to generate all of a host's IIDs and those in which temporary addresses are used together with addresses generated using a different IID generation mechanism. The analysis of the exposure of each IID type to correlation assumes that IPv6 prefixes are shared by a reasonably large number of nodes. As [RFC4941] notes, if a very small number of nodes (say, only one) use a particular prefix for an extended period of time, the prefix itself can be used to correlate the host's activities regardless of how the IID is generated. For example, [RFC3314] recommends that prefixes be uniquely assigned to mobile handsets where IPv6 is used within General Packet Radio Service (GPRS). In cases where this advice is followed and prefixes persist for extended periods of time (or get reassigned to the same handsets whenever those handsets reconnect to the same network router), hosts' activities could be correlatable for longer periods than the analysis below would suggest.

The table below provides a summary of the whole analysis. A "No" entry indicates that the attack is prevented from being carried out on the basis of the IID, but the host may still be vulnerable depending on how it employs other protocols.

Mechanism(s)	Correlation	Location tracking	Address scanning	Device exploits
IEEE identifier	For device lifetime	For device lifetime	Possible	Possible
Static manual	For address lifetime	For address lifetime	Depends on generation mechanism	Depends on generation mechanism
Constant, semantically opaque	For address lifetime	For address lifetime	No	No
CGA	For lifetime of (modifier block + public key)	No	No	No
Stable, semantically opaque	Within single IPv6 link	No	No	No
Temporary	For temp address lifetime	No	No	No
DHCPv6	For lease lifetime	No	Depends on generation mechanism	No

Table 1: Privacy and Security Properties of IID Generation Mechanisms

4.1. IEEE-Identifier-Based IIDs

As discussed in Section 3, addresses that use IIDs based on IEEE identifiers are vulnerable to all four attacks. They allow correlation and location tracking for the lifetime of the device since IEEE identifiers last that long and their structure makes address scanning and device exploits possible.

4.2. Static, Manually Configured IIDs

Because static, manually configured IIDs are stable, both correlation and location tracking are possible for the life of the address.

The extent to which location tracking can be successfully performed depends, to some extent, on the uniqueness of the employed IID. For example, one would expect "low byte" IIDs to be more widely reused than, for example, IIDs where the whole 64 bits follow some pattern that is unique to a specific organization. Widely reused IIDs will typically lead to false positives when performing location tracking.

Whether manually configured addresses are vulnerable to address scanning and device exploits depends on the specifics of how the IIDs are generated.

4.3. Constant, Semantically Opaque IIDs

Although a mechanism to generate a constant, semantically opaque IID has not been standardized, it has been in wide use for many years on at least one platform (Windows). Windows uses the random generation mechanism described in [RFC4941] in lieu of generating an IEEE-identifier-based IID. This mitigates the device-specific exploitation and address-scanning attacks but still allows correlation and location tracking because the IID is constant across IPv6 links and time.

4.4. Cryptographically Generated IIDs

Cryptographically Generated Addresses (CGAs) [RFC3972] bind a hash of the host's public key to an IPv6 address in the SEcure Neighbor Discovery (SEND) protocol [RFC3971]. CGAs may be regenerated for each subnet prefix, but this is not required given that they are computationally expensive to generate. A host using a CGA can be correlated for as long as the lifetime of the combination of the public key and the chosen modifier block since it is possible to rotate modifier blocks without generating new public keys. Because the cryptographic hash of the host's public key uses the subnet prefix as an input, even if the host does not generate a new public key or modifier block when it moves to a different IPv6 link, its

location cannot be tracked via the IID. CGAs do not allow device-specific exploitation or address-scanning attacks.

4.5. Stable, Semantically Opaque IIDs

[RFC7217] specifies an algorithm that generates, for each network interface, a unique random IID per IPv6 link. The aforementioned algorithm is employed not only for global unicast addresses, but also for unique local unicast addresses and link-local unicast addresses since these addresses may leak out via application protocols (e.g., IPv6 addresses embedded in email headers).

A host that stays connected to the same IPv6 link could therefore be tracked at length, whereas a mobile host's activities could only be correlated for the duration of each network connection. Location tracking is not possible with these addresses. They also do not allow device-specific exploitation or address-scanning attacks.

4.6. Temporary IIDs

A host that uses only a temporary address mitigates all four threats. Its activities may only be correlated for the lifetime of a single temporary address.

A host that configures both an IEEE-identifier-based IID and temporary addresses makes the host vulnerable to the same attacks as if temporary addresses were not in use, although the viability of some of them depends on how the host uses each address. An attacker can correlate all of the host's activities for which it uses its IEEE-identifier-based IID. Once an attacker has obtained the IEEE-identifier-based IID, location tracking becomes possible on other IPv6 links even if the host only makes use of temporary addresses on those other IPv6 links; the attacker can actively probe the other IPv6 links for the presence of the IEEE-identifier-based IID. Device-specific vulnerabilities can still be exploited. Address scanning is also still possible because the IEEE-identifier-based address can be probed.

If the host instead generates a constant, semantically opaque IID to use in a stable address for server-like connections together with temporary addresses for outbound connections (as is the default in Windows), it sees some improvements over the previous scenario. The address-scanning attacks and device-specific exploitation attacks are no longer possible because the OUI is no longer embedded in any of the host's addresses. However, correlation of some activities across time and location tracking are both still possible because the semantically opaque IID is constant. And once an attacker has obtained the host's semantically opaque IID, location tracking is

possible on any network by probing for that IID, even if the host only uses temporary addresses on those networks. However, if the host generates but never uses a constant, semantically opaque IID, it mitigates all four threats.

When used together with temporary addresses, the stable, semantically opaque IID generation mechanism [RFC7217] improves upon the previous scenario by limiting the potential for correlation to the lifetime of the stable address (which may still be lengthy for hosts that are not mobile) and by eliminating the possibility for location tracking (since a different IID is generated for each subnet prefix). As in the previous scenario, a host that configures but does not use a stable, semantically opaque address mitigates all four threats.

4.7. DHCPv6 Generation of IIDs

The security and privacy implications of DHCPv6-based addresses will typically depend on whether the client requests an IA_NA (Identity Association for Non-temporary Addresses) or an IA_TA (Identity Association for Temporary Addresses) [RFC3315] and the specific DHCPv6 server software being employed.

DHCPv6 temporary addresses have the same properties as SLAAC temporary addresses (see Section 4.6). On the other hand, the properties of DHCPv6 non-temporary addresses typically depend on the specific DHCPv6 server software being employed. Recent releases of most popular DHCPv6 server software typically lease random addresses with a similar lease time as that of IPv4. Thus, these addresses can be considered to be "stable, semantically opaque". [DHCPv6-IID] specifies an algorithm that can be employed by DHCPv6 servers to generate "stable, semantically opaque" addresses.

On the other hand, some DHCPv6 software leases sequential addresses (typically low-byte addresses). These addresses can be considered to be stable addresses. The drawback of this address generation scheme compared to "stable, semantically opaque" addresses is that, since they follow specific patterns, they enable IPv6 address scans.

4.8. Transition and Coexistence Technologies

Addresses specified based on transition or coexistence technologies that embed an IPv4 address within an IPv6 address are not included in Table 1 because their privacy and security properties are inherited from the embedded address. For example, Teredo [RFC4380] specifies a means to generate an IPv6 address from the underlying IPv4 address and port, leaving many other bits set to zero. This makes it relatively easy for an attacker to scan for IPv6 addresses by guessing the Teredo client's IPv4 address and port (which for many

NATs is not randomized). For this reason, popular implementations (e.g., Windows) began deviating from the standard by including 12 random bits in place of zero bits. This modification was later standardized in [RFC5991].

Some other transition technologies (e.g., [RFC5214], [RFC6052]) specify means to generate an IPv6 address from an underlying IPv4 address without a port. Such mechanisms thus make it much easier for an attacker to conduct an address scan than for mechanisms that require finding a port number as well.

Finally, still other mechanisms (e.g., [RFC7596], [RFC7597], [RFC7599]) are somewhere in between, using an IPv4 address and a port set ID (which for many NATs is not randomized). In general, such mechanisms are thus typically as easy to scan as in the Teredo example above without the 12-bit mitigation.

5. Miscellaneous Issues with IPv6 Addressing

5.1. Network Operation

It is generally agreed that IPv6 addresses that vary over time in a specific IPv6 link tend to increase the complexity of event logging, trouble-shooting, enforcement of access controls and quality of service, etc. As a result, some organizations disable the use of temporary addresses [RFC4941] even at the expense of reduced privacy [Broersma].

5.2. Compliance

Some IPv6 compliance testing suites required (and might still require) implementations to support IEEE-identifier-based IIDs in order to be approved as compliant. This document recommends that compliance testing suites be relaxed to allow other forms of address generation that are more amenable to privacy.

5.3. Intellectual Property Rights (IPRs)

Some IPv6 addressing techniques might be covered by Intellectual Property rights, which might limit their implementation in different operating systems. [CGA-IPR] and [KAME-CGA] discuss the IPRs on CGAs.

6. Security Considerations

This whole document concerns the privacy and security properties of different IPv6 address generation mechanisms.

7. References

7.1. 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>.

[RFC2464] Crawford, M., "Transmission of IPv6 Packets over Ethernet Networks", RFC 2464, DOI 10.17487/RFC2464, December 1998, <http://www.rfc-editor.org/info/rfc2464>.

[RFC3315] Droms, R., Ed., Bound, J., Volz, B., Lemon, T., Perkins, C., and M. Carney, "Dynamic Host Configuration Protocol for IPv6 (DHCPv6)", RFC 3315, DOI 10.17487/RFC3315, July 2003, <http://www.rfc-editor.org/info/rfc3315>.

[RFC3971] Arkko, J., Ed., Kempf, J., Zill, B., and P. Nikander, "SEcure Neighbor Discovery (SEND)", RFC 3971, DOI 10.17487/RFC3971, March 2005, <http://www.rfc-editor.org/info/rfc3971>.

[RFC3972] Aura, T., "Cryptographically Generated Addresses (CGA)", RFC 3972, DOI 10.17487/RFC3972, March 2005, <http://www.rfc-editor.org/info/rfc3972>.

[RFC4380] Huitema, C., "Teredo: Tunneling IPv6 over UDP through Network Address Translations (NATs)", RFC 4380, DOI 10.17487/RFC4380, February 2006, <http://www.rfc-editor.org/info/rfc4380>.

[RFC4862] Thomson, S., Narten, T., and T. Jinmei, "IPv6 Stateless Address Autoconfiguration", RFC 4862, DOI 10.17487/RFC4862, September 2007, <http://www.rfc-editor.org/info/rfc4862>.

[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>.

[RFC5991] Thaler, D., Krishnan, S., and J. Hoagland, "Teredo Security Updates", RFC 5991, DOI 10.17487/RFC5991, September 2010, <http://www.rfc-editor.org/info/rfc5991>.

[RFC6724] Thaler, D., Ed., Draves, R., Matsumoto, A., and T. Chown, "Default Address Selection for Internet Protocol Version 6 (IPv6)", RFC 6724, DOI 10.17487/RFC6724, September 2012, <http://www.rfc-editor.org/info/rfc6724>.

[RFC7136] Carpenter, B. and S. Jiang, "Significance of IPv6 Interface Identifiers", RFC 7136, DOI 10.17487/RFC7136, February 2014, <http://www.rfc-editor.org/info/rfc7136>.

[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>.

7.2. Informative References

[Broersma] Broersma, R., "IPv6 Everywhere: Living with a Fully IPv6-enabled environment", Australian IPv6 Summit 2010, Melbourne, VIC Australia, October 2010, <http://www.ipv6.org.au/10ipv6summit/talks/Ron_Broersma.pdf>.

[CGA-IPR] IETF, "IPR Details: Microsoft's Statement about IPR claimed in RFC 3972", November 2005, <https://datatracker.ietf.org/ipr/676/>.

[DHCPv6-IID] Gont, F. and W. Liu, "A Method for Generating Semantically Opaque Interface Identifiers with Dynamic Host Configuration Protocol for IPv6 (DHCPv6)", Work in Progress, draft-ietf-dhc-stable-privacy-addresses-02, April 2015.

[KAME-CGA] The KAME Project, "The KAME IPR policy and concerns of some technologies which have IPR claims", November 2005, <http://www.kame.net/newsletter/20040525/>.

[Microsoft] Microsoft, "IPv6 interface identifiers", 2013, <http://www.microsoft.com/resources/documentation/windows/xp/all/proddocs/en-us/sag_ip_v6_imp_addr7.mspx?mfr=true>.

[Panopticlick] Electronic Frontier Foundation, "Panopticlick", 2011, <http://panopticlick.eff.org>.

[RFC1971] Thomson, S. and T. Narten, "IPv6 Stateless Address Autoconfiguration", RFC 1971, DOI 10.17487/RFC1971, August 1996, <http://www.rfc-editor.org/info/rfc1971>.

[RFC1972] Crawford, M., "A Method for the Transmission of IPv6 Packets over Ethernet Networks", RFC 1972, DOI 10.17487/RFC1972, August 1996, <http://www.rfc-editor.org/info/rfc1972>.

[RFC3041] Narten, T. and R. Draves, "Privacy Extensions for Stateless Address Autoconfiguration in IPv6", RFC 3041, DOI 10.17487/RFC3041, January 2001, <http://www.rfc-editor.org/info/rfc3041>.

[RFC3261] Rosenberg, J., Schulzrinne, H., Camarillo, G., Johnston, A., Peterson, J., Sparks, R., Handley, M., and E. Schooler, "SIP: Session Initiation Protocol", RFC 3261, DOI 10.17487/RFC3261, June 2002, <http://www.rfc-editor.org/info/rfc3261>.

[RFC3314] Wasserman, M., Ed., "Recommendations for IPv6 in Third Generation Partnership Project (3GPP) Standards", RFC 3314, DOI 10.17487/RFC3314, September 2002, <http://www.rfc-editor.org/info/rfc3314>.

[RFC3484] Draves, R., "Default Address Selection for Internet Protocol version 6 (IPv6)", RFC 3484, DOI 10.17487/RFC3484, February 2003, <http://www.rfc-editor.org/info/rfc3484>.

[RFC5214] Templin, F., Gleeson, T., and D. Thaler, "Intra-Site Automatic Tunnel Addressing Protocol (ISATAP)", RFC 5214, DOI 10.17487/RFC5214, March 2008, <http://www.rfc-editor.org/info/rfc5214>.

[RFC6052] Bao, C., Huitema, C., Bagnulo, M., Boucadair, M., and X. Li, "IPv6 Addressing of IPv4/IPv6 Translators", RFC 6052, DOI 10.17487/RFC6052, October 2010, <http://www.rfc-editor.org/info/rfc6052>.

[RFC6265] Barth, A., "HTTP State Management Mechanism", RFC 6265, DOI 10.17487/RFC6265, April 2011, <http://www.rfc-editor.org/info/rfc6265>.

[RFC6973] Cooper, A., Tschofenig, H., Aboba, B., Peterson, J., Morris, J., Hansen, M., and R. Smith, "Privacy Considerations for Internet Protocols", RFC 6973, DOI 10.17487/RFC6973, July 2013, <http://www.rfc-editor.org/info/rfc6973>.

[RFC7421] Carpenter, B., Ed., Chown, T., Gont, F., Jiang, S., Petrescu, A., and A. Yourtchenko, "Analysis of the 64-bit Boundary in IPv6 Addressing", RFC 7421, DOI 10.17487/RFC7421, January 2015, <http://www.rfc-editor.org/info/rfc7421>.

[RFC7596] Cui, Y., Sun, Q., Boucadair, M., Tsou, T., Lee, Y., and I. Farrer, "Lightweight 4over6: An Extension to the Dual-Stack Lite Architecture", RFC 7596, DOI 10.17487/RFC7596, July 2015, <http://www.rfc-editor.org/info/rfc7596>.

[RFC7597] Troan, O., Ed., Dec, W., Li, X., Bao, C., Matsushima, S., Murakami, T., and T. Taylor, Ed., "Mapping of Address and Port with Encapsulation (MAP-E)", RFC 7597, DOI 10.17487/RFC7597, July 2015, <http://www.rfc-editor.org/info/rfc7597>.

[RFC7599] Li, X., Bao, C., Dec, W., Ed., Troan, O., Matsushima, S., and T. Murakami, "Mapping of Address and Port using Translation (MAP-T)", RFC 7599, DOI 10.17487/RFC7599, July 2015, <http://www.rfc-editor.org/info/rfc7599>.

[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>.

Acknowledgements

The authors would like to thank Bernard Aboba, Brian Carpenter, Tim Chown, Lorenzo Colitti, Rich Draves, Robert Hinden, Robert Moskowitz, Erik Nordmark, Mark Smith, Ole Troan, and James Woodyatt for providing valuable comments on earlier draft versions of this document.

Authors' Addresses

Alissa Cooper
Cisco
707 Tasman Drive
Milpitas, CA 95035
United States

Phone: +1-408-902-3950
Email: alcoop@cisco.com
URI: https://www.cisco.com/

Fernando Gont
Huawei Technologies
Evaristo Carriego 2644
Haedo, Provincia de Buenos Aires 1706
Argentina

Phone: +54 11 4650 8472
Email: fgont@si6networks.com
URI: http://www.si6networks.com

Dave Thaler
Microsoft
One Microsoft Way
Redmond, WA 98052
United States

Phone: +1 425 703 8835
Email: dthaler@microsoft.com