Internet Engineering Task Force (IETF)
Request for Comments: 6619
Category: Standards Track
ISSN: 2070-1721

J. Arkko
Ericsson
L. Eggert
NetApp
M. Townsley
Cisco

June 2012

Scalable Operation of Address Translators with Per-Interface Bindings

Abstract

This document explains how to employ address translation in networks that serve a large number of individual customers without requiring a correspondingly large amount of private IPv4 address space.

Status of This Memo

This is an Internet Standards Track document.

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). Further information on Internet Standards is available in 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/rfc6619.

1. Introduction

This document explains how to employ address translation without consuming a large amount of private address space. This is important in networks that serve a large number of individual customers. Networks that serve more than 2^24 (16 million) users cannot assign a unique private IPv4 address to each user, because the largest reserved private address block reserved is 10/8 [RFC1918]. Many networks are already hitting these limits today -- for instance, in the consumer Internet service market. Even some individual devices may approach these limits -- for instance, cellular network gateways or mobile IP home agents.

If ample IPv4 address space were available, this would be a non-issue, because the current practice of assigning public IPv4 addresses to each user would remain viable, and the complications associated with using the more limited private address space could be avoided. However, as the IPv4 address pool is becoming depleted, this practice is becoming increasingly difficult to sustain.

It has been suggested that more of the unassigned IPv4 space should be converted for private use, in order to allow the provisioning of larger networks with private IPv4 address space. At the time of this writing, the IANA "free pool" contained only 12 unallocated unicast IPv4 /8 prefixes. Although reserving a few of those for private use would create some breathing room for such deployments, it would not result in a solution with long-term viability. It would result in significant operational and management overheads, and it would further reduce the number of available IPv4 addresses.

Segmenting a network into areas of overlapping private address space is another possible technique, but it severely complicates the design and operation of a network.

Finally, the transition to IPv6 will eventually eliminate these addressing limitations. However, during the migration period when IPv4 and IPv6 have to coexist, address or protocol translation will be needed in order to reach IPv4 destinations.

The rest of this document is organized as follows. Section 2 gives an outline of the solution, Section 3 introduces some terms, Section 4 specifies the required behavior for managing NAT bindings, and Section 5 discusses the use of this technique with IPv6.

2. Solution Outline

The need for address or protocol translation during the migration period to IPv6 creates the opportunity to deploy these mechanisms in a way that allows the support of a large user base without the need for a correspondingly large IPv4 address block.

A Network Address Translator (NAT) is typically configured to connect a network domain that uses private IPv4 addresses to the public Internet. The NAT device, which is configured with a public IPv4 address, creates and maintains a mapping for each communication session from a device inside the domain it serves to devices in the public Internet. It does that by translating the packet flow of each session such that the externally visible traffic uses only public addresses.

In many NAT deployments, the network domain connected by the NAT to the public Internet is a broadcast network sharing the same media, where each individual device must have a private IPv4 address that is unique within this network. In such deployments, it is natural also to implement the NAT functionality such that it uses the private IPv4 address when looking up which mapping should be used to translate a given communication session.

It is important to note, however, that this is not an inherent requirement. When other methods of identifying the correct mapping are available, and the NAT is not connecting a shared-media broadcast network to the Internet, there is no need to assign each device in the domain a unique IPv4 address.

This is the case, for example, when the NAT connects devices to the Internet that connect to it with individual point-to-point links. In this case, it becomes possible to use the same private addresses many times, making it possible to support any number of devices behind a NAT using very few IPv4 addresses.

There are tunneling-based techniques that can obtain the same benefits by establishing new tunnels over any IP network [RFC6333]. However, where the point-to-point links already exist, creating an additional layer of tunneling is unnecessary (and even potentially harmful due to effects on the Maximum Transfer Unit (MTU) settings). The approach described in this document can be implemented and deployed within a single device and has no effect on hosts behind it. In addition, as no additional layers of tunneling are introduced, there is no effect on the MTU. It is also unnecessary to implement tunnel endpoint discovery, security mechanisms, or other aspects of a tunneling solution. In fact, there are no changes to the devices behind the NAT.

Note, however, that existing tunnels are a common special case of point-to-point links. For instance, cellular network gateways terminate a large number of tunnels that are already needed for mobility management reasons. Implementing the approach described in this document is particularly attractive in such environments, given that no additional tunneling mechanisms, negotiation, or host changes are required. In addition, since there is no additional tunneling, packets continue to take the same path as they would normally take. Other commonly used network technologies that may be of interest include Point-to-Point Protocol (PPP) [RFC1661] links, PPP over Ethernet (PPPoE) [RFC2516] encapsulation, Asynchronous Transfer Mode (ATM) Permanent Virtual Circuits (PVCs), and per-subscriber virtual LAN (VLAN) allocation in consumer broadband networks.

The approach described here also results in overlapping private address space, like the segmentation of the network to different areas. However, this overlap is applied only at the network edges and does not impact routing or reachability of servers in a negative way.

3. Terms

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 [RFC2119].

"NAT" in this document includes both "Basic NAT" and "Network Address Port Translation (NAPT)" as defined by [RFC2663]. The term "NAT Session" is adapted from [RFC5382] and is defined as follows.

> NAT Session - A NAT session is an association between a transport layer session as seen in the internal realm and a session as seen in the external realm, by virtue of NAT translation. The NAT session will provide the translation glue between the two session representations.

This document uses the term "mapping" as defined in [RFC4787] to refer to state at the NAT necessary for network address and port translation of sessions.

4. Per-Interface Bindings

To support a mode of operation that uses a fixed number of IPv4 addresses to serve an arbitrary number of devices, a NAT MUST manage its mappings on a per-interface basis, by associating a particular NAT session not only with the five tuples used for the transport connection on both sides of the NAT but also with the internal interface on which the user device is connected to the NAT. This

approach allows each internal interface to use the same private IPv4 address range. Note that the interface need not be physical; it may also correspond to a tunnel, VLAN, or other identifiable communications channel.

For deployments where exactly one user device is connected with a separate tunnel interface and all tunnels use the same IPv4 address for the user devices, it is redundant to store this address in the mapping in addition to the internal interface identifier. When the internal interface identifier is shorter than a 32-bit IPv4 address, this may decrease the storage requirements of a mapping entry by a small measure, which may aid NAT scalability. For other deployments, it is likely necessary to store both the user device IPv4 address and the internal interface identifier, which slightly increases the size of the mapping entry.

This mode of operation is only suitable in deployments where user devices connect to the NAT over point-to-point links. If supported, this mode of operation SHOULD be configurable, and it should be disabled by default in general-purpose NAT devices.

All address translators make it hard to address devices behind them. The same is true of the particular NAT variant described in this document. An additional constraint is caused by the use of the same address space for different devices behind the NAT, which prevents the use of unique private addresses for communication between devices behind the same NAT.

5. IPv6 Considerations

Private address space conservation is important even during the migration to IPv6, because it will be necessary to communicate with the IPv4 Internet for a long time. This document specifies two recommended deployment models for IPv6. In the first deployment model, the mechanisms specified in this document are useful. In the second deployment model, no additional mechanisms are needed, because IPv6 addresses are already sufficient to distinguish mappings from each other.

The first deployment model employs dual stack [RFC4213]. The IPv6 side of dual stack operates based on global addresses and direct end-to-end communication. However, on the IPv4 side, private addressing and NATs are a necessity. The use of per-interface NAT mappings is RECOMMENDED for the IPv4 side under these circumstances. Per-interface mappings help the NAT scale, while dual-stack operation helps reduce the pressure on the NAT device by moving key types of traffic to IPv6, eliminating the need for NAT processing.

The second deployment model involves the use of address and protocol translation, such as the one defined in [RFC6146]. In this deployment model, there is no IPv4 in the internal network at all. This model is applicable only in situations where all relevant devices and applications are IPv6 capable. In this situation, per-interface mappings could be employed as specified above, but they are generally unnecessary, as the IPv6 address space is large enough to provide a sufficient number of mappings.

6. Security Considerations

The practices outlined in this document do not affect the security properties of address translation. The binding method specified in this document is not observable to a device that is on the outside of the NAT; i.e., a regular NAT and a NAT specified here cannot be distinguished. However, the use of point-to-point links implies naturally that the devices behind the NAT cannot communicate with each other directly without going through the NAT (or a router). The use of the same address space for different devices implies in addition that a NAT operation must occur between two devices in order for them to communicate.

The security implications of address translation in general have been discussed in many previous documents, including [RFC2663], [RFC2993], [RFC4787], and [RFC5382].

7. References

7.1. Normative References

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.

7.2. Informative References

[L2NAT] Miles, D., Ed., and M. Townsley, "Layer2-Aware NAT", Work in Progress, March 2009.

[RFC1661] Simpson, W., Ed., "The Point-to-Point Protocol (PPP)", STD 51, RFC 1661, July 1994.

[RFC1918] Rekhter, Y., Moskowitz, R., Karrenberg, D., de Groot, G., and E. Lear, "Address Allocation for Private Internets", BCP 5, RFC 1918, February 1996.

[RFC2516] Mamakos, L., Lidl, K., Evarts, J., Carrel, D., Simone, D., and R. Wheeler, "A Method for Transmitting PPP Over Ethernet (PPPoE)", RFC 2516, February 1999.

[RFC2663] Srisuresh, P. and M. Holdrege, "IP Network Address Translator (NAT) Terminology and Considerations", RFC 2663, August 1999.

[RFC2993] Hain, T., "Architectural Implications of NAT", RFC 2993, November 2000.

[RFC4213] Nordmark, E. and R. Gilligan, "Basic Transition Mechanisms for IPv6 Hosts and Routers", RFC 4213, October 2005.

[RFC4787] Audet, F., Ed., and C. Jennings, "Network Address Translation (NAT) Behavioral Requirements for Unicast UDP", BCP 127, RFC 4787, January 2007.

[RFC5382] Guha, S., Ed., Biswas, K., Ford, B., Sivakumar, S., and P. Srisuresh, "NAT Behavioral Requirements for TCP", BCP 142, RFC 5382, October 2008.

[RFC6127] Arkko, J. and M. Townsley, "IPv4 Run-Out and IPv4-IPv6 Co-Existence Scenarios", RFC 6127, May 2011.

[RFC6146] Bagnulo, M., Matthews, P., and I. van Beijnum, "Stateful NAT64: Network Address and Protocol Translation from IPv6 Clients to IPv4 Servers", RFC 6146, April 2011.

[RFC6333] Durand, A., Droms, R., Woodyatt, J., and Y. Lee, "Dual-Stack Lite Broadband Deployments Following IPv4 Exhaustion", RFC 6333, August 2011.

[TRILOGY] "Trilogy Project", <http://www.trilogy-project.org/>.

Appendix A. Contributors

The ideas in this document were first presented in [RFC6333]. This document is also indebted to [RFC6127] and [L2NAT]. However, all of these documents focused on additional components, such as tunneling protocols or the allocation of special IP address ranges. We wanted to publish a specification that just focuses on the core functionality of per-interface NAT mappings. However, David Miles and Alain Durand should be credited with coming up with the ideas discussed in this memo.

Appendix B. Acknowledgments

The authors would also like to thank Randy Bush, Fredrik Garneij, Dan Wing, Christian Vogt, Marcelo Braun, Joel Halpern, Wassim Haddad, Alan Kavanaugh, and others for interesting discussions in this problem space.

Lars Eggert is partly funded by the Trilogy Project [TRILOGY], a research project supported by the European Commission under its Seventh Framework Program.

Authors' Addresses

Jari Arkko
Ericsson
Jorvas 02420
Finland

EMail: jari.arkko@piuha.net

Lars Eggert
NetApp
Sonnenallee 1
85551 Kirchheim
Germany

Phone: +49 151 12055791
EMail: lars@netapp.com
URI: http://eggert.org/

Mark Townsley
Cisco
Paris 75006
France

EMail: townsley@cisco.com