Operational Security Capabilities for
IP Network Infrastructure (opsec)
Internet-Draft
Intended status: BCP
Expires: October 29, 2012

F. Gont
UK CPNI

April 27, 2012

Security Implications of IPv6 on IPv4 Networks
draft-gont-opsec-ipv6-implications-on-ipv4-nets-01

Abstract

This document discusses the security implications of native IPv6 support and IPv6 transition/co-existence technologies on "IPv4-only" networks, and describes possible mitigations for the aforementioned issues.

Status of this Memo

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79. This document may not be modified, and derivative works of it may not be created, and it may not be published except as an Internet-Draft.

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 29, 2012.

Table of Contents

1. Introduction

Most general-purpose operating systems implement and enable by default native IPv6 support and a number of transition-co-existence technologies. In those cases in which such devices are deployed on networks that are assumed to be IPv4-only, the aforementioned technologies could be leveraged by local or remote attackers for a number of (illegitimate) purposes.

For example, a Network Intrusion Detection System (NIDS) might be prepared to detect attack patterns for IPv4 traffic, but might be unable to detect the same attack patterns when a transition/co-existence technology is leveraged for that purpose. Additionally, an IPv4 firewall might enforce a specific security policy in IPv4, but might be unable to enforce the same policy in IPv6. Finally, some transition/co-existence mechanisms (notably Teredo) are designed to traverse Network Address Translators (NATs), which in many deployments provide a minimum level of protection by only allowing those instances of communication that have been initiated from the internal network. Thus, these mechanisms might cause an internal host with otherwise limited IPv4 connectivity to become globally reachable over IPv6, therefore resulting in increased (and possibly unexpected) host exposure. That is, the aforementioned technologies might inadvertently allow incoming IPv6 connections from the Internet to hosts behind the organizational firewall.

In general, the aforementioned security implications can be mitigated by enforcing security controls on native IPv6 traffic and on IPv4-tunneled traffic. Among such controls is the enforcement of filtering policies, such that undesirable traffic is blocked.

This document discusses the security implications of IPv6 and IPv6 transition/co-existence technologies on (allegedly) IPv4-only networks, and provides guidance on how to mitigate the aforementioned issues.

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].

2. Security Implications of native IPv6 support

Most popular operating systems include IPv6 support that is enabled by default. This means that even if a network is expected to be IPv4-only, much of its infrastructure is nevertheless likely to be IPv6 enabled. For example, hosts are likely to have at least link-local IPv6 connectivity which might be exploited by attackers with access to the local network.

> [CORE2007] is a security advisory about a buffer overflow which could be remotely-exploited by leveraging link-local IPv6 connectivity that is enabled by default.

Additionally, unless appropriate measures are taken, an attacker with access to an 'IPv4-only' local network could impersonate a local router and cause local hosts to enable their IPv6 connectivity (e.g. by sending Router Advertisement messages), possibly circumventing security controls that were are enforced only on IPv4 communications.

> [THC-IPV6] is the first publicly-available toolkit that implemented this attack vector (along with many others).
>
> [Waters2011] provides an example of how this could be achieved using publicly available tools (besides incorrectly claiming the discovery of a "0day vulnerability").

In general, network SHOULD enforce on native IPv6 traffic the same security policies they currently enforce on IPv4 traffic. However, in those networks in which IPv6 has not yet been deployed, and enforcing the aforementioned policies is deemed as unfeasible, a network administrator MAY mitigate IPv6-based attack vectors by means of appropriate packet filtering.

2.1. Filtering Native IPv6 Traffic

Some layer-2 devices may have the ability to selectively filter packets based on the type of layer-2 payload. When such functionality is available, IPv6 traffic could be blocked at those layer-2 devices by blocking e.g. Ethernet frames with the Protocol Type field set to 0x86dd [IANA-ETHER].

SLAAC-based attacks [RFC3756] can be mitigated with technologies such as RA-Guard [RFC6105] [I-D.ietf-v6ops-ra-guard-implementation]. However, RA-Guard cannot mitigate attack vectors that employ IPv6 link-local addresses, since configuration of such addresses does not rely on Router Advertisement messages.

In order to mitigate attacks based on native IPv6 traffic, IPv6

security controls should be enforced on both IPv4 and IPv6 networks. The aforementioned controls might include: deploying IPv6-enabled NIDS, implementing IPv6 firewalling, etc.

In some very specific scenarios (e.g., military operations networks) in which only IPv4 service might be desired, a network administrator might MAY disable IPv6 support in all the communicating devices.

3. Security Implications of tunneling Mechanisms

Unless properly managed, tunneling mechanisms may result in negative security implications ([RFC6169] describes the security implications of tunneling mechanisms in detail). Therefore, tunneling mechanisms should be a concern not only to network administrators that have consciously deployed them, but also to network and security administrators whose security policies might be bypassed by exploiting these mechanisms.

> [CERT2009] contains some examples of how tunnels can be leveraged to bypass firewall rules.

To help mitigate these issues, a good security practice is to only allow traffic deemed as "necessary" (i.e., the so-called "default deny" policy). Therefore, security administrators SHOULD block (by default)IPv6 transition/co-existence traffic, and SHOULD only allow it as a result of an explicit decision, rather than as a result of lack of awareness about such traffic.

> It should be noted that this recommendation is aimed at a network that is the target of such traffic (such as an enterprise network). IPv6-transition traffic should not be filtered e.g. by an ISP when it is transit traffic.

Additionally, it is highly recommended that in those networks where specific transition mechanisms are not explicitly deployed, not only the corresponding traffic should be filtered at the organizational perimeter, but also the corresponding mechanisms disabled on each node connected to the organizational network. This not only prevents security breaches resulting from accidental use of these mechanisms, but also disables this functionality altogether, possibly mitigating vulnerabilities that might be present in the host implementation of this transition/co-existence mechanisms.

IPv6-in-IPv4 tunnelling mechanisms (such as 6to4 or configured tunnels) can generally be blocked by dropping IPv4 packets that contain a Protocol field set to 41. Security devices such as NIDS might also include signatures that detect such transition/ co-existence traffic.

3.1. Filtering 6in4

Probably the most basic type of tunnel employed for connecting IPv6 "islands" is the so-called "6in4", in which IPv6 packets are encapsulated within IPv4 packets. These tunnels are typically result from manual configuration at the two tunnel endpoints.

6in4 tunnels can be blocked by blocking IPv4 packets with a Protocol field of 41.

3.2. Filtering 6over4

[RFC2529] specifies a mechanism known as 6over4 or 'IPv6 over IPv4' (or colloquially as 'virtual Ethernet'), which comprises a set of mechanisms and policies to allow isolated IPv6 hosts located on physical links with no directly-connected IPv6 router, to become fully functional IPv6 hosts by using an IPv4 domain that supports IPv4 multicast as their virtual local link.

> This transition technology has never been widely deployed, because of the low level of deployment of multicast in most networks.

6over4 encapsulates IPv6 packets in IPv4 packets with their Protocol field set to 41. As a result, simply filtering all IPv4 packets that have a Protocol field equal to 41 will filter 6over4 (along with many other transition technologies).

A more selective filtering could be enforced such that 6over4 traffic is filtered while other transition traffic is still allowed. Such a filtering policy would block all IPv4 packets that have their Protocol field set to 41, and that have a Destination Address that belongs to the prefix 239.0.0.0/8.

This filtering policy basically blocks 6over4 Neighbor Discovery traffic directed to multicast addresses, thus preventing Stateless Address Auto-configuration (SLAAC), address resolution, etc. Additionally, it would prevent the 6over multicast addresses from being leveraged for the purpose of network reconnaissance.

3.3. Filtering 6rd

6rd builds upon the mechanisms of 6to4 to enable the rapid deployment of IPv6 on IPv4 infrastructures, while avoiding some downsides of 6to4. Usage of 6rd was originally documented in [RFC5569], and the mechanism was generalized to other access technologies and formally standardized in [RFC5969].

6rd can be blocked by blocking IPv4 packets with the Protocol field set to 41.

3.4. Filtering 6to4

6to4 [RFC3056] is an address assignment and router-to-router, host-to-router, and router-to-host automatic tunnelling mechanism that is meant to provide IPv6 connectivity between IPv6 sites and hosts

across the IPv4 Internet.

As discussed in Section 3, all IPv6-in-IPv4 traffic, including 6to4, could be easily blocked by filtering IPv4 that contain their Protocol field set to 41. This is the most effective way of filtering such traffic.

Additional filtering rules that might be incorporated include:

- Filter outgoing IPv4 packets that have their Destination Address set to an address that belongs to the prefix 192.88.99.0/24.

- Filter incoming IPv4 packets that have their Source Address set to an address that belongs to the prefix 192.88.99.0/24.

  It has been suggested that 6to4 relays send their packets with their IPv4 Source Address set to 192.88.99.1.

- Filter outgoing IPv4 packets that have their Destination Address set to the IPv4 address of well-known 6to4 relays.

- Filter incoming IPv4 packets that have their Destination Address set to the IPv4 address of well-known 6to4 relays.

These last two filtering policies will generally be unnecessary, and possibly unfeasible to enforce (given the number of potential 6to4 relays, and the fact that many relays may remain unknown to the network administrator). If anything, they should be applied with the additional requirement that such IPv4 packets have their Protocol field set to 41, to avoid the case where other services available at the same IPv4 address as a 6to4 relay are mistakenly made inaccessible.

If 6to4 traffic is meant to be filtered while other IPv6-in-IPv4 traffic is allowed, then the following filtering rules could be applied:

- Filter outgoing IPv4 packets that have their Protocol field set to 41, and that have an IPv6 Source Address (embedded in the IPv4 payload) that belongs to the prefix 2002::/16.

- Filter incoming IPv4 packets that have their Protocol field set to 41, and that have an IPv6 Destination address (embedded in the IPv4 payload) that belongs to the prefix 2002::/16.

3.5. Filtering ISATAP

ISATAP [RFC5214] is an Intra-site tunnelling protocol, and thus it is generally expected that such traffic will not traverse the organizational firewall of an IPv4-only. Nevertheless, ISATAP traffic is easily filtered as described in Section 3 of this document.

3.6. Filtering Teredo

Teredo [RFC4380] is an address assignment and automatic tunnelling technology that provides IPv6 connectivity to dual-stack nodes that are behind one or more Network Address Translators (NATs), by encapsulating IPv6 packets in IPv4-based UDP datagrams. Teredo is meant to be a 'last resort' IPv6 connectivity technology, to be used only when other technologies such as 6to4 cannot be deployed (e.g., because the edge device has not been assigned a public IPv4 address).

As noted in [RFC4380], in order for a Teredo client to configure its Teredo IPv6 address, it must contact a Teredo server, through the Teredo service port (UDP port number 3544).

To prevent the Teredo initialization process from succeeding, and hence prevent the use of Teredo, an organizational firewall could filter outgoing UDP packets with a Destination Port of 3544.

It is clear that such a filtering policy does not prevent an attacker from running its own Teredo server in the public Internet, using a non-standard UDP port for the Teredo service port (i.e., a port number other than 3544).

The most popular operating system that includes an implementation of Teredo in the default installation is Microsoft Windows. Microsoft Windows obtains the Teredo server addresses (primary and secondary) by resolving the domain name teredo.ipv6.microsoft.com into DNS A records. A network administrator may want to prevent Microsoft Windows hosts from obtaining Teredo service by filtering at the organizational firewall outgoing UDP datagrams (i.e., IPv4 packets with the Protocol field set to 17) that contain in the IPv4 Destination Address any of the IPv4 addresses that the domain name teredo.ipv6.microsoft.com maps to. Additionally, the firewall would filter incoming UDP datagrams from any of the IPv4 addresses to which the domain names of well-known Teredo servers (such as teredo.ipv6.microsoft.com) resolve.

As these IPv4 addresses might change over time, an administrator should obtain these addresses when implementing the filtering policy, and should also be prepared to maintain this list up to date.

The corresponding addresses can be easily obtained from a UNIX host by issuing the command 'dig teredo.ipv6.microsoft.com a' (without quotes).

It should be noted that even with all these filtering policies in place, a node in the internal network might still be able to communicate with some Teredo clients. That is, it could configure an IPv6 address itself (without even contacting a Teredo server), and might send Teredo traffic to those peers for which intervention of the host's Teredo server is not required (e.g., Teredo clients behind a cone NAT).

3.7. Filtering Tunnel Broker with Tunnel Setup Protocol (TSP)

The tunnel broker model enables dynamic configuration of tunnels between a tunnel client and a tunnel server. The tunnel broker provides a control channel for creating, deleting or updating a tunnel between the tunnel client and the tunnel server. Additionally, the tunnel broker may register the user IPv6 address and name in the DNS. Once the tunnel is configured, data can flow between the tunnel client and the tunnel server. [RFC3053] describes the Tunnel Broker model, while [RFC5572] specifies the Tunnel Setup Protocol (TSP), which can be used by clients to communicate with the Tunnel Broker.

TSP can use either TCP or UDP as the transport protocol. In both cases TSP uses port number 3653, which has been assigned by the IANA for this purpose. As a result, TSP (the Tunnel Broker control channel) can be blocked by blocking TCP and UDP packets originating from the local network and destined to UDP port 3653 or TCP port 3653. Additionally, the data channel can be blocked by blocking UDP packets originated from the local network and destined to UDP port 3653, and IPv4 packets with a Protocol field set to 41.

4. Security Considerations

This document discusses the security implications of IPv6 on IPv4 networks, and describes a number of techniques to mitigate the aforementioned issues. In general, the possible mitigations boil down to enforcing on native IPv6 and IPv6 transition/co-existence traffic the same security policies currently enforced for IPv4 traffic, and/or blocking the aforementioned traffic when it is deemed as undesirable.

5. Acknowledgements

The author would like to thank (in alphabetical order) Arturo Servin, for providing valuable comments on earlier versions of this document.

This document resulted from the project "Security Assessment of the Internet Protocol version 6 (IPv6)" [CPNI-IPv6], carried out by Fernando Gont on behalf of the UK Centre for the Protection of National Infrastructure (CPNI).

Fernando Gont would like to thank the UK CPNI for their continued support.

6. References

6.1. Normative References

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

[RFC2460] Deering, S. and R. Hinden, "Internet Protocol, Version 6 (IPv6) Specification", RFC 2460, December 1998.

[RFC2529] Carpenter, B. and C. Jung, "Transmission of IPv6 over IPv4 Domains without Explicit Tunnels", RFC 2529, March 1999.

[RFC3053] Durand, A., Fasano, P., Guardini, I., and D. Lento, "IPv6 Tunnel Broker", RFC 3053, January 2001.

[RFC3056] Carpenter, B. and K. Moore, "Connection of IPv6 Domains via IPv4 Clouds", RFC 3056, February 2001.

[RFC4380] Huitema, C., "Teredo: Tunneling IPv6 over UDP through Network Address Translations (NATs)", RFC 4380, February 2006.

[RFC5214] Templin, F., Gleeson, T., and D. Thaler, "Intra-Site Automatic Tunnel Addressing Protocol (ISATAP)", RFC 5214, March 2008.

[RFC5569] Despres, R., "IPv6 Rapid Deployment on IPv4 Infrastructures (6rd)", RFC 5569, January 2010.

[RFC5969] Townsley, W. and O. Troan, "IPv6 Rapid Deployment on IPv4 Infrastructures (6rd) -- Protocol Specification", RFC 5969, August 2010.

[RFC5572] Blanchet, M. and F. Parent, "IPv6 Tunnel Broker with the Tunnel Setup Protocol (TSP)", RFC 5572, February 2010.

6.2. Informative References

[RFC3756] Nikander, P., Kempf, J., and E. Nordmark, "IPv6 Neighbor Discovery (ND) Trust Models and Threats", RFC 3756, May 2004.

[RFC6105] Levy-Abegnoli, E., Van de Velde, G., Popoviciu, C., and J. Mohacsi, "IPv6 Router Advertisement Guard", RFC 6105, February 2011.

[RFC6169] Krishnan, S., Thaler, D., and J. Hoagland, "Security

Concerns with IP Tunneling", RFC 6169, April 2011.

[I-D.ietf-v6ops-ra-guard-implementation]
Gont, F., "Implementation Advice for IPv6 Router Advertisement Guard (RA-Guard)", draft-ietf-v6ops-ra-guard-implementation-02 (work in progress), March 2012.

[IANA-ETHER]
IANA, "Ether Types", 2012, <http://www.iana.org/assignments/ethernet-numbers>.

[CERT2009]
CERT, "Bypassing firewalls with IPv6 tunnels", 2009, <http://www.cert.org/blogs/vuls/2009/04/bypassing_firewalls_with_ipv6.html>.

[CORE2007]
CORE, "OpenBSD's IPv6 mbufs remote kernel buffer overflow", 2007, <http://www.coresecurity.com/content/open-bsd-advisorie>.

[CPNI-IPv6]
Gont, F., "Security Assessment of the Internet Protocol version 6 (IPv6)", UK Centre for the Protection of National Infrastructure, (available on request).

[THC-IPV6]
"The Hacker's Choice IPv6 Attack Toolkit", <http://www.thc.org/thc-ipv6/>.

[Waters2011]
Waters, A., "SLAAC Attack - 0day Windows Network Interception Configuration Vulnerability", 2011, <http://resources.infosecinstitute.com/slaac-attack/>.

Author's Address

Fernando Gont
UK Centre for the Protection of National Infrastructure

Email: fernando@gont.com.ar
URI: http://www.cpni.gov.uk