

IPv6 Maintenance (6man)
Internet-Draft
Updates: [4380](#) (if approved)
Intended status: Standards Track
Expires: March 12, 2011

F. Gont
UK CPNI
September 8, 2010

Mitigating Teredo Rooting Loop Attacks
draft-gont-6man-teredo-loops-00.txt

Abstract

Recently, a number of routing loop vulnerabilities were discovered in Teredo. This document specifies a number of security checks to be performed by Teredo hosts and Teredo servers such that these vulnerabilities are eliminated.

Status of this Memo

This Internet-Draft is submitted to IETF 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 March 12, 2011.

Copyright Notice

Copyright (c) 2010 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.

Internet-Draft

Teredo routing loops

September 2010

Table of Contents

- [1. Introduction](#) [3](#)
- [2. Teredo Attacks](#) [3](#)
 - [2.1. Teredo Client to NAT](#) [3](#)
 - [2.1.1. Operational considerations](#) [4](#)
 - [2.1.2. Implementation considerations](#) [4](#)
 - [2.2. Teredo Server](#) [4](#)
 - [2.2.1. Operational considerations](#) [5](#)
 - [2.2.2. Implementation considerations](#) [5](#)
- [3. Security Considerations](#) [5](#)
- [4. IANA Considerations](#) [6](#)
- [5. Acknowledgements](#) [6](#)
- [6. References](#) [6](#)
 - [6.1. Normative References](#) [6](#)
 - [6.2. Informative References](#) [6](#)
- [Appendix A. Changes from previous versions of the draft \(to
be removed by the RFC Editor before publishing
this document as an RFC\)](#) [6](#)
- [Author's Address](#) [6](#)

Internet-Draft

Teredo routing loops

September 2010

[1.](#) Introduction

[USENIX-WOOT] describes a number routing loop attacks that can be performed, in a number of scenarios, against IPv6 automatic tunneling mechanisms, possibly resulting in Denial of Service (DoS). This document discusses the two Teredo routing loop attacks described [USENIX-WOOT], and specifies a number of security checks such that these vulnerabilities are eliminated.

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].

[2.](#) Teredo Attacks

[2.1.](#) Teredo Client to NAT

This attack targets a Teredo client and the NAT(s) through which the Teredo client connects to the public Internet. It assumes that the NAT is of type "cone", and that the aforementioned NAT supports hair-pin routing with source address translation.

The attack is initiated by sending a Teredo packet, with its IPv4 Source Address and its IPv4 Destination Address set to the Teredo Mapped Address of the victim Teredo client, and the UDP Source Port and the UDP Destination Port set to the Teredo Mapped Port of the victim Teredo client. The IPv6 Source Address and IPv6 Destination Address of the encapsulated IPv6 packet are Teredo addresses, with their client IPv4 field and their Port field set to the "mapped IPv4 address" and the obfuscated "mapped UDP port" of the victim Teredo client, respectively. The C (cone) bit of the IPv6 Destination Address should be set to "1" (indicating a cone NAT) and the UG bits of the same address should be set to "00" (indicating a non-global unicast identifier). The Server IPv4 field and/or the other bits of the Flags field of the IPv6 Destination Address should be different

from that of the victim Teredo client, such that the resulting address is not the IPv6 address of the victim Teredo client.

The idea is that the forged IPv6 Source Address be such that it passes the source address validation checks recommended in [RFC4380]. The forged IPv6 Destination Address should cause the packet to be looped back to the victim Teredo client, but should not be the Teredo address of the victim Teredo client (or else the packet would be processed by the Teredo client and the loop would not occur).

Assuming that there already exists a corresponding mapping in the NAT

Gont

Expires March 12, 2011

[Page 3]

Internet-Draft

Teredo routing loops

September 2010

(as a result of the Teredo Initial Qualification Procedure), the victim Teredo client will receive the forged packet. [Nakibly and Arov, 2009] found that in some implementations, if the receiving node is in forwarding mode (i.e., it is acting as a router), it will forward the encapsulated IPv6 packet over the Teredo tunnel (as the victim Teredo client was not the final destination of the packet). This will result in a forwarding loop that will finish only when the Hop Limit field of the encapsulated IPv6 packet is decremented to 0, possibly leading to a Denial of Service (DoS).

There are a number of considerations that should be made about this attack vector. Some of these considerations are operational, while others have to do with the Teredo implementation at the victim Teredo client.

2.1.1. Operational considerations

Firstly, given the deployment model of Teredo, it seems unlikely that a node acting as a router would enable Teredo for obtaining its IPv6 connectivity. Secondly, enforcement of ingress/egress filtering would probably mitigate this attack (although it would not prevent a malicious node on the same network as the victim Teredo client from launching the attack).

2.1.2. Implementation considerations

Given that Teredo is a mechanism of "last resort" for obtaining IPv6 connectivity by IPv6 hosts, a node SHOULD NOT forward over the Teredo tunnel IPv6 packets that were not originated on the local node, and

SHOULD discard those packets received over the Teredo tunnel that are not destined to the Teredo client. These security checks completely eliminate this vulnerability.

2.2. Teredo Server

This attack vector engages only one victim, a Teredo server, and consists in having the Teredo server send a Teredo bubble destined to itself, which will result in a forwarding loop that will continue indefinitely.

As the Teredo server decapsulates the bubble packet (an empty IPv6 datagram) and re-encapsulates it in another IPv4 packet before forwarding it, there is no mechanism to limit the number of times a bubble packet is "forwarded".

The attack consists in sending a forged "Teredo bubble" with the IPv4 Source Address and the IPv4 Destination Address both set to the IPv4 address of the victim Teredo server, and the UDP Source Port and the

UDP Destination Port both set to the Teredo UDP Port (3544). The IPv6 Source Address and the IPv6 Destination Address of the encapsulated IPv6 packet should have their client IPv4 field set to the obfuscated IPv4 address of the victim Teredo server, and the their Port field set to the obfuscated Teredo UDP port (3544). The Server IPv4 field and the Flags field can be set to any value.

The idea is that the IPv6 Source Address must be such that the forged Teredo packet will pass the source address validation checks described in [[RFC4380](#)]. The IPv6 Destination Address must be such that the forged Teredo bubble is re-sent by the victim Teredo server to its own IPv4 address and Teredo UDP Port.

There are a number of considerations that should be made about this attack vector. Some of these considerations are operational, while others have to do with the Teredo implementation at the victim Teredo client.

2.2.1. Operational considerations

Implementation of ingress/egress filtering would probably mitigate this attack. However, ingres/egres filtering should not be relied

upon as the "first line of defense".

2.2.2. Implementation considerations

In order for this attack to succeed, a Teredo server must be willing to accept a Teredo packet that contains its own address in the IPv4 Source Address field, and accept the Source Address and the Destination Address of the encapsulated IPv6 packet to embed its own (obfuscated) address in the "client IPv4" field. There are no legitimate reasons for a Teredo packet to contain such values. Therefore, this vulnerability could be eliminated by having Teredo servers silently discard such Teredo packets.

Teredo servers MUST discard Teredo packets that have an IPv4 Source Address equal to one of the receiving server's IPv4 addresses, and MUST discard Teredo packets that embed the (obfuscated) IPv4 address of the receiving server in the "client IPv4" field of the Source Address or the Destination Address of the encapsulated IPv6 packet.

3. Security Considerations

This document updates [[RFC4380](#)] such that the two Teredo routing loop vulnerabilities described in [[USENIX-WOOT](#)] are eliminated.

Gont

Expires March 12, 2011

[Page 5]

Internet-Draft

Teredo routing loops

September 2010

4. IANA Considerations

This document has no actions for IANA.

5. Acknowledgements

The routing loop attacks against Teredo discussed in this document were discovered by Gabi Nakibly and Michael Arov, and documented in [[USENIX-WOOT](#)].

This document is heavily based on the upcoming document "Security Implications of the Internet Protocol version 6 (IPv6)" [[CPNI-IPv6](#)].

Fernando Gont would like to thank CPNI (<http://www.cpni.gov.uk>) 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.

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

6.2. Informative References

[CPNI-IPv6]

Gont, F., "Security Assessment of the Internet Protocol version 6 (IPv6)", UK Centre for the Protection of National Infrastructure, (to be published).

[USENIX-WOOT]

Nakibly, G., "Routing Loop Attacks using IPv6 Tunnels", USENIX WOOT, 2009.

[Appendix A.](#) Changes from previous versions of the draft (to be removed by the RFC Editor before publishing this document as an RFC)

Gont

Expires March 12, 2011

[Page 6]

Internet-Draft

Teredo routing loops

September 2010

Author's Address

Fernando Gont
UK Centre for the Protection of National Infrastructure

Email: fernando@gont.com.ar

URI: <http://www.cpni.gov.uk>

