Network Working Group
Internet-Draft
Expires May 2002

Scott Bradner
HT Kung
Harvard University
November 2001

Requirements for an Anonymizing Packet Forwarder

<draft-bradner-annfwd-req-00.txt>

1. Status of This Memo

This document is an Internet-Draft and is subject to all provisions of Section 10 of RFC 2026.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF), its areas, and its working groups. Note that other groups may also distribute working documents as Internet-Drafts.

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."

The list of current Internet-Drafts can be accessed at http://www.ietf.org/ietf/1id-abstracts.txt.

The list of Internet-Draft Shadow Directories can be accessed at http://www.ietf.org/shadow.html.

2. Abstract

There are a number of situations in the Internet where it would be useful to be able to have an application be able to send traffic to a destination without revealing the IP address of the destination to the source, or the IP address of the source to the destination, or both. One way to do this is to have a network resident set of servers which can forward packets, with encryption and decryption applied to their source and destination addresses when appropriate. We will call this server an anonymizing forwarder.

This memo describes the requirements for such a server. A companion document [FRW-DRAFT] will describe a proposed framework on the usage of anonymizing forwarders and how these requirements are applicable.

3. Conventions Used in This Document

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 [2].

4. Background

This memo describes the requirements for an anonymizing packet forwarder to be used in situations such as the following.

A client needs to interact with a server (the "target server") but the IP address of the target server needs to be hidden in order to minimize the potential for denial of service (DoS) attacks on the target server. The client first interacts with an initialization server, which may be a local application or a network-based server. This initialization server securely sends the client a message that includes the IP address of the target server encrypted in the key of a set of forwarding servers and an IP address that can be used to reach the same set of forwarding servers. Additional information can also be included to be used for authenticating the request. The forwarding servers can be using anycast-style addressing (see [RFC1546] and [ANYCAST]).

When the client wishes to send a message to the target server, it builds the packet without including the destination IP address, and sends this along with the encrypted address of the target server to the IP address for the forwarding servers. The forwarding server decrypts the target server address. If additional authenticating information is present, the forwarding server can check that information before proceeding. When all checks have been completed, the forwarding server builds a packet by taking the packet supplied by the client and inserting the IP address of the target server into the destination address field. The packet is then sent to the target server.

The design of the anonymizing forwarding server described in this memo assumes that the use will be for low-bandwidth signaling, not data transfer that may require high bandwidth. This assumption is made so that rate limiting can be used to minimize the chance that the anonymizing forwarding server could be used to hide a denial of service attack.

There are many applications that fit the model of signaling defined here. These include request/response messages in connection setup and termination, user authentication, service registration, and service discovery. These types of applications only need medium bandwidth to function properly.

The anonymizing forwarding server of this type differs from other

address translation servers or devices such as NAT/VPN/Proxy. First, as mentioned above, the anonymizing forwarding server is for signaling, not for data transfer. Second, it is a stateless packet forwarder. In particular, it does not keep any connection or session state or any mapping tables that, for example, could be used to generate translated IP addresses and port numbers. A consequence is that the anonymizing forwarding server is oblivious to the number of connections or sessions, and does not have translation tables to manage.

The anonymizing forwarding server performs a function of decrypting a packet to recover the IP address of the next forwarding hop or the final destination. This function is similar to that performed by a node in onion routing [ONION]. The anonymizing forwarding server differs from onion routing in that it is intended to provide a stand-alone anonymizing network infrastructure at the IP layer.

There are other application level anonymizers, including mixmaster for anonymous email forwarding [MIX] and anonymizing proxy [TRUST].

The anonymizing forwarding server intends to contribute to the goal of supporting anonymity at the IP layer, as envisioned by the "Controlled Nymity IP" effort [NymIP].

5. Threat Model

One of the main threats we are trying to deal with is revealing the IP address of the target server, but in doing so, we do not want our anonymizing forwarders to become the conduit for denial of service attacks.

Since all traffic going in and out an anonymizing forwarding server is visible, our approaches need to be resistant to traffic analysis.

An anonymizing forwarding server may be compromised due to reasons such as password leak and broken systems. In addition, an imperfection of a system that hides the location of a forwarder and its output links may allow an adversary to monitor the output links. In these cases, the IP address of the target server could be revealed to the adversary. To mitigate this threat, various approaches can be taken. For example, the forwarding infrastructure can forward a packet in multiple hops using forwarders under different management authorities. Forwarders can hide their existence by hiding themselves behind other forwarders. Forwarders can also use different IP addresses at times. In this case, initialization servers will need to be synchronized with the change of forwarders' IP addresses. When the risk of compromised forwarders is determined to be sufficiently high, the IP address of the target server can be

changed.

An anonymizing forwarding server can itself be a target of denial of service attacks. It needs to be resistant to such attacks.

In this requirement we are not specifically concerned with protecting the packet contents between the client and forwarding server since we expect the end to end packet contents will be encrypted.

6. Requirements

Forwarding Function: The forwarding server MUST be able to receive an arriving packet on a port, encrypt, decrypt and extract some of the contents of the packet, select the IP address of the next hop using the decrypted content, verify the source, build the outgoing packet, and forward it to the next hop.

Stateless: The forwarding server MUST NOT keep any state about individual packet transmissions in order to do its work. This means that packets of a given application session may use different forwarding servers. All messages to a forwarding server MUST contain all the information that the forwarding server will need to verify the source, build the packet and forward it to the next hop. All information about a particular packet SHOULD be discarded as soon as the transmission is complete. The destination and source IP addresses SHOULD be kept, but SHOULD NOT be in a correlated way, so that rate limiting can be done (see below).

Anycast: The forwarding server MAY use anycast-style addressing (see [RFC1546] and [ANYCAST]), so that any of a number of forwarding servers using the same anycast address may forward a packet sent to it.

Multi-hop Forwarding: A forwarding server MAY support multi-hop forwarding, under which a client's packet may be forwarded by a sequence of two or more forwarders before reaching its target server. These forwarders together are sufficient in decrypting the IP address of the target server, but a subset of them are not. Multi-hop forwarding can provide added protection against an adversary who attempts to compromise a forwarder or monitor its output links. For example, by involving forwarders protected under strong but different security measures in multi-hop forwarding, we can reduce the chance that a set of forwarders that are sufficient for the decryption are all compromised. That is, the adversary would need to defeat all the security measures, rather than just one of them, in order to succeed.

Scalable Deployment: Forwarding servers MUST be able to be deployed

in such a way that the number of servers can be raised as the demand increases.

Overall Rate Limiting: Each forwarding server SHOULD be able to limit the maximum rate at which it will forward messages in order to minimize the ease with which it could be used in a denial of service attack.

Per Next-hop Rate Limiting: In addition to putting a limit on the overall rate that a single forwarding server will forward traffic to a next-hop forwarder or to any one destination, there SHOULD be a maximum limit on the rate at which traffic can be sent to any of them.

Per Preceding-hop Rate Limiting: Similar to the above, there MAY be a maximum rate at which traffic will be forwarded for any preceding-hop forwarder or for any single source.

DoS Resistant: The forwarding infrastructure itself SHOULD be denial of service resistant. Any forwarding server that may receive packets directly from clients should be able to sustain requests arriving at the wire speed. This means that the server should not have extra services running that could prevent it from achieving wire-speed performance. It SHOULD also be protected against SYN and other well-known attacks.

Managing Request Queue: For any forwarding server that may receive packets directly from clients, its queue of forwarding requests SHOULD be DoS resistant as well. A Random Early Discard [RED] queue management could be used to discard requests when they arrive at too high a rate.

Traffic Analysis Resistant: The forwarding server SHOULD provide means of inhibiting an adversary from correlating input and output traffic, and thus inferring the IP address or location of the next hop or the target server. For example, the forwarding server MAY randomly delay packets, make all output packets of the same size, randomize their transmission order, and insert false traffic.

Statistics and logging: The forwarding server MAY provide facilities for statistics gathering and logging purposes, provided the information does not correlate input and output IP addresses.

Encryption Schemes: The address of the target host MUST be so encrypted that clients of the forwarding server and other unauthorized parties cannot see the address before it is decrypted. The key infrastructure required by the encryption scheme SHOULD be scalable to support a large number of clients and a moderate number

of forwarding servers.

7. Security Considerations

We have identified two areas of security concerns. The first is the necessity of minimizing the risk that forwarding servers could be used to launch DoS attacks. The rate limiting requirement is one of the methods to be used to address this issue. The second is the need for a key management infrastructure to support encryption in the forwarding server's key, and to prevent the server from being compromised. Existing and future public-key infrastructure (PKI) could be exploited.

8. References

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

[MIX] Chaum, D. L., "Untraceable Electronic Mail, Return Addresses, and Digital Pseudonyms," Communications of the ACM February 1981 Volume 24 Number 2.

[RED] Floyd, S., and Jacobson, V., "Random Early Detection gateways for Congestion Avoidance," Volume 1, Number 4, August 1993, p. 397-413.

[ANYCAST] Katabi, D., and Wroclawski, J., "A Framework for Global IP-Anycast (GIA)," Proceedings of ACM SIGCOMM 2000, Stockholm, Sweden, 2000.

[FRW-DRAFT] Kung, H. T. and Bradner, S., " A Framework for an Anonymizing Packet Forwarder" <draft-kung-annfwd-framework.txt>, Draft, November 2001

[NymIP] The NymIP Effort, http://nymip.velvet.com.

[RFC1546] Milliken, W., Partridge C., and Mendez, T., "Host anycasting service," RFC 1546, November 1993.

[ONION] Reed, M., Syverson, P., and Goldschlag, D., "Anonymous Connections and Onion Routing," IEEE Journal on Selected Areas in Communications, vol. 16 no. 4, May 1998, pp. 482-494.

[TRUST] Waldman, M., Cranor, L. F., and Rubin, A., "Trust," in P2P: Harnessing the Benefits of a Disruptive Technology, edited by A. Oram, O'Reilly & Associates, Sebastopol, California 2001, pp.

242-270.

8. Authors' Addresses

Scott Bradner
Harvard University
29 Oxford St.
Cambridge MA 02138

Email: sob@harvard.edu
Phone: +1-617-495-3864

HT Kung
Harvard University
33 Oxford St.
Cambridge MA 02138

Email: kung@harvard.edu
Phone: +1-617-496-6211