Status of this Memo

This Internet-Draft is submitted to IETF in full conformance with the provisions of BCP 78 and BCP 79. This document may contain material from IETF Documents or IETF Contributions published or made publicly available before November 10, 2008. The person(s) controlling the copyright in some of this material may not have granted the IETF Trust the right to allow modifications of such material outside the IETF Standards Process. Without obtaining an adequate license from the person(s) controlling the copyright in such materials, this document may not be modified outside the IETF Standards Process, and derivative works of it may not be created outside the IETF Standards Process, except to format it for publication as an RFC or to translate it into languages other than English.

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.

This Internet-Draft will expire on April 23, 2010.

Copyright Notice

Copyright (c) 2009 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
Abstract

The Source Address Validation Improvement protocol was developed to complement ingress filtering with finer-grained, standardized IP source address validation. To facilitate deployment in networks of various kinds, the SAVI protocol was designed to be modular and extensible. This document describes and motivates the design of the SAVI protocol, explains how the protocol is composed of components, and compares the properties of alternative protocol components.

Table of Contents

1. Introduction ................................................. 3
2. Protocol Model ............................................. 3
3. Deployment Options ......................................... 5
4. Scalability Optimizations ................................. 6
5. Acknowledgment ............................................. 8
6. References .................................................. 8
Author’s Address .............................................. 8
1. Introduction

Since IP source addresses are used by hosts and network entities to determine the origin of a packet and as a destination for return data, spoofing of IP source addresses can enable impersonation, concealment, and malicious traffic redirection. Unfortunately, the Internet architecture alone fails to prevent IP source address spoofing. Since the IP source address of a packet generally takes no role in forwarding the packet, it can be selected arbitrarily by the sending host without jeopardizing packet delivery. Extra methods are necessary for IP source address validation, to augment packet forwarding with an explicit check of whether a given packet’s IP source address is legitimate.

IP source address validation can happen at different granularity: Ingress filtering [BCP38], a widely deployed standard for IP source address validation, functions at the coarse granularity of networks. It verifies that the prefix of an IP source address routes to the network from which the packet was received. An advantage of ingress filtering is simplicity: The decision of whether to accept or to reject an IP source address can be made solely based on the information available from routing protocols. Unfortunately, the simplicity comes at the cost of not being able to validate IP source addresses at a finer granularity, due to the aggregated nature of the information available from routing protocols. Finer-grained IP source address validation would be helpful to enable IP-source-address-based authentication, authorization, and host localization, as well as to efficiently identify misbehaving hosts and impose protective measures. Partial solutions [BA2007] exist for finer-grained IP source address validation, but are proprietary and hence often unsuitable for corporate procurement.

The Source Address Validation Improvement protocol was developed to complement ingress filtering with standardized IP source address validation at the maximally fine granularity of individual IP addresses: It prevents hosts attached to the same link from spoofing each other’s IP addresses. To facilitate deployment in networks of various kinds, the SAVI protocol was designed to be modular and extensible. This document describes and motivates the design of the SAVI protocol, explains how protocol components fit together, and compares the properties of alternative protocol components.

2. Protocol Model

To enable network operators to deploy fine-grained IP source address validation without a dependency on supportive functionality on hosts, the SAVI protocol was designed to be purely network-based. A SAVI
1. Identify which IP source addresses are legitimate for a host, based on monitoring packets exchanged by the host.

2. Bind a legitimate IP address to a link layer property of the host’s network attachment. This property, called a "binding anchor", must be verifiable in every packet that the host sends, and harder to spoof than the host’s IP source address itself.

3. Enforce that the IP source addresses in packets match the binding anchors to which they were bound.

The closer a SAVI protocol instance is located to the hosts, the more effective the SAVI protocol is. This is because each of the three steps of the SAVI protocol model can best be accomplished in a position close to the host:

- Identifying a host’s legitimate IP source addresses is most efficient close to the host, because the likelihood that the host’s packets bypass a SAVI protocol instance, and hence cannot be monitored, increases with the distance between the protocol instance and the host.

- Selecting a binding anchor for a host’s IP source address is easiest close to the host, because many link layer properties are unique for a given host only on a link segment directly attaching to the host.

- Enforcing a host’s use of a legitimate IP source address is most reliable when pursued close to the host, because the likelihood that the host’s packets bypass a SAVI protocol instance, and hence do not undergo IP source address validation, increases with the distance between the protocol instance and the host.
The preferred location of SAVI protocol instances is therefore close to hosts, such as in switches that directly attach to the hosts whose IP source addresses are being validated.

3. Deployment Options

The model of the SAVI protocol, as explained in Section 2, is deployment-specific in two ways:

- The identification of legitimate IP source addresses is dependent on the IP address assignment method in use on a link, since it is through assignment that a host becomes the legitimate user of an IP source address.

- Binding anchors are dependent on the technology used to build the link on which they are used, as binding anchors are link layer properties of a host’s network attachment.

To facilitate the deployment of the SAVI protocol in networks of various kinds, the SAVI protocol is designed to support different IP address assignment methods, and to function with different binding anchors. Naturally, both the IP address assignment methods in use on a link and the available binding anchors have an impact on the strength of IP source address validation. This impact further motivates a prioritization of IP address assignment methods, to resolve situations in which a single IP source address is attempted to be bound to different binding anchors through different IP address assignment methods. The following two sub-sections explain the impact that IP address assignment methods and binding anchors have on the strength of IP source address validation, as well as the prioritization of IP address assignment methods defined for the SAVI protocol.

3.1. IP Address Assignment Methods

To be added: Enumeration and prioritization of IP address assignment methods. This is to include references to the SAVI specifications for DHCP-assigned, SLAAC-assigned, and SEND-assigned IP addresses.

3.2. Binding Anchors

To be added: Enumeration of binding anchors, along with a discussion of the security properties of those.
4. Scalability Optimizations

The preference to locate a SAVI protocol instance close to hosts implies that multiple SAVI protocol instances must be able to co-exist in order to support large links. Although the SAVI protocol model is independent of the number of protocol instances per link, co-existence of multiple protocol instances without further measures can lead to higher-than-necessary memory requirements: Since a SAVI protocol instance creates bindings for the IP source addresses of all hosts on a link, bindings are replicated if multiple protocol instances co-exist on the link. High memory requirements, in turn, increase the cost of a SAVI protocol instance. This is problematic in particular for SAVI protocol instances that are located on a switch, since it may significantly increase the cost of such a switch.

To reduce memory requirements for SAVI protocol instances that are located on a switch, the SAVI protocol enables the suppression of binding replication on links with multiple protocol instances. This requires manual disabling of IP source address validation on switch ports that connect to other switches running a SAVI protocol instance. Each SAVI protocol instance is then responsible for validating IP source addresses only on those ports to which hosts attach either directly, or via switches without a SAVI protocol instance. On ports towards other switches running a SAVI protocol instance, IP source addresses are not validated. The switches running SAVI protocol instances thus form a "protection perimeter". The IP source addresses in packets passing the protection perimeter are validated by the ingress SAVI protocol instance, but no further validation takes place as long as the packets remain within, or leave the protection perimeter.
Figure 1: Protection perimeter concept

Figure 1 illustrates the concept of the protection perimeter. The figure shows a link with six switches, of which four, denoted "SAVI switch", run a SAVI protocol instance. The protection perimeter created by the four SAVI protocol instances is shown as a dotted line in the figure. IP source address validation is enabled on all switch ports on the protection perimeter, and it is disabled on all other switch ports. Four hosts, denoted A through D in the figure, attach to the protection perimeter.

In the example of figure Figure 1, the protection perimeter encompasses one of the legacy switches, located in the middle of the depicted link topology. This enables a single, unpartitioned protection perimeter. A single protection perimeter minimizes memory requirements for the SAVI protocol instances because every binding is kept only once, namely, by the SAVI protocol instance that attaches to the host being validated. Excluding the legacy switch from the protection perimeter would result in two smaller protection perimeters to the left and to the right of the depicted link topology. The memory requirements for the SAVI protocol instances would then be higher: Since IP source address validation would be activated on the two ports connecting to the legacy switch, the SAVI protocol instances adjacent to the legacy switch would replicate all bindings from the respectively other protection perimeter. The reason why it is possible to include the legacy switch in the protection perimeter is because the depicted link topology guarantees
that packets cannot enter the protection perimeter via this legacy switch. Without this guarantee, the legacy switch would have to be excluded from the protection perimeter in order to ensure that packets entering the protection perimeter undergo IP source address validation.

5. Acknowledgment

The author would like to thank the SAVI working group for a thorough technical discussion on the design and the framework of the SAVI protocol, as captured in this document, in particular Jianping Wu, Jun Bi, Fred Baker, Guang Yao, Marcelo Bagnulo, Erik Nordmark, Eric Levy-Abegnoli, and Alberto Garcia. Thanks also to Torben Melsen for reviewing this document.

This document was generated using the xml2rfc tool.

6. References


Author’s Address

Christian Vogt (editor)
Ericsson
200 Holger Way
San Jose, CA 95134
United States

Email: christian.vogt@ericsson.com