Distributed Source Address Validation (DSAV) Framework
draft-li-dsav-framework-01

Abstract

This document provides an overall framework of Distributed Source Address Validation (DSAV) including both intra-domain and inter-domain levels. It also describes related considerations.

Requirements Language

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 8174 [RFC8174].

Status of This Memo

This Internet-Draft is submitted 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 https://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 15 July 2022.

Copyright Notice

Copyright (c) 2022 IETF Trust and the persons identified as the document authors. All rights reserved.
Source address validation (SAV) is important to mitigate source address spoofing and contribute to the Internet security. However, existing SAV mechanisms have limitations in accuracy. Specifically, intra-domain SAV mechanisms (e.g. strict uRPF [RFC3704]) usually improperly block legitimate traffic in the case of routing asymmetry, while inter-domain SAV mechanisms (e.g. loose uRPF [RFC3704] and EFP-uRPF [RFC8704]) provide overly loose SAV rules which can improperly permit spoofed traffic. The root cause of their limitations is that they all achieve SAV based on local forwarding information base (FIB) or routing information base (RIB), which may not match the real forwarding direction from the source. In order to guarantee the accuracy, SAV should follow the real data-plane forwarding path.

This document provides a framework to generate accurate SAV rules on routers at both intra-domain and inter-domain levels. In Distributed Source Address Validation (DSAV) framework, each router or AS originates individual protocol messages to its neighbors, carrying local source information and corresponding destination information which takes the neighbor as the next hop. Upon receiving a protocol message, the router or AS identifies a valid incoming interface for the related source addresses. After that, it decides whether to terminate the message or further relay new protocol messages to its
neighbors based on the destination information of the received message. In this way, the source information will propagate through all possible forwarding paths originated from the source.

This document also describes basic considerations related to DSAV, including accuracy, consistency, deployability, and security.

2. Terminology

Some definitions during a propagation process:

* Node: A router or AS in this document.

* Initial node: The node generating original protocol messages.

* Terminal node: The node terminating the received protocol message from a neighbor node.

3. DSAV Framework

DSAV provides a framework for distributedly generating SAV rules on nodes at both intra-domain and inter-domain levels. Intra-domain SAV avoids source address spoofing within the same AS. Inter-domain SAV prevents source address spoofing among ASes. Despite of different application scenarios and protocol details, DSAVs at the two levels hold the same key idea. The core workflow of DSAV is briefly described as follows:

a. An initial node A generates an original message for each neighbor node, carrying the source prefixes originated locally and the destination prefixes which take the neighbor node as the next hop.

b. When node B receives a protocol message at interface I, it determines interface I as a valid incoming interface for the source prefixes of the received message. In other words, it generates the SAV rule <source prefixes of A, interface I>.
c. After that, node B checks the destination prefixes of the received message against its local FIB/RIB. If the next hop of all the destination prefixes point to its local subnets/networks, the message is terminated; otherwise, node B relays new messages. It groups all destination prefixes according to their next-hop node. For each next-hop node C, node B generates a new message destined to C, with the corresponding destination prefixes taking node C as the next hop. The source prefixes in each relayed message should keep the same.

d. In DSAV, with the exception of some special cases, such as multipath routing, nodes usually receive only one message originated from each source.

e. The above steps can be executed periodically or when any of source prefixes, destination information, or forwarding paths change. The updated message should add updated SAV rules or delete outdated SAV rules for the affected Nodes. Particularly, to reduce the communication overhead, only the changed information should be propagated again when dynamic updating.

Figure 1 illustrates the workflow of DSAV. The network runs some routing protocols such as OSPF, IS-IS, or BGP among the four nodes. Each node owns a unique source prefix (e.g. P1 for Node 1). Let’s consider the propagation process where Node 1 is the initial node. Node 1 sends an original message to the neighbor Node 2, carrying its source prefix (i.e., P1) and destination prefixes whose next-hop node in FIB is Node 2 (i.e., P2, P3, P4). When Node 2 receives the message, it specifies interface ‘#’ as the valid incoming interface for prefix P1. Then, Node 2 checks the destination prefixes according to its local FIB. Since P3 and P4 are not Node 2’s source prefixes, it should relay messages to the corresponding next-hop nodes, i.e. Node 3 and Node 4. The message destined to Node 3 carries the destination prefix P3, while the message destined to Node 4 carries the destination prefix P4. The source prefix in each relayed message keeps the same. When Node 3 or Node 4 receives the message from Node 2, it also learns and enables the SAV rule <P1, interface ‘#’> but terminates the message.
- P1, P2, P3, and P4 are prefixes belonging to Node 1, 2, 3, and 4, respectively.
- Node 1 is the initial node, and Node 3 and Node 4 are the terminate nodes in this propagation process.
- ‘#’ means the legitimate interface for the data-plane packets with source addresses of P1.

Figure 1: The workflow of DSAV

3.1. Separate Source Information Advertisement

Containing source prefixes and destination prefixes in a message sometimes induces much unnecessary overhead. For example, a change on a destination prefix or forwarding path will make the initial node advertise its source prefixes again even though no changes happen on its local source prefixes at all. A separate source information advertisement is taken to tackle the above problem.

Particularly, a node can be represented by a node ID (e.g., the router-ID for a router or the ASN for an AS). For each initial node, its source prefixes together with its node ID can be advertised to other nodes through broadcast or existing underlay routing protocols (such as OSPF, IS-IS, and BGP). Then, other nodes will know the mapping from a node ID to a list of source prefixes. Now, the protocol message does not need to carry a long list of source prefixes whose field can be replaced with just one source node ID.
3.2. Destination Information Identifier

Although separate source information advertisement help reduce communication overhead, including destination prefixes in messages can still be costly, especially in inter-domain scenarios with a large number of destination prefixes.

Similarly, a list of destination prefixes can also be replaced with a destination node IDs (e.g., the router-ID for a router or the ASN for an AS). Considering that a node may have hundreds of different prefixes, this can significantly reduce overhead. However, the replacement of destination prefixes may result in accuracy problems in some scenarios where the destination prefixes belonging to a same destination node have different forwarding paths. Some additional mechanisms need to be imported into these scenarios.

4. Accuracy

The goal of DSAV is to achieve high accuracy, i.e., avoid improper block problems and try best to reduce improper permit problems. The improper block problem means legitimate traffic is mistakenly dropped. The improper permit problem means spoofed traffic is mistakenly passed.

The accuracy of DSAV is determined by the accuracy of source information advertisement and propagation process. The incompleteness of received source information can compromise the accuracy of SAV. So, each initial node should discover and advertise local source information carefully with the help of either automatic programs or manual configurations. In the case of incomplete source information, the node can take a remedy method at the data plane, i.e., only drop packets with known source addresses but coming from invalid interfaces. Packets with unknown source addresses should be accepted by default. More details will be described in Section 6.

The key of DSAV is to generate SAV rules strictly following the real data-plane forwarding paths. Any factor that can affect forwarding should be considered. Here are three kinds of common forwarding cases:

* Only FIBs affect forwarding.
* ECMP (Equal-cost multi-path routing) or UCMP (Unequal-cost multi-path routing). To achieve multi-path routing, hashing functions are usually taken, which map packet header field values (e.g., source/destination IP address, source/destination port number, protocol number) to candidate next hops. Packets with the same destination IP address may be forwarded to different next hops.

* ACL redirection. An ACL rule can have multiple match fields, and the match field of destination IP addresses can be included or not in an ACL rule. So, similar to ECMP/UCMP, the packets with the same destination IP address may have different next-hop interfaces.

As described in Section 3, DSAV can work well in the first case. To ensure accuracy in arbitrary routing scenarios, the last two cases should also be considered.

5. Consistency

The factors influencing the accuracy of DSAV may change with time. Such changes will lower the performance of SAV and lead to improper block or improper permit problems. The SAV rules generated through DSAV should be updated in time so as to keep consistent with routing states. The consistency of DSAV is important for the SAV framework working well in real networks.

A simple method is to send updated messages periodically. An aging mechanism can also be used for SAV rules. That is, SAV rules will expire after a period of time. However, these solutions may take much time before eliminating improper block and improper permit problems. Some quick convergence mechanisms are necessary to achieve consistency of DSAV in time. Here are some preliminary ideas for different cases:

* Source information changes. A node sends new source information advertisements immediately upon discovering its local source prefixes change.

* Routing state changes. When route configuration or topology changes, the forwarding path to a destination prefix may change. These changes can trigger the initial node to generate updated messages for the changed forwarding paths. Then, new SAV rules can be added and outdated SAV rules can be withdrawn at other nodes quickly. For the scenarios where fast reroute (FRR) is deployed, the initial node can send message to the backup forwarding paths in advance, and the backup SAV rules can be installed for fast convergence under failures.
6. Deployability

It is difficult to ensure that all nodes deploy DSAV simultaneously, especially at inter-domain level. In this case, each node only learns partial source address information or incomplete legitimate incoming interfaces for a source prefix, which can lead to improper block problems. Therefore, DSAV should support incremental and partial deployment.

When deployed incrementally or partially, nodes should still avoid improper block problems and minimize improper permit problems based on incomplete SAV tables. The process of data-plane SAV is as follows:

* For the source address whose source address information and incoming interface information are fully learned, nodes can strictly validate the authenticity by querying <source prefix, interface> in SAV tables.

* For the source address whose source address information or incoming interface information is only partially learned or even not learned, nodes should pass those packets by default to avoid improper block problems, since it is hard to identify the authenticity with incomplete information.

Since inter-domain topology is greatly complex and ASes are managed by individual network operators, determining whether the incoming interface information for a source prefix is learned completely is a real challenge. Besides, in DSAV framework, neighboring (next-hop) node plays an important role in the propagation of probing packets, namely, a node cannot send or receive any probing packet if its neighboring nodes don’t support DSAV. Hence, at inter-domain level, DSAV recommends incremental deployment by customer cones. This deployment pattern ensures that each AS learns complete source address information and incoming interface information for other ASes within the same customer cone. With the merger of different customer cones where DSAV is deployed, the deployment scope of DSAV will gradually expand, and the defense capability against source address spoofing will gradually increase.

7. Security

TBD

8. Normative References


Authors' Addresses

Dan Li
Tsinghua University
Beijing
China

Email: tolidan@tsinghua.edu.cn

Jianping Wu
Tsinghua University
Beijing
China

Email: jianping@cernet.edu.cn
Abstract

This document identifies scenarios where existing IP spoofing approaches for detection and mitigation don’t perform perfectly. Existing SAV (source address validation) approaches, either Ingress ACL filtering [RFC2827], unicast Reverse Path Forwarding (uRPF) [RFC3704], Feasible Path uRPF [RFC 3704], or Enhanced Feasible-Path uRPF [RFC8704] has limitations regarding either automated implementation objective or detection accuracy objective (0% false positive and 0% false negative). This document provides the gap analysis of the existing SAV approaches, and also provides solution discussions.

Requirements Language

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].
1. Introduction

1.1. Source Address Validation

The Internet is open to traffic, which means that a sender can generate traffic and send to any receiver in the Internet as long as the address is reachable. Although this openness design improves the scalability of the Internet, it also leaves security risks, e.g., a sender can forge the source address when sending the packets, which is also known as IP spoofing. IP spoofing is constantly used in Denial of Service (DoS) attacks, which seriously compromise network security. DoS attacks using IP spoofing makes it difficult for operators to locate the attacker’s actual source address. [RFC6959] identifies different types of DoS attacks with IP spoofing, i.e., single-packet attack, flood-based DoS, poisoning attack, spoof-based...
worm/malware propagation, reflective attack, accounting subversion, man-in-the-middle attack, third-party recon, etc.

1.2. Existing SAV Techniques Overview

Source address validation (SAV) verifies the authenticity of the packet’s source address to detect and mitigate IP spoofing [RFC2827]. Existing methods, such as Source Address Validation Improvement (SAVI) [RFC7039], unicast Reverse Path Forwarding (uRPF) (i.e., Strict uRPF, Feasible uRPF and Loose uRPF) [RFC3704], as well as Enhanced Feasible-Path Unicast Reverse Path Forwarding (EFP-uRPF) methods [RFC8704] are deployed at different network levels to prevent IP spoofing.

Overall, when evaluating a SAV technique, one should consider the following two perspectives.

1) Precise filtering: Two important indicators for precise filtering.
   1) 0% false positive (FP) rate. If legitimate packets are dropped, it can seriously affect the user experience. 2) 0% false negative (FN) rate. If some packets with a forged source address passes, it poses potential security risks.

2) Automatic implementation: In practice, the address space may grow, and routing policies may be dynamically adjusted. SAV solutions that rely entirely on manual configuration are either non-scalable or error-prone.

SAVI, typically performed at the access network, is enforced in switches, where the mapping relationship between an IP address and other "trust anchor" is maintained. A "trust anchor" can be link-layer information (such as MAC address), physical port of a switch to connect a host, etc. It enforces hosts to use legitimate IP source addresses. However, given numerous access networks managed by different operators, it is far from practice for all the access networks to simultaneously deploy SAVI. Therefore, in order to mitigate the security risks raised by source address spoofing, SAV performed in network border routers is also necessary. Although it does not provide the same filtering granularity as SAVI does, it still helps the tracing of spoofing to a minimized network range.

Ingress ACLs [RFC2827], typically performed at the network border routers, is performed by manually maintaining a traffic filtering access list which contains acceptable source address for each interface. Only packets with a source address encompassed in the access list can be accepted. It strictly specifies the source address space of incoming packets. However, manual-based filtering method is error-prone and face scalability issues.
Strict uRPF, typically performed at the network (IGP areas or ASes) border routers, requires that a data packet can be only accepted when the FIB contains a prefix that encompasses the source address and the corresponding out-interface matches the data incoming interface. It has the advantages of simple operation, easy deployment, and automatic update. However, in case of multihoming, when the data incoming interface is different from the out-interface, which is also referred to as asymmetric routing of data packets, Strict uRPF exhibits FP.

Loose uRPF, sacrificing the directionality of Strict uRPF, only requires that the packet’s source IP exists as a FIB entry. Intuitively, Loose uRPF cannot prevent the attacker from forging a source address that already exists in the FIB, which incurs FN detection.

Feasible uRPF (FP-uRPF), typically performed at the network border routers, helps mitigate FP of Strict uRPF in the multihoming scenarios. Instead of installing only the best route into FIB as Strict uRPF does, Feasible uRPF installs all alternative paths into the FIB. It helps reduce FP filtering compared with the Strict uRPF, in the case when multiple paths are learnt from different interfaces. However, it should be noted that Feasible uRPF only works when multiple paths are learnt. There are cases when a device only learns one path but still has packets coming from other valid interfaces. Thus, FP-uRPF performs better than Loose uRPF regarding FP detection, but still doesn’t not guarantee 0% FP.

EFP-uRPF, specifically performed at the AS border routers, further improves FP-uRPF in the inter-AS scenario. An ASBR, performing EFP-uRPF, maintains an RPF filtering list on each customer/peer interface. It introduces two algorithms (i.e., Algorithm A and Algorithm B) regarding different application scenarios. In the case that a customer interface fails to learn any route from a directly connected customer AS, enabling Algorithm A at this customer interface may exhibit false positive detection. In this case, Algorithm B can mitigate the FP. However, in case of two customer ASes spoofing each other, Algorithm B exhibits FN.

This document specifically identifies two scenarios, where the above mentioned SAV techniques, i.e., Strict uRPF, Loose uRPF, FP-uRPF, and EFP-uRPF, fail to guarantee 0% FP and 0% FN detection.

2. Terminology

IGP: Interior Gateway Protocol

IS-IS: Intermediate System to Intermediate System
BGP: Boarder Gateway Protocol
RIB: Routing Information Base
FIB: Forwarding Information Base
SAV: Source Address Validation
AD: Administrative Domain

3. Problem Statement

3.1. Use Case 1: Inter-AS Multi-homing

Figure 1 illustrates an inter-AS multihoming case.

AS2 is multi-homed to AS1 and AS4. AS2 announces P1/P2 to AS1 through BGP. AS2 doesn’t announce any of its routes to AS4 due to policy control. P1/P2 are propagated from AS1 to AS4 through BGP.

AS3 is single-homed to AS4. AS3 announces P3 to AS4 through BGP. AS4 propagates P3 to AS1 through BGP.

Now suppose two data flows coming from AS2 to AS4: Flow 1 with source IP as P1, and Flow 2 with source IP as P3 (IP spoofing). Using existing SAV methods at AS4, Flow 1 is supposed to be passed, while Flow 2 is supposed to be dropped.

- Loose uRPF: works for Flow 1, but fails for Flow 2.
- Strict uRPF: works for Flow 2, but fails for Flow 1 (the incoming interface does not match P1/P2’s out-interface).
- FP-uRFP: works for Flow 2, but fails for Flow 1 (no feasible path for P1/P2 other than the best route exists).
- EFP-uRPF: works for Flow 1, but fails for Flow 2 using Algorithm B. Works for Flow 2, but fails for Flow 1 when using Algorithm A.
3.2. Use Case 2: Intra-AS Multi-homing

Figure 2 illustrates an intra-AS multihoming case. To facilitate management, one AS can be divided into several administrative domains (ADs) and managed by different inner groups. In Figure 2, AD1 is the upper level compared to AD2 and AD3, meaning that AD2 or AD3 needs to connect through AD1 for external reachability (i.e., networks outside AD1). For example, AD1 is the backbone of one national education network, while AD2 and AD3 are the campus networks of the two universities.

Router 1 is multi-homed to Router 2 and Router 3. No dynamic routing protocol set up between Router 1 and Router 2, as well as between Router 1 and Router 3. In AD2, static routes to outside AD2 are configured on Router 1 with Router 3 as the next hop. In AD1, static route to P1 is configured on Router 2 and static route to P2 is configured on Router 3, due to traffic control purpose. Router 2 and Router 3 are connected with each other using ISIS or OSPF.

Router 5 is single-homed to Router 3. In AD3, static routes to outside AD3 are configured on Router 5 with Router 3 as the next hop. In AD3, static route to P3 is configured on Router 3 with Router 5 as the next hop.
Now suppose two data flows coming from Router 1 to Router 3: Flow 1 with source IP as P1, and Flow 2 with source IP as P3 (IP spoofing). Using existing SAV methods at Router 3, Flow 1 is supposed to be passed, while Flow 2 is supposed to be dropped.

- Loose uRPF: works for Flow 1, but fails for Flow 2.
- Strict uRPF: works for Flow 2, but fails for Flow 1 (the incoming interface does not match P1’s out-interface).
- FP-uRFP: works for Flow 2, but fails for Flow 1 (no feasible path for P1 other than the best route exists).
- EFP-uRPF: does not apply at the intra-AS case.

![Diagram of asymmetric data flow in the Intra-AS scenario](image)

Figure 2: Asymmetric data flow in the Intra-AS scenario
4. Solution Discussions

Both EFP-uRPF and FP-uRPF try to achieve a balance between flexibility (Loose uRPF) and directionality (Strict uRPF).

In the inter-AS multi-homing scenario, EFP-uRPF further improves FR-uRPF’s directionality. The key improvement of EFP-uRPF is that it synchronizes certain information between interfaces that share the same RPF filtering list, so as to construct an RPF list as comprehensive as possible, although [RFC8704] does not explicitly specify how the information is synchronized, e.g., what information, in which format and in which way. In addition, the construction of RPF lists can be further augmented with data from Route Origin Authorization (ROA) [RFC6482], as well as Internet Routing Registry (IRR) data. In fact, the global availability of ROA and IRR databases provides a secondary information synchronization approach. However, EFP-uRPF still fails to achieve 0% FN and 0% FP in case of Figure 1. Further information synchronization between interfaces might provide further improvement.

The above description works similarly for the intra-AS scenario. Information synchronization is also required in order to achieve higher filtering accuracy.

5. Security Considerations

TBD

6. Contributors

TBD

7. Acknowledgments

TBD

8. Normative References

[I-D.brockners-inband-oam-requirements]
Brockners, F., Bhandari, S., Dara, S., Pignataro, C., Gredler, H., Leddy, J., Youell, S., Mozes, D., Mizrahi, T., <>, P. L., and R. Chang, "Requirements for In-situ OAM", draft-brockners-inband-oam-requirements-03 (work in progress), March 2017.
[I-D.ietf-grow-bmp-adj-rib-out]

[I-D.ietf-grow-bmp-local-rib]

[I-D.ietf-netconf-yang-push]

[I-D.openconfig-rtgwg-gnmi-spec]

[I-D.song-ntf]


Authors’ Addresses

Dan Li
Tsinghua
Beijing
China
Email: tolidan@tsinghua.edu.cn

Jianping Wu
Tsinghua
Beijing
China
Email: jianping@cernet.edu.cn
Source Address Validation: Use Cases and Gap Analysis
draft-li-sav-gap-analysis-01

Abstract

This document identifies the importance and use cases of source
address validation (SAV) at both intra-domain level and inter-domain
level (see [RFC5210]). Existing intra-domain and inter-domain SAV
mechanisms, either Ingress ACL filtering [RFC2827], unicast Reverse
Path Forwarding (uRPF) [RFC3704], or Enhanced Feasible-Path uRPF
(EFP-uRPF) [RFC8704] has limitations in scalability or accuracy.
This document provides gap analysis of the existing SAV mechanisms.

Requirements Language

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 8174 [RFC8174].

Status of This Memo

This Internet-Draft is submitted 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 https://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 15 July 2022.
1. Introduction

Source Address Validation (SAV) is important for defending against source address forgery attacks and accurately tracing back to the attackers. Considering that the Internet is extremely large and complex, it is very difficult to solve the source address spoofing problem at a single "level" or through a single SAV mechanism. On the one hand, it is unrealistic to require all networks to deploy a single SAV mechanism. On the other hand, the failure of a single SAV mechanism will completely disable SAV.

To address the issue, Source Address Validation Architecture (SAVA) was proposed [RFC5210]. According to the operating feature of the Internet, SAVA presents a hierarchical architecture which carries out source IP address validation at three checking levels, i.e., access network, intra-domain, and inter-domain. Different levels provide
different granularities of source IP address authenticity. In contrast to the single-level/point model, SAVA allows incremental deployment of SAV mechanisms while keeps effective because of its multiple-fence design. So, enhancing the source IP address validity in all the three checking levels is of high importance. Furthermore, one or more independent and loosely-coupled SAV mechanisms can coexist and cooperate under SAVA, which is friendly to different users (e.g., providers) with different policies or considerations. Obviously, the quality of SAV mechanisms for their target checking levels is key to the performance of SAV.

There are many SAV mechanisms for different checking levels. For the access network level, Source Address Validation Improvement (SAVI) was proposed to force each host to use legitimate source IP address [RFC7039]. SAVI acts as a purely network-based solution without special dependencies on hosts. It dynamically binds each legitimate IP address to a specific port/MAC address and verifies each packet’s source address through the binding relationship. One of the most attractive features of SAVI is that it supports the maximally fine granularity of individual IP addresses, which previous ingress filtering mechanisms cannot provide.

At the intra-domain level, static Access Control List (ACL) is a typical solution of SAV. Operators can configure some matching rules to specify which kind of packets are acceptable (or unacceptable). The information of ACL should be updated manually so as to keep consistent with the newest filtering criteria, which inevitably limits the flexibility and accuracy of SAV. Strict unicast Reverse Path Forwarding (uRPF) [RFC3704] is another solution suitable to intra-domain. Routers deploying strict uRPF accept a data packet only when i) the local FIB contains a prefix encompassing the packet’s source address and ii) the corresponding forwarding action for the prefix matches the packet’s incoming interface. Otherwise, the packet will be dropped. However, in the scenarios (e.g., multihoming cases) where data packets are under asymmetric routing, strict uRPF often improperly blocks legitimate traffic.

At the inter-domain level, a combination of Enhanced Feasible-Path uRPF (EFP-uRPF) and loose uRPF is recommended in [RFC8704]. Particularly, EFP-uRPF is suggested to be applied on customer interfaces. EFP-uRPF on an AS can prevent its customers from spoofing its upstream ASes’ source addresses but fails in the case of two customers spoofing each other. On lateral peer interfaces and transit provider interfaces, loose uRPF [RFC3704] is taken. The routers deploying loose uRPF accept any packets whose source addresses appear in the local FIB tables. Due to the loss of directionality, loose uRPF often improperly permits spoofed traffic.
To summarize, given that it is impossible to deploy SAVI on every access network in the Internet, the "fences" at intra- and inter-domain levels are very important for filtering source address forgery packets that are let go by access networks. However, there exist some instinctive drawbacks in the existing SAV mechanisms designed for both the intra- and inter-domain levels, which leads to inevitable improper permit or improper block problems. A more complete SAV mechanism is required for both intra- and inter-domain levels.

This document identifies the use cases of intra- and inter-domain SAVs. These cases will help analyze the instinctive drawbacks of the existing SAV mechanisms. After that, some SAV requirements will be presented.

2. Terminology

EAST-WEST traffic denotes the traffic originated and terminated within an AS. Intra-domain SAV aims to check EAST-WEST traffic and prevents hosts/routers from spoofing other source IP blocks in the same AS.

NORTH-SOUTH traffic denotes the traffic arriving from an external AS. Particularly, the traffic arriving from the customer AS is Northward traffic. The traffic received from the provider/peer AS is Southward traffic. Inter-domain SAV aims to verify the authenticity of the source address of NORTH-SOUTH traffic.

3. Use Cases

Figure 1 illustrates the use cases of SAV in both intra- and inter-domain levels. AS1-AS5 belong to the same customer cone, and AS1 is the stub AS. The topology of AS2 is presented while other ASes’ inner structures are hidden for brevity.
3.1. Use Case 1: Intra-domain SAV

In some scenarios especially very large ASes, hosts/routers in the same AS may spoof each other’s IP addresses. In Figure 1, Router2 spoofs P1 that originates from Router3. With Intra-domain SAV, EAST-WEST traffic can be checked, and source address spoofing attacks can be prevented. In the figure, Router1, Router3, and Router4 will drop the packets with P1’ while accept those with P1, when they deploy Intra-domain SAV mechanisms. Overall, Intra-domain SAV can prevent the source address spoofing from the same AS.

Figure 1: Illustration of the use cases of SAV in both intra- and inter-domain levels

P1 is the source IP address prefix of Router3. P1’ is the spoofed P1 by Router2 located in the same AS as Router3. P1" is the spoofed P1 by Routers located in another AS, i.e., AS4.
3.2. Use Case 2: Inter-domain SAV

In Figure 1, AS4 spoofs AS2’s IP address prefix, i.e., P1 originated from Router3. AS5 will receive the Northward traffic from AS2 and AS4 with legitimate and spoofed IP addresses, respectively. An SAV mechanism is necessary for AS5 to drop the illegal traffic. From the viewpoint of Southward traffic, AS1 may also receive spoofed traffic from AS3 (if AS3 accepts the data packets with source prefix P1). So, the deployment of SAV on AS1 is also important. Overall, Inter-domain SAV is necessary and can improve the confidence of the source IP address validity among ASes.

4. Gap Analysis

High accuracy is the basic requirement of any intra- or inter-domain SAV mechanism. For any SAV mechanism, improper block problems must be avoided because legitimate traffic must not be influenced. On that basis, SAV should also reduce improper permit problems as much as possible. However, existing SAV mechanisms can not well meet these requirements.

4.1. Existing Intra-domain SAV mechanisms

Operators can configure static ACLs on border routers to validate source addresses. The main drawback of ACL-based SAV is the high operational overhead. Limited application scenarios make the ACL-based method unable to do sufficient SAV on EAST-WEST traffic.

Strict uRPF can generate SAV tables automatically, but it also has limited application scenarios. Figure 2 illustrates an intra-domain scenario. In the scenario, AS1 runs strict uRPF. An access network having IP address prefix 10.0.0.0/15 is attached to two border routers (Router1 and Router2) of AS1. Due to customer’s policy, it advertises 10.0.0.0/16 to Router1 and 10.1.0.0/16 to Router2. Then, Router1 and Router2 will advertise the learned IP address prefixes to other routers in AS1 through intra-domain routing protocols such as OSPF and IS-IS.

Although customer only advertises 10.0.0.0/16 to Router1, it may send packets with source IP addresses belonging to 10.1.0.0/16 to Router1 due to load balancing requirements. Suppose the destination node is Router5. Then the path to destination is Customer->Router1->Router3->Router5, while the reverse path is Router5->Router4->Router2->Customer. The round trip routing path is asymmetric, which cannot be dealt with well by strict uRPF.
Specifically speaking, strict uRPF is faced with improper block problems under asymmetric routing scenarios. When Router1/Router3 runs strict uRPF, it learns SAV rules that packets with source address prefix of 10.0.0.0/16 must enter the router on interface ‘#’. When the packets with source addresses of 10.1.0.0/16 arrive, they will be dropped, which results in improperly blocking legitimate traffic. Similarly, when strict uRPF is deployed on Router2, the improper block problem still exists.

![Diagram of intra-domain scenario]

Figure 2: An intra-domain scenario

4.2. Existing Inter-domain SAV mechanisms

The most popular inter-domain SAV is suggested by [RFC8704], which combines EFP-uRPF algorithm B and loose uRPF. In particular, EFP-uRPF algorithm B is for Northward traffic validation. It sacrifices the directionality of customer interfaces for reducing improper permit cases. Loose uRPF is for validating Southward traffic on lateral peer and transit provider interfaces. It sacrifices directionality of Southward traffic completely. Such a combined method sacrificing directionality will leads to improper permit problems sometimes.

Figure 3 illustrates a common inter-domain scenario where the above inter-domain SAV method will fail. In the figure, there are two customer ASes, i.e., AS1 and AS2. Both of them are attached to a
provider AS, i.e., AS4. AS4 has a lateral peer and a provider, i.e., AS3 and AS5. Particularly, AS1 has IP address prefix P1 and advertises it to AS4. IP address prefix P2 is allocated to AS2 and is also advertised to AS4. AS3 has IP address prefix P3 and AS5 has IP address prefix P5. P3 and P5 are also advertised to AS4 through BGP. All arrows represent BGP advertisements. Assume AS4 deploys inter-domain SAV policies, i.e., a combination of EFP-uRPF algorithm B and loose uRPF.

For Northward traffic, AS4 applies EFP-uRPF. Under EFP-uRPF, AS4 will generate SAV rules considering P1 and P2 are legitimate on both the two customer interfaces. When AS1 spoofs IP address prefix P2 of AS2, the malicious Northward traffic cannot be filtered by AS4. The same is true when AS2 forges P1 of AS1. That is to say, EFP-uRPF cannot prevent source address spoofing among customers even though it only focus on Northward traffic.

For Southward traffic, AS4 deploys loose uRPF for the interfaces of AS3 and AS5. It will learn that the packets with source addresses of P3 or P5 can be accepted without validating the specific arrival interface. Since loose uRPF loses directionality completely, it obviously will fail in dealing with the source address spoofing between its lateral peer and provider, i.e., AS3 and AS5.
5. SAV Requirements

High accuracy, i.e., avoiding improper block problems while trying best to reduce improper permit problems, is the basic requirement of an ideal SAV mechanism. As described above, existing SAV mechanisms cannot meet this requirement. The root cause of their limitations is that they all achieve SAV based on local forwarding information base (FIB) or routing information base (RIB), which may not match the real forwarding direction from the source. In order to guarantee the accuracy, SAV should follow the real data-plane forwarding path. To solve this problem and provide accurate SAV for arbitrary network scenarios, it is required to exchange/explore/probe the forwarding-path information among routers/ASes. In other words, network-wide protocols should be considered.

The network-wide protocols should also consider some practical issues:

* High scalability. The protocols should not induce much overhead (e.g., bandwidth cost of path probing). Fast convergence under environment changes is also important for improving the scalability in different scales of networks.

* High deployability. A strategy of incremental deployment needs to be considered. If some routers/ASes do not support the new protocols, improper block should be avoided.

* High security. The protocols should include mechanisms to guarantee the integrity of protocol packets. Security risks such as Man-in-the-Middle Attack should be avoided.

6. Security Considerations

TBD

7. Contributors

TBD

8. Acknowledgments

TBD

9. Normative References


Authors’ Addresses

Dan Li
Tsinghua University
Beijing
China

Email: tolidan@tsinghua.edu.cn

Jianping Wu
Tsinghua University
Beijing
China

Email: jianping@cernet.edu.cn
Practical Inter-Domain Source Address Validation
draft-xu-psav-00

Abstract

Because the Internet forwards packets according to the IP destination address, packet forwarding typically takes place without inspection of the source address and malicious attacks have been launched using spoofed source addresses. The inter-domain source address validation architecture is an effort to enhance the Internet by using state machine to generate consistent tags. When communicating between two end hosts at different ASes, tags will be added to the packets to identify the authenticity of the IP source address.

This memo introduces PSAV, an Inter-AS source address validation mechanism.

Status of This Memo

This Internet-Draft is submitted 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 https://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 16 August 2022.

Copyright Notice

Copyright (c) 2022 IETF Trust and the persons identified as the document authors. All rights reserved.
IP spoofing has been a long-recognized threat to Internet security for decades. Inter-domain source address validation (SAV) has long served as the primary defense mechanism due to its better cost-effectiveness. However, over years of effort, the deployment of inter-domain source address validation is still not optimistic. An important reason for this is the difficulty of balancing the clear security benefits of partial deployments with the scalability of large-scale deployments. uRPF [RFC5635], for example, routing-based schemes to filter spoofed traffic, which may result in a lack of security benefits due to the dynamic nature of routing or incomplete information caused by partial deployments. And while cryptography-based schemes such as IPsec [RFC4301] can provide clear security gains, the additional end-to-end overhead will present new challenges in scalability.
This document provides a framework of practical inter-domain SAV (PSAV). PSAV is a cryptography-based SAV to guarantee consistent security benefits. Key maintenance is performed between the source and destination ASes, and the key is used to generate packet tags to validate the authenticity of the source address. Meanwhile, in PSAV, ASes are organized as a hierarchical structure to provide scalability, in which only fully-connected key maintenance is performed between ASes on the same layer, and ASes between different layers achieve end-to-end source address validation through cross-layer validation and tag replacement.

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, BCP 14 [RFC2119] and indicate requirement layers for compliant CoAP implementations.

2. Terminology and Abbreviation

<table>
<thead>
<tr>
<th>Abbreviation</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>TA</td>
<td>Trust Alliance, the IPv6 network that uses the SAVA-X mechanism.</td>
</tr>
<tr>
<td>ACS</td>
<td>AD Control Server, the server that maintains state machine with other ACS and distribute information to AER.</td>
</tr>
<tr>
<td>ABR</td>
<td>AS or AS community border router, which is placed at the boundary of an AS of trust alliance.</td>
</tr>
<tr>
<td>Tag</td>
<td>The authentic identification of source address of a packet.</td>
</tr>
</tbody>
</table>

Table 1

3. PSAV Framework

PSAV is a cryptography-based end-to-end inter-domain source address verification method that guarantees security benefits at partial deployment. PSAV implements inter-AS tag maintenance by establishing a hierarchical community structure that utilizes border nodes on the forwarding path for tag replacement and validation. This mainly includes the following components.
1. Tag generation. In PSAV, the packet tag is generated by maintaining the key between ASes and using the generation algorithm. The destination AS will validate the source address by the packet tag. The above process requires a mapping relationship between AS-IP-Prifix-Key, which will be provided based on existing Internet infrastructure, e.g., such as RPKI, ROVER, etc.

2. Hierarchical structure. In PSAV, AS is organized into hierarchical AS communities, which can provide good scalability by reducing the tag maintenance overhead in large-scale deployments, managing the validation responsibilities corresponding to address allocation, and shielding external community changes. To implement tag validation in AS communities, PSAV will provide corresponding tag cross-layer validation and replacement methods.

3. Membership configuration. AS sends join, exit, or update to all participating nodes through a specific message format, and the participating nodes further complete membership configuration by verifying the authenticity of the messages to form a distributed consensus.

A typical workflow of PSAV is shown in Figure 1. AS1 joins the PSAV trust alliance with the signed join information, maintains the packet tag with AS2. After that, AS1 sends out the packet with Tag <AS1, AS2>, and AS2 validates it and replaces the Tag with <AS2, AS3>. Then AS3 validates and replaces the tag with <AS3, AS4>. After AS4 validation, confirm that the packet source address is true.
4. Control Plane

The functions of control plane of PSAV includes AS community information management, ACS-ACS communication, and ACS-ABR communication.

To eliminate the impact of routing dynamic caused by BGP or other routing protocols, PSAV requires its own AS community information management. These information of one AS includes AS Number (ASN), AS Community Number (ASCN), IP Prefix and Public Key. PSAV does not bind any methods of inter-domain mapping information, and it can both use centralized or distributed methods to maintain AS community information independently. When an AS or AS community wants to join or exit the Trust Alliance constituted by all the member ASes and AS communities, it SHOULD submit an certificate signing request message containing its own information. It also needs to submit such CSR message for updating its information recorded by all members in trust alliance.

The communication among ACSes is to maintain the tags used in packets in network. PSAV provides a tag generation mechanism on one-to-one state machine. In this mechanism, each AS or AS community needs one ACS. ACS negotiates initial state of the state machine with its
relavent ACS. The state transfers to the next state triggered by

time flies. For crossing different layer of AS communities, it is

used the tag generated by the state machine maintained by AS or AS

community with its direct paternal AS community in PSAV. The

communication between ACS and ABR is to deliver the AS community

information and tags.

PSAV requires a heart-beat mechanism for service availability

implemented in ACS-ACS communication and ACS-ABR communication.

When it detects that one ACS or ABR has ‘died’, the other end WOULD

remove its tag generation mechanism maintained with this ‘died’ end

and sends a request message to force execute the exit trust alliance

process of the other end.

5. Data Plane

The functions of data plane of PSAV includes prefix checking and tag

processing.

The tag delivered from the control plane indicates the source address

of one packet is not tampered. As the tag in use is generated by

one-to-one state machine pair, it MUST be completely consistent at

the same time.

It needs to divide the role of different interfaces of an ABR for

functioning properly. In ABR, the interface takes the role of

INGRESS, EGRESS, or TRUST. The INGRESS port links to the devices

inside the AS or AS community, the EGRESS port links to the devices

outside the AS or AS community, and the TRUST port links to the ABR

inside the same AS or AS community. The INGRESS port validates and

removes the tag in use. The EGRESS port adds or replaces the tag in

the packet. The TRUST port does nothing to the packet.

When a packet arrives at the ABR, it SHOULD be checked its source

address and destination address first. If it originates and

destinates the trust alliance, it MUST be tagged with a tag at the

first hop and removed tag at the last hop. When this packet forwards

crossing different layers of AS communities, it SHOULD be replaced

with relavent tags maintained by its ACS with direct paternal ACS.

In ABR, it maintains two mapping tables to record the AS community

information and tags in use. The AS-Prefix mapping table preserves

the ASN or ASCN and IP address prefix relationships. The AS-Tag

mapping table holds the ASN or ASCN and relevant tags. When a packet

is needed to add, replace, or remove tag, the ABR WOULD get the ASN

or ASCN which the packet belongs to first via the source address of

the packet from the AS-Prefix mapping table. The ABR WOULD obtain

the tag should be used by the ASN or ASCN from the AS-Tag mapping

table.

6. Consistency

PSAV is a cryptography-based source address validation mechanism to guarantee consistent security benefits and provide scalability for different deployment scales and validation granularity. PSAV uses the hierarchical structure to reduce the size of the secret symmetric keys to cut down the maintenance overhead. Hierarchy validation filters malicious traffic as early as possible to avoid wasting network resources. PSAV also provides clear security responsibilities corresponding to IP address allocation authority.

7. Scalability

7.1. Compatibility

Hierarchy effectively blocks external changes and provides scalability in large-scale deployments. AS the forwarding path is independent of the tag validation by using a mechanism for crossing different layers, PSAV is a segmented end-to-end cryptography scheme essentially. So it does not need to obtain the routing information and has nothing influence on existing routing infrastructure. Meanwhile, PSAV supports that packets can pass through networks where PSAV has not yet been deployed without affecting validation as it is end-to-end validation in nature, which is guaranteeing a definite security benefit for the deployer without requiring a deployment rate.

7.2. Expansion Management

On one hand, PSAV effectively isolates structural changes outside the community from internal nodes, as the hierarchical community design minimizes the impact of changes on the rest of the system. On the other hand, PSAV can be implemented with any existing distributed consensus algorithm for inter-AS consensus infrastructure. It should be noted that PSAV has no special requirements for the efficiency of this process based on the assumption that AS community information does not change frequently. Therefore, the decentralized maintenance approach can further reduce the management complexity of the expansion process.

8. Security Consideration
8.1. Attack towards community information

The distributed method to maintain the AS community information MAY suffer from the consistency challenges, such as witch attacks and eclipse attacks. However, the situation in PSAV is different from the normal distributed consensus scenario. Due to the hierarchical structure of PSAV, the failure of consensus on local community information does not affect other non-adjacent communities in the system. At the same time, the updated community information only needs the signature confirmation of its parent, brother and child communities, which means that the attack on the special node needs to hold specific resources, which further increases the difficulty of the attack.

8.2. Attacks towards Initial Status Negotiation

This is the problem posed in the PSAV implementation. As the clock-synchronized state machine will run locally after the initial status negotiation stage, the attacker can only attack on this negotiation. However, when the ACS-ACS pair or ACS-ABR pair is going to connect, the SSL/TLS will be used to guarantee security in communication. Therefore PSAV can ensure that attackers cannot obtain the initial status even if it can eavesdrop the negotiation packet online.

8.3. Tag Guessing and Key Cracking

For resisting reply attack, the eventual tag used in a packet is generated by the ABR with hashing a five-tuple including the signature generated from the state machine, the source address, the destination address, the first 8-bit of payload and source address prefix length. The attacker could guess the tag and crack that key using brute force. Nevertheless, it depends on the irreversibility of a Hash function to prevent backstepping the key from the tag. Furthermore, to decrease such probability, the signature generated from the state machine will be updated periodically.

9. IANA Consideration

TBD.

10. Acknowledgements

TBD.

11. Normative References


Authors’ Addresses

Ke Xu
Computer Science, Tsinghua University
Qinghuayuan street, Haidian District
Beijing
100084
China

Email: xuke@tsinghua.edu.cn

Jianping Wu
Computer Science, Tsinghua University
Qinghuayuan street, Haidian District
Beijing
100084
China

Email: jianping@cernet.edu.cn
Xiaoliang Wang  
Computer Science, Tsinghua University  
Qinghuayuan street, Haidian District  
Beijing  
100084  
China  
Email: wangxiaoliang0623@foxmail.com

Yangfei Guo  
Institute for Network Sciences and Cyberspace, Tsinghua University  
Qinghuayuan street, Haidian District  
Beijing  
100084  
China  
Email: guoyangf19@mails.tsinghua.edu.cn