Internet Engineering Task Force (IETF)
Request for Comments: 8704
BCP: 84
Updates: 3704
Category: Best Current Practice
ISSN: 2070-1721

K. Sriram
D. Montgomery
USA NIST
J. Haas
Juniper Networks, Inc.

February 2020

Enhanced Feasible-Path Unicast Reverse Path Forwarding

Abstract

This document identifies a need for and proposes improvement of the unicast Reverse Path Forwarding (uRPF) techniques (see RFC 3704) for detection and mitigation of source address spoofing (see BCP 38). Strict uRPF is inflexible about directionality, the loose uRPF is oblivious to directionality, and the current feasible-path uRPF attempts to strike a balance between the two (see RFC 3704). However, as shown in this document, the existing feasible-path uRPF still has shortcomings. This document describes enhanced feasible-path uRPF (EFP-uRPF) techniques that are more flexible (in a meaningful way) about directionality than the feasible-path uRPF (RFC 3704). The proposed EFP-uRPF methods aim to significantly reduce false positives regarding invalid detection in source address validation (SAV). Hence, they can potentially alleviate ISPs' concerns about the possibility of disrupting service for their customers and encourage greater deployment of uRPF techniques. This document updates RFC 3704.

Status of This Memo

This memo documents an Internet Best Current Practice.

This document is a product of the Internet Engineering Task Force (IETF). It represents the consensus of the IETF community. It has received public review and has been approved for publication by the Internet Engineering Steering Group (IESG). Further information on BCPs is available in Section 2 of RFC 7841.

Information about the current status of this document, any errata, and how to provide feedback on it may be obtained at https://www.rfc-editor.org/info/rfc8704.

Copyright Notice

Table of Contents

1. Introduction

Source address validation (SAV) refers to the detection and mitigation of source address (SA) spoofing [RFC2827]. This document identifies a need for and proposes improvement of the unicast Reverse Path Forwarding (uRPF) techniques [RFC3704] for SAV. Strict uRPF is inflexible about directionality (see [RFC3704] for definitions), the loose uRPF is oblivious to directionality, and the current feasible-path uRPF attempts to strike a balance between the two [RFC3704]. However, as shown in this document, the existing feasible-path uRPF still has shortcomings. Even with the feasible-path uRPF, ISPs are often apprehensive that they may be dropping customers' data packets with legitimate source addresses.

This document describes enhanced feasible-path uRPF (EFP-uRPF) techniques that aim to be more flexible (in a meaningful way) about directionality than the feasible-path uRPF. It is based on the principle that if BGP updates for multiple prefixes with the same origin AS were received on different interfaces (at border routers), then incoming data packets with source addresses in any of those prefixes should be accepted on any of those interfaces (presented in Section 3). For some challenging ISP-customer scenarios (see Section 3.3), this document also describes a more relaxed version of the enhanced feasible-path uRPF technique (presented in Section 3.4). Implementation and operations considerations are discussed in Section 3.6.

Throughout this document, the routes under consideration are assumed to have been vetted based on prefix filtering [RFC7454] and possibly origin validation [RFC6811].

The EFP-uRPF methods aim to significantly reduce false positives regarding invalid detection in SAV. They are expected to add greater operational robustness and efficacy to uRPF while minimizing ISPs' concerns about accidental service disruption for their customers. It is expected that this will encourage more deployment of uRPF to help realize its Denial of Service (DoS) and Distributed DoS (DDoS) prevention benefits network wide.

1.1. Terminology

The Reverse Path Forwarding (RPF) list is the list of permissible source-address prefixes for incoming data packets on a given interface.

Peering relationships considered in this document are provider-to-customer (P2C), customer-to-provider (C2P), and peer-to-peer (P2P). Here, "provider" refers to a transit provider. The first two are transit relationships. A peer connected via a P2P link is known as a lateral peer (non-transit).

AS A's customer cone is A plus all the ASes that can be reached from A following only P2C links [Luckie].

A stub AS is an AS that does not have any customers or lateral peers. In this document, a single-homed stub AS is one that has a single transit provider and a multihomed stub AS is one that has multiple (two or more) transit providers.

1.2. Requirements Language

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "NOT RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in BCP 14 [RFC2119] [RFC8174] when, and only when, they appear in all capitals, as shown here.

2. Review of Existing Source Address Validation Techniques

There are various existing techniques for the mitigation of DoS/DDoS attacks with spoofed addresses [RFC2827] [RFC3704]. SAV is performed in network edge devices, such as border routers, Cable Modem Termination Systems (CMTS) [RFC4036], and Packet Data Network Gateways (PDN-GWs) in mobile networks [Firmin]. Ingress Access Control List (ACL) and uRPF are techniques employed for implementing SAV [RFC2827] [RFC3704] [ISOC].

2.1. SAV Using Access Control List

Ingress/egress ACLs are maintained to list acceptable (or alternatively, unacceptable) prefixes for the source addresses in the incoming/outgoing Internet Protocol (IP) packets. Any packet with a source address that fails the filtering criteria is dropped. The ACLs for the ingress/egress filters need to be maintained to keep them up to date. Updating the ACLs is an operator-driven manual process; hence, it is operationally difficult or infeasible.

Typically, the egress ACLs in access aggregation devices (e.g., CMTS, PDN-GW) permit source addresses only from the address spaces (prefixes) that are associated with the interface on which the customer network is connected. Ingress ACLs are typically deployed on border routers and drop ingress packets when the source address is spoofed (e.g., belongs to obviously disallowed prefix blocks, IANA special-purpose prefixes [SPAR-v4][SPAR-v6], provider's own prefixes, etc.).

2.2. SAV Using Strict Unicast Reverse Path Forwarding

Note: In the figures (scenarios) in this section and the subsequent sections, the following terminology is used:

* "fails" means drops packets with legitimate source addresses.

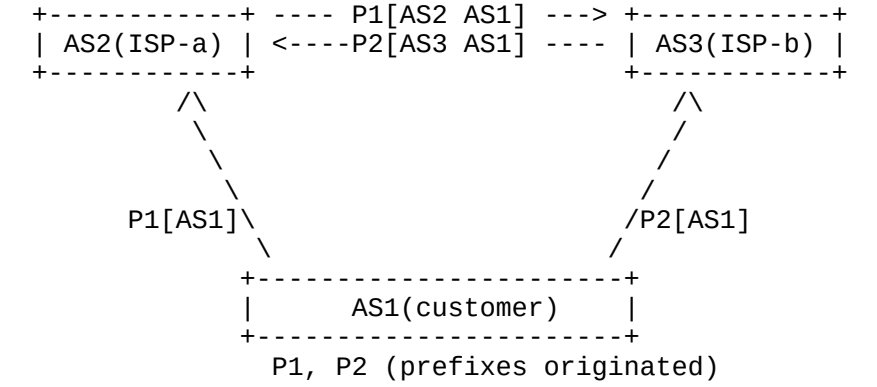
* "works (but not desirable)" means passes all packets with

legitimate source addresses but is oblivious to directionality.

* "works best" means passes all packets with legitimate source addresses with no (or minimal) compromise of directionality.

* The notation Pi[ASn ASm ...] denotes a BGP update with prefix Pi and an AS_PATH as shown in the square brackets.

In the strict uRPF method, an ingress packet at a border router is accepted only if the Forwarding Information Base (FIB) contains a prefix that encompasses the source address and forwarding information for that prefix points back to the interface over which the packet was received. In other words, the reverse path for routing to the source address (if it were used as a destination address) should use the same interface over which the packet was received. It is well known that this method has limitations when networks are multihomed, routes are not symmetrically announced to all transit providers, and there is asymmetric routing of data packets. Asymmetric routing occurs (see Figure 1) when a customer AS announces one prefix (P1) to one transit provider (ISP-a) and a different prefix (P2) to another transit provider (ISP-b) but routes data packets with source addresses in the second prefix (P2) to the first transit provider (ISP-a) or vice versa. Then, data packets with a source address in prefix P2 that are received at AS2 directly from AS1 will get dropped. Further, data packets with a source address in prefix P1 that originate from AS1 and traverse via AS3 to AS2 will also get dropped at AS2.



Consider data packets received at AS2
(1) from AS1 with a source address (SA) in P2, or
(2) from AS3 that originated from AS1 with a SA in P1:
* Strict uRPF fails
* Feasible-path uRPF fails
* Loose uRPF works (but not desirable)
* Enhanced feasible-path uRPF works best

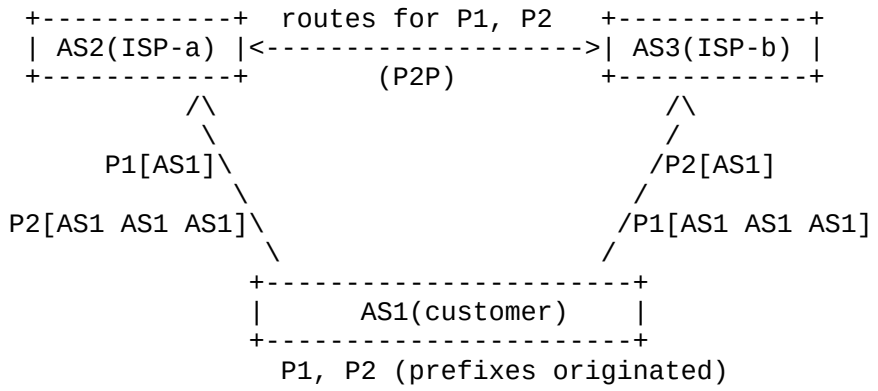
Figure 1: Scenario 1 for Illustration of Efficacy of uRPF Schemes

2.3. SAV Using Feasible-Path Unicast Reverse Path Forwarding

The feasible-path uRPF technique helps partially overcome the problem identified with the strict uRPF in the multihoming case. The feasible-path uRPF is similar to the strict uRPF, but in addition to inserting the best-path prefix, additional prefixes from alternative announced routes are also included in the RPF list. This method relies on either (a) announcements for the same prefixes (albeit some may be prepended to effect lower preference) propagating to all transit providers performing feasible-path uRPF checks or (b) announcement of an aggregate less-specific prefix to all transit providers while announcing more-specific prefixes (covered by the less-specific prefix) to different transit providers as needed for traffic engineering.

As an example, in the multihoming scenario (see Scenario 2 in Figure 2), if the customer AS announces routes for both prefixes (P1, P2) to both transit providers (with suitable prepends if needed for traffic engineering), then the feasible-path uRPF method works. It should be mentioned that the feasible-path uRPF works in this scenario only if customer routes are preferred at AS2 and AS3 over a shorter non-customer route. However, the feasible-path uRPF method has limitations as well. One form of limitation naturally occurs when the recommendation (a) or (b) mentioned above regarding propagation of prefixes is not followed.

Another form of limitation can be described as follows. In Scenario 2 (described here, illustrated in Figure 2), it is possible that the second transit provider (ISP-b or AS3) does not propagate the prepended route for prefix P1 to the first transit provider (ISP-a or AS2). This is because AS3's decision policy permits giving priority to a shorter route to prefix P1 via a lateral peer (AS2) over a longer route learned directly from the customer (AS1). In such a scenario, AS3 would not send any route announcement for prefix P1 to AS2 (over the P2P link). Then, a data packet with a source address in prefix P1 that originates from AS1 and traverses via AS3 to AS2 will get dropped at AS2.



Consider data packets received at AS2 via AS3 that originated from AS1 and have a source address in P1:
* Feasible-path uRPF works (if the customer route to P1 is preferred at AS3 over the shorter path)
* Feasible-path uRPF fails (if the shorter path to P1 is preferred at AS3 over the customer route)
* Loose uRPF works (but not desirable)
* Enhanced feasible-path uRPF works best

Figure 2: Scenario 2 for Illustration of Efficacy of uRPF Schemes

2.4. SAV Using Loose Unicast Reverse Path Forwarding

In the loose uRPF method, an ingress packet at the border router is accepted only if the FIB has one or more prefixes that encompass the source address. That is, a packet is dropped if no route exists in the FIB for the source address. Loose uRPF sacrifices directionality. It only drops packets if the source address is unreachable in the current FIB (e.g., IANA special-purpose prefixes [SPAR-v4][SPAR-v6], unallocated, allocated but currently not routed).

2.5. SAV Using VRF Table

The Virtual Routing and Forwarding (VRF) technology [RFC4364] [Juniper] allows a router to maintain multiple routing table instances separate from the global Routing Information Base (RIB). External BGP (eBGP) peering sessions send specific routes to be

stored in a dedicated VRF table. The uRPF process queries the VRF table (instead of the FIB) for source address validation. A VRF table can be dedicated per eBGP peer and used for uRPF for only that peer, resulting in strict mode operation. For implementing loose uRPF on an interface, the corresponding VRF table would be global, i.e., contains the same routes as in the FIB.

3. SAV Using Enhanced Feasible-Path uRPF

3.1. Description of the Method

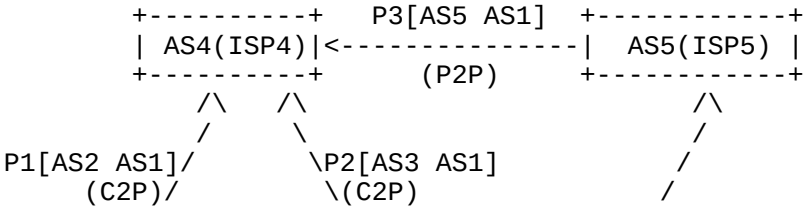
The enhanced feasible-path uRPF (EFP-uRPF) method adds greater operational robustness and efficacy to existing uRPF methods discussed in Section 2. That is because it avoids dropping legitimate data packets and compromising directionality. The method is based on the principle that if BGP updates for multiple prefixes with the same origin AS were received on different interfaces (at border routers), then incoming data packets with source addresses in any of those prefixes should be accepted on any of those interfaces. The EFP-uRPF method can be best explained with an example, as follows:

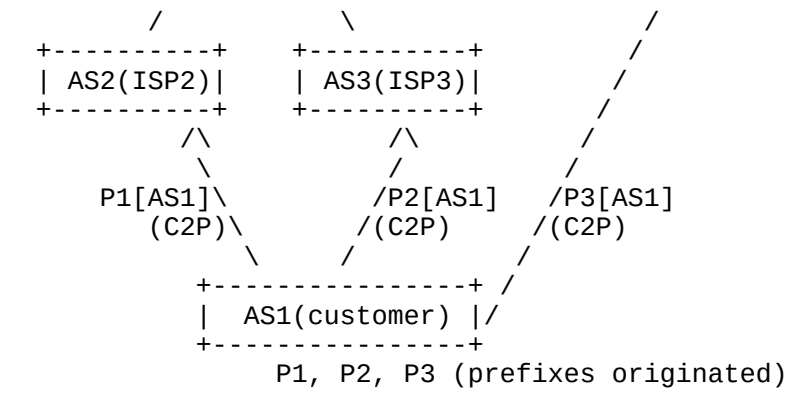
Let us say, in its Adj-RIBs-In [RFC4271], a border router of ISP-A has the set of prefixes {Q1, Q2, Q3}, each of which has AS-x as its origin and AS-x is in ISP-A's customer cone. In this set, the border router received the route for prefix Q1 over a customer-facing interface while it learned the routes for prefixes Q2 and Q3 from a lateral peer and an upstream transit provider, respectively. In this example scenario, the enhanced feasible-path uRPF method requires Q1, Q2, and Q3 be included in the RPF list for the customer interface under consideration.

Thus, the EFP-uRPF method gathers feasible paths for customer interfaces in a more precise way (as compared to the feasible-path uRPF) so that all legitimate packets are accepted while the directionality property is not compromised.

The above-described EFP-uRPF method is recommended to be applied on customer interfaces. It can also be extended to create the RPF lists for lateral peer interfaces. That is, the EFP-uRPF method can be applied (and loose uRPF avoided) on lateral peer interfaces. That will help to avoid compromising directionality for lateral peer interfaces (which is inevitable with loose uRPF; see Section 2.4).

Looking back at Scenarios 1 and 2 (Figures 1 and 2), the EFP-uRPF method works better than the other uRPF methods. Scenario 3 (Figure 3) further illustrates the enhanced feasible-path uRPF method with a more concrete example. In this scenario, the focus is on operation of the EFP-uRPF at ISP4 (AS4). ISP4 learns a route for prefix P1 via a C2P interface from customer ISP2 (AS2). This route for P1 has origin AS1. ISP4 also learns a route for P2 via another C2P interface from customer ISP3 (AS3). Additionally, AS4 learns a route for P3 via a lateral P2P interface from ISP5 (AS5). Routes for all three prefixes have the same origin AS (i.e., AS1). Using the enhanced feasible-path uRPF scheme and given the commonality of the origin AS across the routes for P1, P2, and P3, AS4 includes all of these prefixes in the RPF list for the customer interfaces (from AS2 and AS3).





Consider that data packets (sourced from AS1)
may be received at AS4 with a source address
in P1, P2, or P3 via any of the neighbors (AS2, AS3, AS5):
* Feasible-path uRPF fails
* Loose uRPF works (but not desirable)
* Enhanced feasible-path uRPF works best

Figure 3: Scenario 3 for Illustration of Efficacy of uRPF Schemes

3.1.1. Algorithm A: Enhanced Feasible-Path uRPF

The underlying algorithm in the solution method described above (Section 3.1) can be specified as follows (to be implemented in a transit AS):

1. Create the set of unique origin ASes considering only the routes in the Adj-RIBs-In of customer interfaces. Call it Set A = {AS1, AS2, ..., ASn}.

2. Considering all routes in Adj-RIBs-In for all interfaces (customer, lateral peer, and transit provider), form the set of unique prefixes that have a common origin AS1. Call it Set X1.

3. Include Set X1 in the RPF list on all customer interfaces on which one or more of the prefixes in Set X1 were received.

4. Repeat Steps 2 and 3 for each of the remaining ASes in Set A (i.e., for ASi, where i = 2, ..., n).

The above algorithm can also be extended to apply the EFP-uRPF method to lateral peer interfaces. However, it is left up to the operator to decide whether they should apply the EFP-uRPF or loose uRPF method on lateral peer interfaces. The loose uRPF method is recommended to be applied on transit provider interfaces.

3.2. Operational Recommendations

The following operational recommendations will make the operation of the enhanced feasible-path uRPF robust:

For multihomed stub AS:

* A multihomed stub AS should announce at least one of the prefixes it originates to each of its transit provider ASes. (It is understood that a single-homed stub AS would announce all prefixes it originates to its sole transit provider AS.)

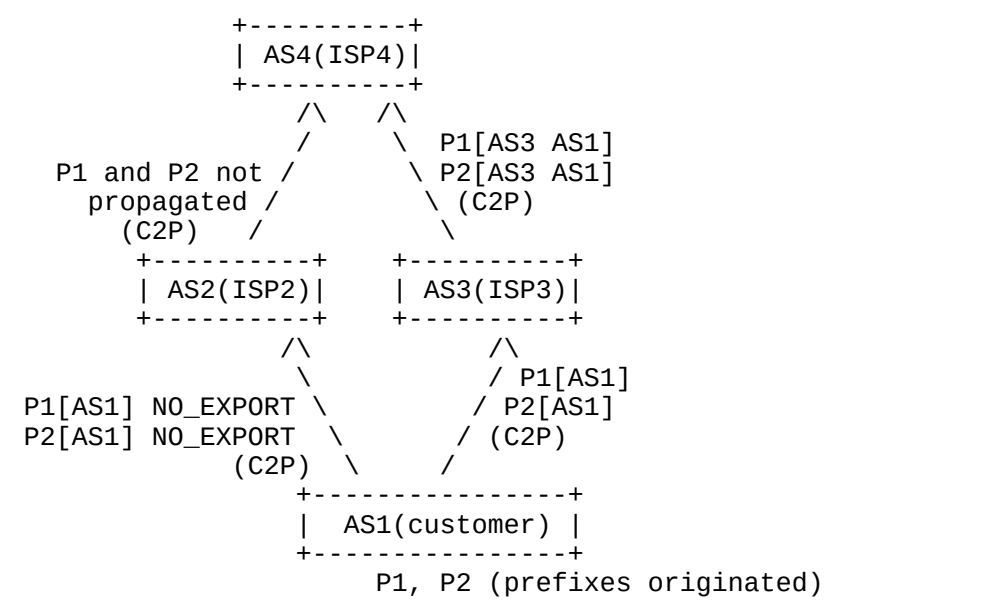
For non-stub AS:

* A non-stub AS should also announce at least one of the prefixes it originates to each of its transit provider ASes.

* Additionally, from the routes it has learned from customers, a non-stub AS SHOULD announce at least one route per origin AS to each of its transit provider ASes.

3.3. A Challenging Scenario

It should be observed that in the absence of ASes adhering to above recommendations, the following example scenario, which poses a challenge for the enhanced feasible-path uRPF (as well as for traditional feasible-path uRPF), may be constructed. In the scenario illustrated in Figure 4, since routes for neither P1 nor P2 are propagated on the AS2-AS4 interface (due to the presence of NO_EXPORT Community), the enhanced feasible-path uRPF at AS4 will reject data packets received on that interface with source addresses in P1 or P2. (For a little more complex example scenario, see slide #10 in [Sriram-URPF].)



Consider that data packets (sourced from AS1) may be received at AS4 with a source address in P1 or P2 via AS2:
* Feasible-path uRPF fails
* Loose uRPF works (but not desirable)
* Enhanced feasible-path uRPF with Algorithm A fails
* Enhanced feasible-path uRPF with Algorithm B works best

Figure 4: Illustration of a Challenging Scenario

3.4. Algorithm B: Enhanced Feasible-Path uRPF with Additional Flexibility across Customer Cone

Adding further flexibility to the enhanced feasible-path uRPF method can help address the potential limitation identified above using the scenario in Figure 4 (Section 3.3). In the following, "route" refers to a route currently existing in the Adj-RIBs-In. Including the additional degree of flexibility, the modified algorithm called Algorithm B (implemented in a transit AS) can be described as follows:

1. Create the set of all directly connected customer interfaces. Call it Set I = {I1, I2, ..., Ik}.

2. Create the set of all unique prefixes for which routes exist in Adj-RIBs-In for the interfaces in Set I. Call it Set P = {P1, P2, ..., Pm}.

3. Create the set of all unique origin ASes seen in the routes that exist in Adj-RIBs-In for the interfaces in Set I. Call it Set A = {AS1, AS2, ..., ASn}.

4. Create the set of all unique prefixes for which routes exist in Adj-RIBs-In of all lateral peer and transit provider interfaces such that each of the routes has its origin AS belonging in Set A. Call it Set Q = {Q1, Q2, ..., Qj}.

5. Then, Set Z = Union(P,Q) is the RPF list that is applied for every customer interface in Set I.

When Algorithm B (which is more flexible than Algorithm A) is employed on customer interfaces, the type of limitation identified in Figure 4 (Section 3.3) is overcome and the method works. The directionality property is minimally compromised, but the proposed EFP-uRPF method with Algorithm B is still a much better choice (for the scenario under consideration) than applying the loose uRPF method, which is oblivious to directionality.

So, applying the EFP-uRPF method with Algorithm B is recommended on customer interfaces for the challenging scenarios, such as those described in Section 3.3.

3.5. Augmenting RPF Lists with ROA and IRR Data

It is worth emphasizing that an indirect part of the proposal in this document is that RPF filters may be augmented from secondary sources. Hence, the construction of RPF lists using a method proposed in this document (Algorithm A or B) can be augmented with data from Route Origin Authorization (ROA) [RFC6482], as well as Internet Routing Registry (IRR) data. Special care should be exercised when using IRR data because it is not always accurate or trusted. In the EFP-uRPF method with Algorithm A (see Section 3.1.1), if a ROA includes prefix Pi and ASj, then augment the RPF list of each customer interface on which at least one route with origin ASj was received with prefix Pi. In the EFP-uRPF method with Algorithm B, if ASj belongs in Set A (see Step #3 Section 3.4) and if a ROA includes prefix Pi and ASj, then augment the RPF list Z in Step 5 of Algorithm B with prefix Pi. Similar procedures can be followed with reliable IRR data as well. This will help make the RPF lists more robust about source addresses that may be legitimately used by customers of the ISP.

3.6. Implementation and Operations Considerations

3.6.1. Impact on FIB Memory Size Requirement

The existing RPF checks in edge routers take advantage of existing line card implementations to perform the RPF functions. For implementation of the enhanced feasible-path uRPF, the general necessary feature would be to extend the line cards to take arbitrary RPF lists that are not necessarily the same as the existing FIB contents. In the algorithms (Sections 3.1.1 and 3.4) described here, the RPF lists are constructed by applying a set of rules to all received BGP routes (not just those selected as best path and installed in the FIB). The concept of uRPF querying an RPF list (instead of the FIB) is similar to uRPF querying a VRF table (see Section 2.5).

The techniques described in this document require that there should be additional memory (i.e., ternary content-addressable memory (TCAM)) available to store the RPF lists in line cards. For an ISP's AS, the RPF list size for each line card will roughly equal the total number of originated prefixes from ASes in its customer cone (assuming Algorithm B in Section 3.4 is used). (Note: EFP-uRPF with Algorithm A (see Section 3.1.1) requires much less memory than EFP-

uRPF with Algorithm B.)

The following table shows the measured customer cone sizes in number of prefixes originated (from all ASes in the customer cone) for various types of ISPs [Sriram-RIPE63]:

Type of ISP	Measured Customer Cone Size in # Prefixes (in turn this is an estimate for RPF list size on the line card)
Very Large Global ISP #1	32393
Very Large Global ISP #2	29528
Large Global ISP	20038
Mid-size Global ISP	8661
Regional ISP (in Asia)	1101

Table 1: Customer Cone Sizes (# Prefixes) for Various Types of ISPs

For some super large global ISPs that are at the core of the Internet, the customer cone size (# prefixes) can be as high as a few hundred thousand [CAIDA], but uRPF is most effective when deployed at ASes at the edges of the Internet where the customer cone sizes are smaller, as shown in Table 1.

A very large global ISP's router line card is likely to have a FIB size large enough to accommodate 2 million routes [Cisco1]. Similarly, the line cards in routers corresponding to a large global ISP, a midsize global ISP, and a regional ISP are likely to have FIB sizes large enough to accommodate about 1 million, 0.5 million, and 100k routes, respectively [Cisco2]. Comparing these FIB size numbers with the corresponding RPF list size numbers in Table 1, it can be surmised that the conservatively estimated RPF list size is only a small fraction of the anticipated FIB memory size under relevant ISP scenarios. What is meant here by relevant ISP scenarios is that only smaller ISPs (and possibly some midsize and regional ISPs) are expected to implement the proposed EFP-uRPF method since it is most effective closer to the edges of the Internet.

3.6.2. Coping with BGP's Transient Behavior

BGP routing announcements can exhibit transient behavior. Routes may be withdrawn temporarily and then reannounced due to transient conditions, such as BGP session reset or link failure recovery. To cope with this, hysteresis should be introduced in the maintenance of the RPF lists. Deleting entries from the RPF lists SHOULD be delayed by a predetermined amount (the value based on operational experience) when responding to route withdrawals. This should help suppress the effects due to the transients in BGP.

3.7. Summary of Recommendations

Depending on the scenario, an ISP or enterprise AS operator should follow one of the following recommendations concerning uRPF/SAV:

1. For directly connected networks, i.e., subnets directly connected to the AS, the AS under consideration SHOULD perform ACL-based SAV.

2. For a directly connected single-homed stub AS (customer), the AS under consideration SHOULD perform SAV based on the strict uRPF method.

3. For all other scenarios:

   * The EFP-uRPF method with Algorithm B (see Section 3.4) SHOULD be applied on customer interfaces.

   * The loose uRPF method SHOULD be applied on lateral peer and transit provider interfaces.

It is also recommended that prefixes from registered ROAs and IRR route objects that include ASes in an ISP's customer cone SHOULD be used to augment the pertaining RPF lists (see Section 3.5 for details).

3.7.1. Applicability of the EFP-uRPF Method with Algorithm A

The EFP-uRPF method with Algorithm A is not mentioned in the above set of recommendations. It is an alternative to EFP-uRPF with Algorithm B and can be used in limited circumstances. The EFP-uRPF method with Algorithm A is expected to work fine if an ISP deploying it has only multihomed stub customers. It is trivially equivalent to strict uRPF if an ISP deploys it for a single-homed stub customer. More generally, it is also expected to work fine when there is absence of limitations, such as those described in Section 3.3. However, caution is required for use of EFP-uRPF with Algorithm A because even if the limitations are not expected at the time of deployment, the vulnerability to change in conditions exists. It may be difficult for an ISP to know or track the extent of use of NO_EXPORT (see Section 3.3) on routes within its customer cone. If an ISP decides to use EFP-uRPF with Algorithm A, it should make its direct customers aware of the operational recommendations in Section 3.2. This means that the ISP notifies direct customers that at least one prefix originated by each AS in the direct customer's customer cone must propagate to the ISP.

On a lateral peer interface, an ISP may choose to apply the EFP-uRPF method with Algorithm A (with appropriate modification of the algorithm). This is because stricter forms of uRPF (than the loose uRPF) may be considered applicable by some ISPs on interfaces with lateral peers.

4. Security Considerations

The security considerations in BCP 38 [RFC2827] and RFC 3704 [RFC3704] apply for this document as well. In addition, if considering using the EFP-uRPF method with Algorithm A, an ISP or AS operator should be aware of the applicability considerations and potential vulnerabilities discussed in Section 3.7.1.

In augmenting RPF lists with ROA (and possibly reliable IRR) information (see Section 3.5), a trade-off is made in favor of reducing false positives (regarding invalid detection in SAV) at the expense of another slight risk. The other risk being that a malicious actor at another AS in the neighborhood within the customer cone might take advantage (of the augmented prefix) to some extent. This risk also exists even with normal announced prefixes (i.e.,

without ROA augmentation) for any uRPF method other than the strict uRPF. However, the risk is mitigated if the transit provider of the other AS in question is performing SAV.

Though not within the scope of this document, security hardening of routers and other supporting systems (e.g., Resource PKI (RPKI) and ROA management systems) against compromise is extremely important. The compromise of those systems can affect the operation and performance of the SAV methods described in this document.

5. IANA Considerations

This document has no IANA actions.

6. References

6.1. Normative References

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997, <https://www.rfc-editor.org/info/rfc2119>.

[RFC2827] Ferguson, P. and D. Senie, "Network Ingress Filtering: Defeating Denial of Service Attacks which employ IP Source Address Spoofing", BCP 38, RFC 2827, DOI 10.17487/RFC2827, May 2000, <https://www.rfc-editor.org/info/rfc2827>.

[RFC3704] Baker, F. and P. Savola, "Ingress Filtering for Multihomed Networks", BCP 84, RFC 3704, DOI 10.17487/RFC3704, March 2004, <https://www.rfc-editor.org/info/rfc3704>.

[RFC4271] Rekhter, Y., Ed., Li, T., Ed., and S. Hares, Ed., "A Border Gateway Protocol 4 (BGP-4)", RFC 4271, DOI 10.17487/RFC4271, January 2006, <https://www.rfc-editor.org/info/rfc4271>.

[RFC8174] Leiba, B., "Ambiguity of Uppercase vs Lowercase in RFC 2119 Key Words", BCP 14, RFC 8174, DOI 10.17487/RFC8174, May 2017, <https://www.rfc-editor.org/info/rfc8174>.

6.2. Informative References

[CAIDA] CAIDA, "Information for AS 174 (COGENT-174)", October 2019, <https://spoofer.caida.org/as.php?asn=174>.

[Cisco1] Cisco, "Internet Routing Table Growth Causes %ROUTING-FIB-4-RSRC_LOW Message on Trident-Based Line Cards", January 2014, <https://www.cisco.com/c/en/us/support/docs/routers/asr-9000-series-aggregation-services-routers/116999-problem-line-card-00.html>.

[Cisco2] Cisco, "Cisco Nexus 7000 Series NX-OS Unicast Routing Configuration Guide, Release 5.x (Chapter 15: 'Managing the Unicast RIB and FIB')", March 2018, <https://www.cisco.com/c/en/us/td/docs/switches/datacenter/sw/5_x/nx-os/unicast/configuration/guide/l3_cli_nxos/l3_NewChange.html>.

[Firmin] Firmin, F., "The Evolved Packet Core", <https://www.3gpp.org/technologies/keywords-acronyms/100-the-evolved-packet-core>.

[ISOC] Internet Society, "Addressing the challenge of IP spoofing", September 2015,

<https://www.internetsociety.org/resources/doc/2015/addressing-the-challenge-of-ip-spoofing/>.

[Juniper] Juniper Networks, "Creating Unique VPN Routes Using VRF Tables", May 2019, <https://www.juniper.net/documentation/en_US/junos/topics/topic-map/l3-vpns-routes-vrf-tables.html#id-understanding-virtual-routing-and-forwarding-tables>.

[Luckie] Luckie, M., Huffaker, B., Dhamdhere, A., Giotsas, V., and kc. claffy, "AS Relationships, customer cones, and validation", In Proceedings of the 2013 Internet Measurement Conference, DOI 10.1145/2504730.2504735, October 2013, <https://dl.acm.org/doi/10.1145/2504730.2504735>.

[RFC4036] Sawyer, W., "Management Information Base for Data Over Cable Service Interface Specification (DOCSIS) Cable Modem Termination Systems for Subscriber Management", RFC 4036, DOI 10.17487/RFC4036, April 2005, <https://www.rfc-editor.org/info/rfc4036>.

[RFC4364] Rosen, E. and Y. Rekhter, "BGP/MPLS IP Virtual Private Networks (VPNs)", RFC 4364, DOI 10.17487/RFC4364, February 2006, <https://www.rfc-editor.org/info/rfc4364>.

[RFC6482] Lepinski, M., Kent, S., and D. Kong, "A Profile for Route Origin Authorizations (ROAs)", RFC 6482, DOI 10.17487/RFC6482, February 2012, <https://www.rfc-editor.org/info/rfc6482>.

[RFC6811] Mohapatra, P., Scudder, J., Ward, D., Bush, R., and R. Austein, "BGP Prefix Origin Validation", RFC 6811, DOI 10.17487/RFC6811, January 2013, <https://www.rfc-editor.org/info/rfc6811>.

[RFC7454] Durand, J., Pepelnjak, I., and G. Doering, "BGP Operations and Security", BCP 194, RFC 7454, DOI 10.17487/RFC7454, February 2015, <https://www.rfc-editor.org/info/rfc7454>.

[SPAR-v4] IANA, "IANA IPv4 Special-Purpose Address Registry", <https://www.iana.org/assignments/iana-ipv4-special-registry/>.

[SPAR-v6] IANA, "IANA IPv6 Special-Purpose Address Registry", <https://www.iana.org/assignments/iana-ipv6-special-registry/>.

[Sriram-RIPE63] Sriram, K. and R. Bush, "Estimating CPU Cost of BGPSEC on a Router", Presented at RIPE 63 and at the SIDR WG meeting at IETF 83, March 2012, <http://www.ietf.org/proceedings/83/slides/slides-83-sidr-7.pdf>.

[Sriram-URPF] Sriram, K., Montgomery, D., and J. Haas, "Enhanced Feasible-Path Unicast Reverse Path Filtering", Presented at the OPSEC WG meeting at IETF 101, March 2018, <https://datatracker.ietf.org/meeting/101/materials/slides-101-opsec-draft-sriram-opsec-urpf-improvements-00>.

Acknowledgements

The authors would like to thank Sandy Murphy, Alvaro Retana, Job Snijders, Marco Marzetti, Marco d’Itri, Nick Hilliard, Gert Doering,

Fred Baker, Igor Gashinsky, Igor Lubashev, Andrei Robachevsky, Barry Greene, Amir Herzberg, Ruediger Volk, Jared Mauch, Oliver Borchert, Mehmet Adalier, and Joel Jaeggli for comments and suggestions. The comments and suggestions received from the IESG reviewers are also much appreciated.

Authors' Addresses

Kotikalapudi Sriram
USA National Institute of Standards and Technology
100 Bureau Drive
Gaithersburg, MD 20899
United States of America

Email: ksriram@nist.gov

Doug Montgomery
USA National Institute of Standards and Technology
100 Bureau Drive
Gaithersburg, MD 20899
United States of America

Email: dougm@nist.gov

Jeffrey Haas
Juniper Networks, Inc.
1133 Innovation Way
Sunnyvale, CA 94089
United States of America

Email: jhaas@juniper.net