Enhanced Feasible-Path Unicast Reverse Path Filtering

draft-sriram-opsec-urpf-improvements-03

K. Sriram (NIST), D. Montgomery (NIST), and J. Haas (Juniper)

OPSEC WG Meeting, IETF 101, London
March 2018

Acknowledgements: The authors are grateful to many folks who offered feedback and suggestions in the OPSEC WG meeting at IETF-99 and earlier on the GROW mailing list.
Difficulties with Adoption of uRPF Solutions

- Strict uRPF is usable in very limited scenarios
- Loose uRPF is not very effective for denying traffic with IPv4 address spoofing (except bogons, etc.)
- Feasible path uRPF is a refinement but ISPs apprehensive that they might deny traffic with legitimate customer source IP addresses
  - When faced with multi-homing and asymmetric routing
- Is there a way to make feasible-path more generalized and accurate?
- Goal: Encourage wider deployment of uRPF
Reverse Path Filter (RPF) List

The list of permissible prefixes for source address validation on ingress data packets on a given interface.
Enhanced Feasible Path uRPF
Algorithm A

Algorithm for customer facing ISP eBGP router:

1. Set $A = \{AS1, AS2, \ldots, ASn\}$ is the list of all unique origin ASes in Adj-RIB-Ins on customer interfaces

2. Set $X_1$ is the list of unique prefixes in *all* Adj-RIB-Ins routes that have a common origin AS1.

3. Include $X_1$ in RPF list on all customer interfaces on which one or more of the prefixes in set $X_1$ were received

4. Repeat Steps 2 and 3 for all ASes in set $A$

(Apply Loose uRPF on lateral peer and transit-provider interfaces.)
Basic Scenario A

Consider data packets received on customer interfaces at AS2 with source address in P1 or P2:

- **X** Strict uRPF fails
- **X** Feasible-path uRPF fails (since routes for P1, P2 are selectively announced to different upstream ISPs)
- ✔ Loose uRPF works (but not desirable)
- ✔ Enhanced Feasible-path uRPF works best
Consider data packets received on customer interfaces at AS2 with source address in P1 or P2:

- Feasible-path uRPF works
- Loose uRPF works (but not desirable)
- Enhanced Feasible-path uRPF works best
Consider that data packets (sourced from AS1) may be received on customer interfaces at AS4 with source address in P1, P2 or P3:

- X Feasible-Path uRPF fails
- ✓ Loose uRPF works (but not desirable)
- ✓ Enhanced Feasible-Path uRPF works best
Scenario 2: Example of a Challenging Scenario (from OPSEC & GROW WG discussions)
Adding More Flexibility to Enhanced Feasible Path uRPF Algorithm B (meets with the challenge)

- Let $I = \{I_1, I_2, ..., I_n\}$ represent the set of all directly-connected customer interfaces at customer-facing edge routers in a transit provider's AS.
- Let $P = \{P_1, P_2, ..., P_m\}$ represent the set of all unique prefixes for which routes were received over the interfaces in Set $I$.
- Let $A = \{AS_1, AS_2, ..., AS_k\}$ represent the set of all unique origin ASes seen in the routes that were received over the interfaces in Set $I$.
- Let $Q = \{Q_1, Q_2, ..., Q_j\}$ represent the set of all unique prefixes for which routes were received over peer or provider interfaces such that each of the routes has its origin AS belonging in Set $A$.
- Then, $Z = \text{Union}\{P, Q\}$ is the RPF list for each of the interfaces in Set $I$.

(Apply Loose uRPF on lateral peer and transit-provider interfaces.)
Scenario 3: Example of a Challenging / Complex Scenario (Algorithm B works)

P1 and P2 NOT PROPAGATED

P1 [AS1] NO_EXPORT
P2 [AS1] NO_EXPORT
Customer Cone Size (# Prefixes) = RPF List Size (worst case; Algorithm B)

<table>
<thead>
<tr>
<th>Type of ISP</th>
<th>Measured Customer Cone Size in # Prefixes (in turn this is an estimate for RPF list size on line card)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Very Large Global ISP</td>
<td>32392</td>
</tr>
<tr>
<td>Very Large Global ISP</td>
<td>29528</td>
</tr>
<tr>
<td>Large Global ISP</td>
<td>20038</td>
</tr>
<tr>
<td>Mid-size Global ISP</td>
<td>8661</td>
</tr>
<tr>
<td>Regional ISP (in Asia)</td>
<td>1101</td>
</tr>
</tbody>
</table>

References:
Available FIB Sizes in Router Line Cards

<table>
<thead>
<tr>
<th>Type of ISP</th>
<th>Guesstimated Line Card FIB Memory Size (#prefixes)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Very Large Global ISP</td>
<td>2M to 6M</td>
</tr>
<tr>
<td>Large Global ISP</td>
<td>1M</td>
</tr>
<tr>
<td>Mid-size Global ISP</td>
<td>0.5M</td>
</tr>
<tr>
<td>Regional ISP (in Asia)</td>
<td>100K</td>
</tr>
</tbody>
</table>

- RPF list sizes (slide 11) seem very small compared to the corresponding Line Card FIB sizes – correct?

Summary of BCP 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 and not multi-homed, the AS in consideration SHOULD perform ACL-based SAV.

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

3. For all other scenarios:
   * If the scenario does not involve complexity such as NO_EXPORT of routes (see Section 3.3, Figure 4), then the enhanced feasible-path uRPF method in Algorithm A (see Section 3.1.1) SHOULD be applied.
   * Else, if the scenario involves the aforementioned complexity, then the enhanced feasible-path uRPF method in Algorithm B (see Section 3.4) SHOULD be applied.