Unicast Reverse Path Forwarding

From the inside out:
Campus subnets to regional networks

Andrew Gallo
uRPF - Overview

• Problem statement
• Options for the campus network
  • Campus subnet routing
  • At the campus border
• Implementation for a small regional
  • Options aren’t great, but something is better than nothing
uRPF - Terminology

• uRPF, SAV
• BCP 38
  • RFC 2827 – May 2000
• Modes
  • Strict
  • Loose
  • Feasible (+ enhanced feasible)
• Assignment, ”used by”
  • Even if you have PI space (allocated or assigned), you likely have addresses outside these blocks in use
uRPF – A Plea

• This presentation may be review for many of you
  • Great- please share your experience- tell us what has worked, what hasn’t

• If this is new, if you have questions:

  **PLEASE, PLEASE, PLEASE, ASK!**

• Like other routing security topics, **your** effort improves **my** security

• Let me (us, the I2 community) help you help
uRPF – The Problem

• *Why are you sending me garbage?*

• Why would a host originate a packet with a source other than it’s own?
  • (ignore multihome stations for now) (maybe?? Maybe talk about this now)
  • Special case- it doesn’t have an address (yet)
    • DHCP/BOOTP – source address could be
      • Discover/Request
  • Mobile IP
    • Traffic to client is tunneled. Traffic from client is not (unless reverse tunnel used)
    • Anyone using this?

• Other cases
  • Link Local / Self Assigned
    • Shouldn’t leave local network anyway

• If you’re going to emit a packet, and expect a response (and even if you don’t) –
  Be a good neighbor and tell us who you are
uRPF – Think Global (and Local*), but *act* Local

- The ideal place to enforce source address validation is as close to the client as possible
- Even at the layer-2 port:
  - Cisco IP Source guard: ‘ip verify source vlan dhcp-snooping’
  - Ideal, yes, but let’s make our lives a little easier

* Remember – uRPF can help protect your own network from internally spoofed traffic
Configuration at a Layer 3

• First Hop layer 3 interface is a logical place to enforce source address validation
Configuration at a Layer 3

- Default configuration: accept packets from 192.168.56.22 – but why?
Configuration at a Layer 3 – strict mode

- Cisco IOS command
  - `ip verify unicast source reachable-via rx`

- Router checks incoming packet SA against route table:
  - Is this interface *the best* way to get back to SA?
  - If yes, accept
  - If no, drop

- Good option for “end-user” or “single-homed” networks
Configuration at a Layer 3 – strict mode

**Gotcha**

- Experience with early versions of NX-OS, strict mode broke DHCP (50% of the time)
- Work around – use loose mode
- Newer versions of code might have fixed this
- Network evolution moved end-user subnets to single control plane boxes (single routers or VSS)
Configuration at a Layer 3 – loose mode

• Cisco IOS command
  • `ip verify unicast source reachable-via any`

• Router checks incoming packet SA against route table:
  • Do I have any route back to the SA
    • even if it isn’t the best – it just has to exist in the route table
  • If yes, accept
  • If no, drop
Configuration at a Layer 3 – loose mode

• Consider-
  • what if I have a default route in my table (0.0.0.0/0)?
    • It matches all packets, so what’s the point of checking?
    • Junos – it depends on the platform
    • IOS – by default, will ignore default route of loose mode RPF checking
  • Static route to Null0 (IOS) or discard (Junos)
    • Drop if SA matches a route that points to these interfaces
    • Knobs to modify behavior
    • Juniper – platform/software specifics
  • Many caveats with loose mode – which is why strict mode as close to the hosts as possible is ideal
Our design

- We have strict mode enabled at end user subnets (except when routed by NX-OS)
- We do not have any RFP checking on backbone links
- We have static ACLs (firewall filters) at the border of our network:
  - For packets leaving, the SA must be something that can return to us
  - For packets arriving, the DA must be for something ‘assigned’ to us
Tools

- CAIDA’s Spoofer:
  - https://www.caida.org/projects/spoofer/
uRPF For Transit Networks – It’s Complicated

• It’s the Internet – Asymmetry abounds.
• The best we can hope for is loose mode – does the source address exist anywhere on the internet?
  • How much value does this provide?
uRPF For Transit Networks – It’s Complicated

• Things aren’t too bad if all customers are non-transit

• In this topology, uRPF strict mode will drop traffic sent by AS11039, originated by interfaces address from other providers

Route Policy:
- Internet2: LP500
- Peering: LP200
- TRCPS: LP150
- Commercial Transit: (default LP)

BGP advertise:
- 128.164.0.0/16
- 64.124.161.16/30
- 64.124.161.17/30
- 64.124.161.18/30
- 64.124.161.19/30

All EBGP links addressed from provider space
uRPF For Transit Networks – It’s Complicated

- In Junos, use “fail-filter”
  - How to generate and maintain??
  - Look at BGP, whois, IRR, fail-filter with log, ask customers about infrastructure addresses
  - Continuously monitor fail-log
- What did we see when we turned uRPF on?
  - NAT
  - High number of failures swamped logging service
  - One customer - only certain prefixes advertised to AS4901
- This is much more complicated farther upstream
Summary

• Source Address Validation is critical to minimizing the impact of spoofed traffic
• It is best done as close to the source as possible
• Your efforts protect other networks
  • *Internet hygiene is a shared responsibility*