Never via route servers

Johannes Moos

DE-CIX

jmo@de-cix.net

RIPE 80

Motivation: Route Leak Detection

Improve the detection of route leaks![1]

Example BGP AS Path
64514 64513 65501 64512

- AS65501 is classified as Tier 1
- AS64512 (originator of prefix) is downstream of AS65501
- AS64514 is our eBGP neighbor

- Does AS64513 really provide upstream/transit to AS65501?
- Unlikely!

---

[1]Scope: IXPs

Improving Prefix Filters

Example BGP AS path

64514 64513 65501 64512

- What about RPKI/IRRDB filters?
    - RPKI: Origin AS64512 and maxLength valid: ok
    - IRR: If AS64513 includes AS-SET of AS65501[2]: ok
    - IRR: AS64512 might also be a direct member of AS64514's AS-SET: ok
- Some more intelligence if this announcement is valid is needed

- **Goal:** Spot networks that should not show up in the BGP AS path[3]

---

[2]ASPA/AS-Cones to the rescue
[3]except on the leftmost position

Improving Prefix Filters

- Let's compile a list and filter based on that!
- Networks themselves know their peering relations best
- Let them indicate on their own
- Where? Peering DB

Protocols Supported	☑ Unicast IPv4 ☐ Multicast ☑ IPv6 ☐ Never via route servers

Participate!

- 42 networks[4] already participate, including:
  - Cogent Communications, Inc.
  - Vodafone Global Network
  - Telia Carrier
  - NTT Global IP Network
  - Deutsche Telekom
  - PCCW Global
  - Liberty Global

- If you are a Tier 1, check the box!

---

[4]as of 2020-05-09

More Information

📄 Rene Hernandez (2020)

Filtering with 'Never Via Route Server' on PeeringDB

https://www.mdcdatacenters.com/interconnection-mdc/
avoid-bgp-route-leaks-with-never-via-route-server-on-peeringdb/

📄 Various contributors (2018 - 2020)

PeeringDB Feature Request (GitHub)

https://github.com/peeringdb/peeringdb/issues/394