Secure Routing with RPKI: Status, Challenges and the Smart-Validator

Amir Herzberg
Univ of Connecticut, Bar Ilan Univ, Fraunhofer SIT

Joint project with
Tomas Hlavacek, Yafim Kazak, Rafi Peretz and Haya Shulman
Route-Hijacking: Real-Life Example

- Many proposed/deployed defenses, over many years...
- Challenge & focus: deployable yet effective defenses
Prefix Hijacking: prefer shorter route

1.2.0.0/16
Route: 22-333

1.2.0.0/16
Route: 333

1.2.0.0/16
Route: 666

Inter-domain link
BGP announcement
Data flow to 1.2.0.0/16
**Subprefix Hijacking:**
always prefer longest matching prefix

![Diagram showing subprefix hijacking]

- 1.2.0.0/16 (Route: 333)
- 1.2.3.0/24 (Route: 6-666)

Inter-domain link

BGP announcement

Data flow to 1.2.3.0/24
Idea: prevent hijacks using Route Origin Validation (ROV)

Domain 1 uses the (longer but correct) route 22-333, since only domain 333 is authorized origin for prefix 1.2.0.0/16
How to do Route Origin Validation (ROV)??
How to do Route Origin Validation (ROV)?

Naïvely: keep a list of valid (authorized) origin ASes for each prefix

Online check: consult DBs, e.g., Internet Routing Registries (IRRs)

Offline: digitally-signed Route Origin Authorization (ROA)
Route Origin Validation (ROV) prevents Prefix and Subprefix Hijacks

Domain 1 uses the (longer but correct) route 22-333, since only domain 333 is authorized origin for prefix 1.2.0.0/16
**RPKI: Resource Public Key Infrastructure**

- IETF standard [RFC 6480];
  main goal: prevent (sub)prefix hijacks (false origin domain)
- Allows signing Route Origin Authorizations (ROAs):
  
  **Prefix**: 1.2.0.0/16
  **Origin**: 333
  **Max-length**: 20

- Facilitates **Route Origin Validation (ROV)**:
  - Drop BGP announcements where origin AS conflicts with ROA
  - I.e.: Origin AS is **not** 333
  - **Or**: more specific than /20
RPKI Deployment: Agenda

• RPKI: What and Why  [done]

• **State of Deployment**
  • ROA adoption: trends
  • Wrong ROA: causes and damages
  • ROV adoption status, challenges
  • Impact of partial ROV adoption

• Improving deployment: The Smart Validator
  • Phase I
  • Demo
  • Phase II

• Conclusions
ROA Adoption History

Drop BGP announcements → lose (good?) traffic…
So, how many domains do Route Origin Validation?

- Announced without ROA: 647,192 (93%)
- Valid ROAs: 43,796 (6.3%)
- Wrong ROAs: 5,015 (0.7%)

About 10% wrong ROAs!! Consistently!!
Wrong ROAs??

- Requires **both** authorizations (ROAs) and validation (ROV)
- Risk: ROV with **Wrong ROA** → drop legit-yet-invalid announcements
  - Does wrong-ROAs happen? – Typical, real-life example:

  ![Diagram](image)

  Legend:
  - Resource Certificate
  - Wrong ROA
  - Legit-yet-Invalid BGP Announcement

Examples:
- 194.2.0.0/15 Domain 3215
- 194.2.35.0/24 Domain 1272 (Danone)
- 194.2.155.0/24 Domain 8361 (Ubisoft)
- 194.3.118.0/24 Domain 34444 (Eutelsat)
Measuring Adoption of Route Origin Validation

- Challenge: no direct way to measure the adoption of ROV ➔ no published measurements
- Idea: use Route-View-project’s BGP-collectors – and wrong ROAs!
- Observation: if collector receives invalid announcement ➔ Entire route does not enforce ROV!

```
ROA: 1.2.0.0/16
Domain 333

1.2.0.0/16

1

A

B

C

D

E

F

Collector

1.2.0.0/16

Route: C-A-1

1.2.0.0/16

Route: F-E-D-2

2

1.2.0.0/16
```
Measuring Adoption of Route Origin Validation

- Challenge: no direct way to measure the adoption of ROV
  ➔ no published measurements
- Observation: if collector receives invalid announcement
  ➔ Entire route does not enforce ROV!

At least 80 of 100 largest domains do not enforce ROV! Can we measure more precisely?
Better ROV Measurements...

• Dependency on existing wrong ROAs may be misleading
• More reliable: publish correct/wrong ROAs (same origin)
• Three different controlled experiments, multiple times:
  • Use RouteView Collectors (as before)
  • Use Trace-route to RIPE atlas probes
  • Use `echo’ from servers (ICMP ping or TCP SYN/ACK)
• Experiments still ongoing
• Initial results: only handful of domains enforce ROV
  • None of the 100 largest domains (cf. <20)
• Similar results apparently from measurements by Randy Bush and others (didn’t yet see details)
• What’s the impact of partial-deployment of ROV?
Partial Adoption of ROV:
Collateral damage

- Domains **not doing** ROV might cause ROV-enforcing domains to fall victim to prefix hijacking
- **Control-Plane vs. Data-Plane Mismatch:** domain discards invalid announcement, yet data flows to attacker

---

Domain 2 advertises both valid and invalid routes

Domain 2 uses invalid route for subprefix ➔ traffic to 1.1.1.0/24 still hijacked!

Domain 3 enforces ROV: discards invalid subprefix route

ROA: 1.1.0.0/16
Origin 1

To: 1.1.1.0/24
route: 2-666

To: 1.1.0.0/16
route: 2-1
Partial Adoption of ROV: Collateral benefit

Adopters protect domains behind them by discarding invalid announcements

ROA: 1.1.0.0/16
Domain 1

Domain 3 is only offered valid routes

Drawback: less incentive to deploy (`free-riders`)
Security in Partial ROV Adoption: Simulation Framework

- Use Internet domain topology of CAIDA
- Pick victim & attacker
- Victim’s prefix has a ROA
- Pick domains doing ROV
- Find domains sending to victim vs. domains sending to attacker

Empirically-derived topology from CAIDA. Includes inferred peering links [Giotsas et al., SIGCOMM’13]
Security with Partial ROV Adoption

- Subprefix-hijack success rate for adoption by x largest domains
- Compare: 100% vs. 25% adoption by other domains
- Significant benefit - but only if almost all large domains adopt – and most other domains adopt too
- We are very far from this!
RPKI Deployment: Agenda

• RPKI: What and Why
• State of Deployment
  • ROA adoption: trends
  • Wrong ROA: causes and damages
  • ROV adoption status, challenges
  • Impact of partial ROV adoption

• **Improving deployment: The Smart Validator**
  • Phase I
  • Demo
  • Phase II
• Conclusions
Fixing ROAs and ROV deployment

• Improve deployment of ROAs
  • ROAlert.org: identify wrong ROAs
  • email alerts when sysadmin-email located: 40% fixed!
  • → Should be deployed `officially’

• Smart validator
  • Encourage, improve adoption of Route Origin Validation (ROV)
  • Free, open source; extends RIPE’s RPKI validator
  • Phase I: `easy and safe deployment’ – Do No Harm
    • Fix Conflicting-ROAs [conflicting with long-lived BGP announcements]
    • Ready, experiments beginning – join us!
  • Phase II: improved security, incentives
    • In development, will be based on new version of RIPE validator
Idea: Hijacks are Short Lived

<table>
<thead>
<tr>
<th>Possible Hijacks duration [Days] from 08-2016 -&gt; 06-2017</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 Day</td>
</tr>
<tr>
<td>Serie1</td>
</tr>
<tr>
<td>60,90%</td>
</tr>
</tbody>
</table>

➡️ Allowing long-lived (>3 weeks) BGP announcements, even if conflicting with ROA, would still catch most hijacks!
Smart-Validator: Phase I

- Easy and safe to deploy: `plug and play’
- Do No Harm

**Recommend Mode** (default):
- Observes ROAs and BGP announcements
- **Recommend** BGP announcement filters
- Operator manually applies BGP announcement filters
- `What-if’ measurements: impact of safe-deployment modes

**Safe-deployment modes**
- **Ignore mode**: ignore conflicting-ROAs
- **Extend mode**: add auto-ROAs to cancel conflicts

**Experiments**: Cisco, LinkedIn, ... **You??**

- Based on RIPE’s validator; free, open source
Smart-Validator: Architecture

Data warehouse

Dashboard

Data resources

The engine
Smart Validator Dashboard Examples

**Recommend mode**

**Extend safe-deployment mode**
Demo (link)
Smart-Validator: Phase II

- Extend phase I with new ROV features:
  - **ROV++:**
    - Prefer ROV++ compliant providers
      - When learning of attack... or always/usually
    - Reduces risk of collateral-damage
    - An incentive to deploy
  - **Path-end validation: easy, strong extension to RPKI**
    - Prevent `origin hijacking` by extending ROA to identify neighbor AS
    - SigComm16 paper shows: surprisingly effective!!
Beyond BGP: Routing Against DoS

• BGP is limited to single fixed route
  • Easier to congest – e.g., in Denial-of-Service (DoS)

• BGP isn’t congestion-sensitive
  • Route does not depend on congestion, delays, loss
  • Slow response to link failure

• IP provides only best-effort service
  • No quality guarantees (max delay, max loss rate)
  • Quality-of-Service (QoS) extensions: only within domain

• Secure Accountable Inter-domain Forwarding
  • On going project – talk to me...
Conclusions

• Routing security: fun & important research area
• RPKI improves BGP’s security... if deployed widely
• Smart-validator improves ROV:
  • Phase I: make it easy and safe to deploy
  • Phase II: improve security and incentives to deployers
• Talk with us:
  • To see demo
  • To join experiments
  • To give feedback
More questions? Thanks!

42
Security with Partial ROV Adoption

Route Origin Validation (ROV) by the top domains is necessary and sufficient for substantial security benefits from RPKI.

Comparison between two scenarios:
- ROV adopted with probability $p$ (x axis)
- Same, but also by the 100 top (largest) domains

[Diagram showing the attacker's success rate with different deployment probabilities.]
Path-End Authorization, Validation: authorized neighbors of origin

1.2.0.0/16
Route: 3-2-1

Victim

1.2.0.0/16
Origin: 1

1’s neighbors: { 2 }

666

666 is not a neighbor of 1!

1.2.0.0/16
Route: 666-1

False `link`
BGP
Data flow
Path-End fails for **Path Hijacking**

11’s neighbors: { 1 }

1.2.0.0/16
Origin: 1
Route: 2-666-1-11

1.2.0.0/16
Route: 666-1-11

Path Hijacking

**Real routes are mostly short** (avg ~3.7, important content often 1!), attacker can’t change relationship ➔ path hijacking rarely works!!
Path-end validation

- Extend RPKI to authenticate the “last hop”
Simulation results:
$\text{RPKI} \approx \text{partial-BGPsec} \ll \text{Path-End}$
Path-End Validation: Properties

- Design ➜ Easy to deploy (≅ RPKI)
- Simulations ➜ Effective (>> BGPsec, RPKI)
- Analysis ➜
  - Do no harm property:
    preserve convergence of BGP
  - Security-monotone property:
    more adoption ➜ more security
    (BGPsec does not have this property!)

Skip theorems