RPKI in practice

Sebastian Wiesinger
sebastian.wiesinger@noris.net

DE-CIX Technical Meeting June 2017

noris network

Generate ROAs

Generate ROAs for your prefixes

- RIPE NCC makes this very easy
- Available at the LIR portal `https://lirportal.ripe.net/`
- End Users: your sponsoring LIR can do this for you!

RIPE NCC RPKI Validator

Install RPKI Validator

- RPKI Validator cache made by the RIPE NCC
- `https://github.com/RIPE-NCC/rpki-validator`
- Requires Oracle JDK 8 and rsync
- Didn't try OpenJDK 8 as Oracle JDK 8 was already installed

Configure RPKI Validator

- Secure the web interface!
  - Apache Reverse Proxy with SSL
  - Firewall
  - Enable “Kiosk mode” or Apache authentication

    ```
    ui.kiosk.enable=true
    ui.kiosk.user=foo
    ui.kiosk.pass=bar
    ```
- Add ARIN TAL (now publicly available!)
  - `https://www.arin.net/resources/rpki/tal.html`
  - Save to `conf/tal/arin.tal`

Configure Juniper MX for RPKI

Overview

- Feature “Origin validation for BGP” – Available since JunOS 12.2R1
- Peering routers need TCP session with RPKI Validator
- Route Validation (RV) records cached by router (default: 1 hour)
- RV database is not automatically applied to BGP `validation-state`
- iBGP peers get validation state via community

Configure Validation Session

- Don't forget your RE firewall filter(s)

```
[edit routing-options]
    validation {
        group rpki-cache {
            session 2001:db8::f00:baa {
                port 8282;
                local-address 2001:db8::1;
            }
        }
    }
```

```
swiesinger@router> show validation session
Session                                 State   Flaps     Uptime #IPv4/IPv6 records
2001:db8::f00:baa                       Up          4 3w6d 00:29:12 60726/7004
```

Configure eBGP import policy

- Match on validation database and set validation state
- Set community for other iBGP peers

```
term valid {
    from {
        protocol bgp;
        validation-database valid;
    }
    then {
        validation-state valid;
        community add bgp-rpki-valid;
        next policy;
    }
}
```

```
term unknown {
    from {
        protocol bgp;
        validation-database unknown;
    }
    then {
        validation-state unknown;
        community add bgp-rpki-unknown;
        next policy;
    }
}
```

Configure eBGP import policy

- Invalid routes – *tag* or *reject*?

```
term invalid {
    from {
        protocol bgp;
        validation-database invalid;
    }
    then {
        validation-state invalid;
        community add bgp-rpki-invalid;
        next policy;
    }
}
```

```
term invalid {
    from {
        protocol bgp;
        validation-database invalid;
    }
    then {
        reject;
    }
}
```

Apply eBGP import policy

```
[edit protocols bgp]
group DE-CIX {
  import [ bgp-reject-garbage bgp-reject-noris-inbound bgp-reject-private-as
 ↪   bgp-peer-cleanup-in bgp-rpki-validation bgp-de-cix-in deny-everything ];
}
```

Check validation state

```
swiesinger@router> show route table inet.0 protocol bgp validation-state invalid
 ↪  terse active-path

A V Destination    P Prf    Metric 1   Metric 2  Next hop        AS path
* I 5.28.192.0/19  B 170        500        151                  200612 12880 42337 I
  invalid                                      >80.81.195.19
* I 5.28.192.0/21  B 170        500        151                  200612 12880 42337 I
  invalid                                      >80.81.195.19
* I 5.28.224.0/21  B 170        500        151                  200612 12880 42337 I
  invalid                                      >80.81.195.19
[...]
```

iBGP Import Policy

- Replicate validation state on other iBGP peers
- No validation session required

```
term valid {
    from {
        protocol bgp;
        community bgp-rpki-valid;
    }
    then {
        validation-state valid;
        next policy;
    }
}
```

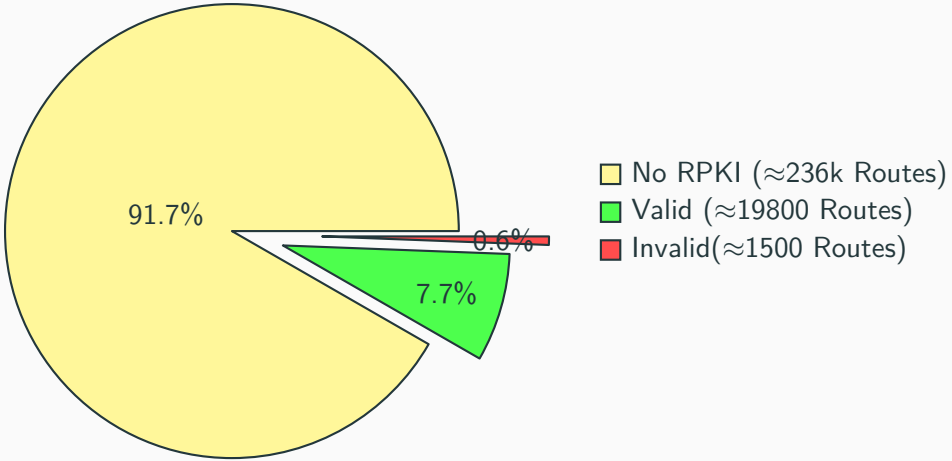
```
term unknown {
    from {
        protocol bgp;
        community bgp-rpki-unknown;
    }
    then {
        validation-state unknown;
        next policy;
    }
}
```

- If invalid routes are redistributed, set `validation-state invalid` accordingly

Statistics

RPKI Statistics at DE-CIX

- Routes received at DE-CIX (RS+direct)
- 7% of ROA covered routes invalid



Considerations

Technical considerations when using RPKI

- RPKI only validates origin of route
  - No path validation
  - Protects against misconfiguration **but not deliberate manipulation**
- External DDoS mitigation needs /24 (/48 IPv6) more specifics
- Want to validate customer BGP routes?
  - More specifics for traffic engineering
  - Blackholing (RTBH)
- Mistakes(?) are made (≈1500 at DE-CIX)
- How do you “punish” invalid routes and/or “reward” valid ones?

Somewhat political (philosophical?) considerations

- Central CA can invalidate ROAs / LIR CAs
- Currently not a useful measure because almost no one uses RPKI
- Protection against mistakes / accidental hijacks more important?
- What happens in 5 to 10 years?
- RIPE NCC under Dutch law
- What if ARIN gets “trumped”?

Questions?

Sebastian Wiesinger
sebastian.wiesinger@noris.net

noris network