Implementing RPKI and Origin-validation 101

All you wanted to know to deploy RPKI but were afraid to ask

First, what type of user are you?

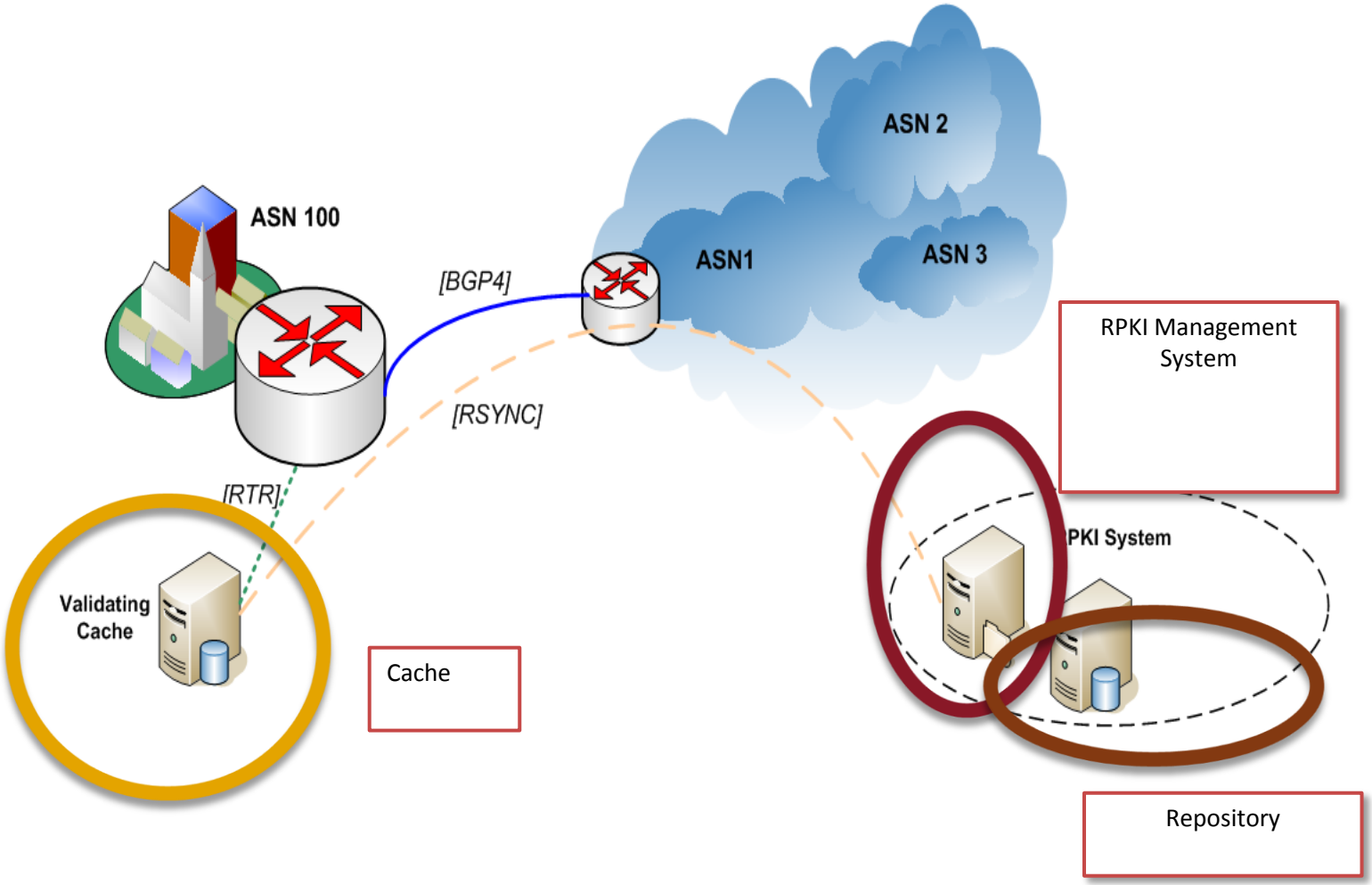
Types of users

- Prefix holder
  - You want to certify your prefixes and create ROAs
- Router operator
  - You want to validate prefixes using RPKI and origin-validation
- You are both

Prefix Holder

- You need to create and publish your resource certificate and your ROAs
  - One way is to use RIRs systems already deployed
  - Run your own CA and repository

Router Operator



Router Operator

- You need an origin-validation capable router, an RPKI cache and at least one trust anchor
- Cisco, Juniper and Quagga (srx-module) are capable routers
- RIPE NCC and ISC have cache implementations
- Each RIR is the trust anchor of the resources (IPv6 and IPv4) that they have allocated

Router Operator (2)

- Configure your cache to pull the TALs from RIRs
- Configure your router and cache to speak RTR
- Configure policies in your router
- Check your BGP routes

Thanks!

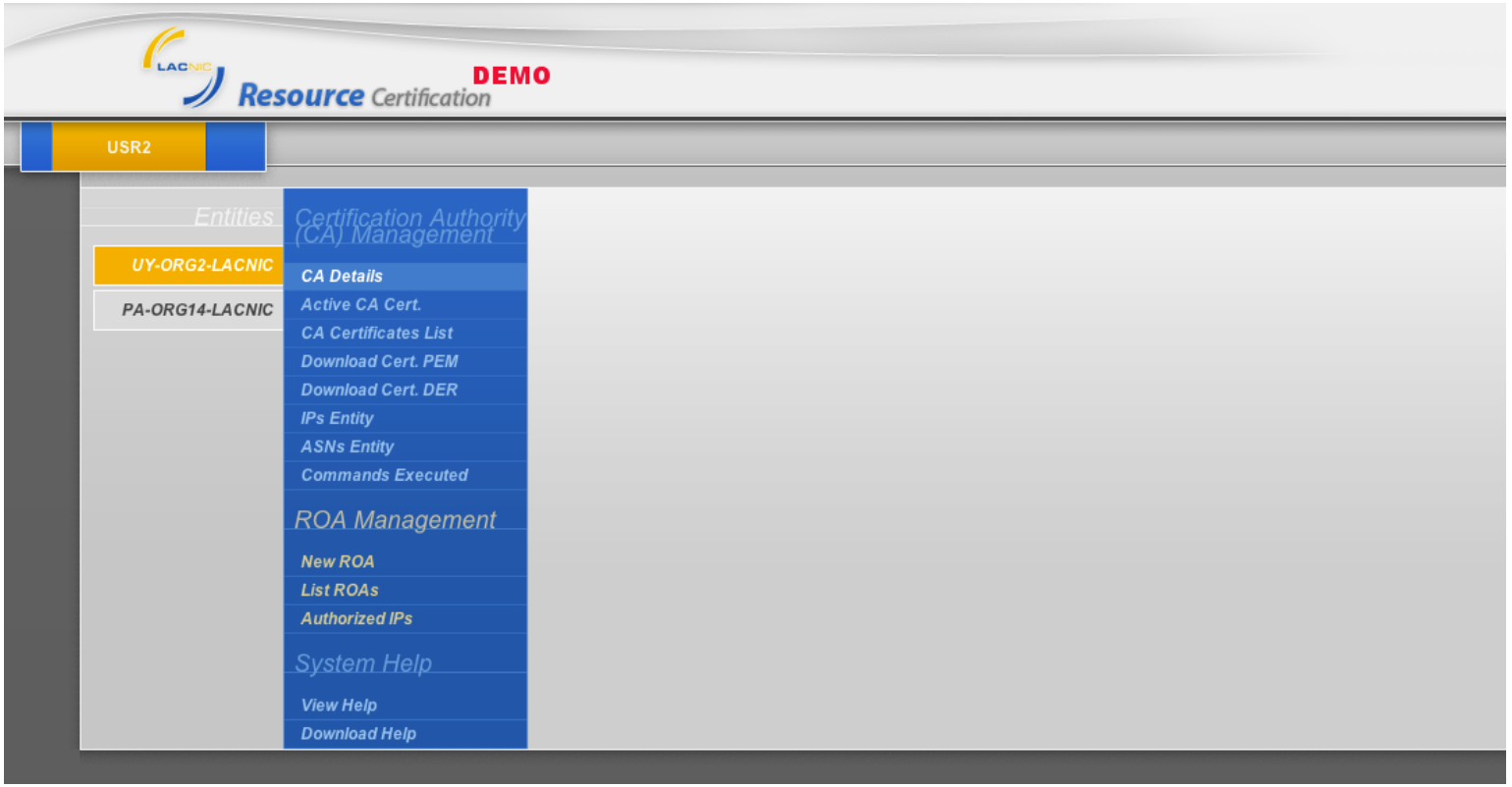
- Questions?
  - If you do, then keep reading

Where to go

```
if prefixHolder & haveQuestions:
        nextSlide()
else:
        routerOperator(slide=11)
```

Prefix Holder, the easy way

- Create certificate and ROA in your RIR system



The other way

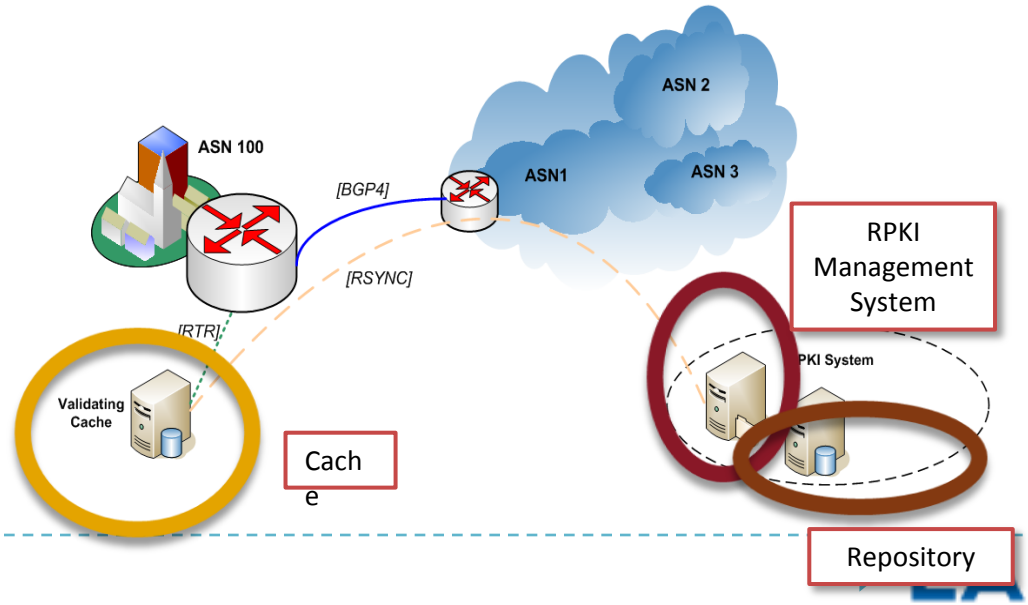
- Create your own CA
  - You can use RPKID (http://rpki.net/)
- Create Certificates, send them to your RIR using provisioning protocol
- Create ROAs

Some notes about ROAs

- Use max-prefix as close and precise as your real BGP announces
- If only a small porcion of your prefix need more specific better create multiple ROAs
- For more info, read RPKI-Based Origin Validation Operation

Router Operators

- You need:
  - A validation cache supporting RTR
  - At least a TAL for a Trust Anchor
  - A router capable of origin-validation



Validation Cache

- RIPE NCC
  - Java, runs almost anywhere, supports (RPKI routing protocol
  - Download: http://labs.ripe.net/Members/agowland/ripencc-rpki-validator.zip/view
- ISC, Rcynic
  - Runs in unix-like systems
  - Download: http://rpki.net

TALs (Trust Anchor Locator)

- This is the format to distribute Trust Anchor Material
- A TA is a self-signed certificate
- Enables selected data in the trust anchor to change, without needing to effect re-distribution of the trust anchor per se
- See: Resource Certificate PKI (RPKI) Trust Anchor Locator
- Today each RIR has its own TAL, so in practice you need 5 TALs.

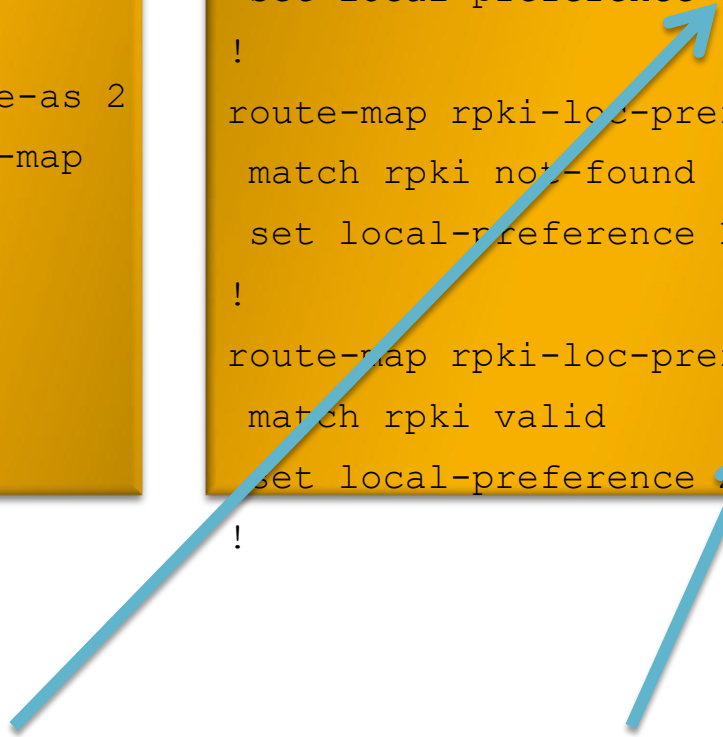
Router Configuration

- Step 1: Configure communication between cache and router
- Step 2: Enable RPKI origin validation (some platforms)
- Step 3: Apply routing policies
- Step 4: Verify and enjoy your new shining validation

Cisco

```
router bgp 1
 bgp log-neighbor-changes
 bgp rpki server tcp 192.168.1.10
port 2000 refresh 5
 network 192.168.1.0
 neighbor 192.168.255.2 remote-as 2
 neighbor 192.168.255.2 route-map
rpki-loc-pref in
```

```
!
route-map rpki-loc-pref permit 10
 match rpki invalid
 set local-preference 50
!
route-map rpki-loc-pref permit 20
 match rpki not-found
 set local-preference 100
!
route-map rpki-loc-pref permit 30
 match rpki valid
 set local-preference 200
!
```



You can deny too

Better preference

Cisco, verify

```
sh ip bgp

BGP table version is 11, local router ID is 192.168.255.5

Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale, m multipath, b backup-path, f RT-Filter,
              x best-external, a additional-path, c RIB-compressed,

Origin codes: i - IGP, e - EGP, ? - incomplete

RPKI validation codes: V valid, I invalid, N Not found

     Network          Next Hop            Metric LocPrf Weight Path
*> N10.2.4.0/24      192.168.255.2                100      0 2 4 i
*> V10.4.0.0/16      192.168.255.2                200      0 2 4 i
*> V192.168.1.0      0.0.0.0                  0         32768 i
*> V192.168.2.0      192.168.255.2            0    200      0 2 i
   I192.168.2.128/25 192.168.255.6            0     50      0 3 ?
```

Cisco, verify

- A hijack!

```
R1#sh ip bgp 192.168.2.128

BGP routing table entry for 192.168.2.128/25, version 20
Paths: (1 available, no best path)
  Not advertised to any peer
  Refresh Epoch 1
  3
    192.168.255.6 from 192.168.255.6 (192.168.2.129)
      Origin incomplete, metric 0, localpref 100, valid, external
      path 67F57464 RPKI State invalid
```

Cisco, check cache

- show ip bgp rpki server
- show ip bgp rpki table

```
R1#show ip bgp rpki table

7 BGP sovc network entries using 616 bytes of memory
7 BGP sovc record entries using 140 bytes of memory

Network            Maxlen  Origin-AS  Source  Neighbor
10.1.0.0/16        16      1          0       192.168.1.10/2000
10.2.0.0/16        16      2          0       192.168.1.10/2000
10.3.0.0/16        16      3          0       192.168.1.10/2000
10.4.0.0/16        16      4          0       192.168.1.10/2000
192.168.1.0/24     24      1          0       192.168.1.10/2000
192.168.2.0/24     24      2          0       192.168.1.10/2000
192.168.3.0/24     24      3          0       192.168.1.10/2000
```

Juniper, configure and check cache

```
routing-options {
    autonomous-system 10;
    validation {

        group cache1 {
            session 10.1.1.6 {
                refresh-time 120;
                hold-time 180;
                port 8282;
                local-address 10.1.1.5;
            }
        }
    }
}
```

```
show validation database

RV database for instance master

Prefix      Origin-AS Session State
10.0.0.0/16-19    20 12.1.1.6 valid
10.0.0.0/19-24     2 12.1.1.6 valid

  IPv4 records: 2
  IPv6 records: 0
```

Juniper BGP and Policies

```
protocols {
    bgp {
        group ASN200 {
            import rv;
            export p;
            peer-as 200;
            neighbor 10.1.1.2;
        }
    }
}

policy-options {
    policy-statement p {
        from protocol direct;
        then accept;
    }
    policy-statement rv {
        term a {
            from {
                protocol bgp;
                validation-state valid;
            }
            then {
                local-preference 110;
                validation-state valid;
                accept;
            }
        }
        term b {
            from {
                protocol bgp;
                validation-state invalid;
            }
            then {
                local-preference 9;
                validation-state invalid;
                accept;
            }
        }
        term c {
            from {
                protocol bgp;
                validation-state unknown;
            }
            then {
                validation-state unknown;
                accept;
            }
        }
    }
}
```

Juniper, Check BGP

```
>show route protocol bgp

inet.0: 17 destinations, 17 routes (17 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

10.0.0.0/16         *[BGP/170] 00:45:53, localpref 110
                       AS path: 200 20 I, validation-state: valid
                     > to 10.1.1.2 via em1.0
10.0.0.0/24         *[BGP/170] 00:42:58, localpref 9
                       AS path: 200 20 I, validation-state: invalid
                     > to 10.1.1.2 via em1.0
13.1.1.4/30         *[BGP/170] 3w1d 17:34:46, localpref 100
                       AS path: 200 20 I, validation-state: unknown
                     > to 10.1.1.2 via em1.0
```

Quagga (SRX)

- You need the SRX-server to speak RTR with cache and also to send data to router (quagga)
- Download from**
- Compile srx-server and run with /usr/local/bin/srx-server
- Compite Quagga with srx libraries and flags (read INSTALL for details)

Quagga

```
router bgp 1
 network 10.0.1.0/24
 neighbor 192.168.56.104 remote-as 20
 !SRx Configuration Settigns
 srx display
 srx connect 127.0.0.1 17900
 srx evaluation roa_only
 srx keep-window 900
```

Quagga, check BGP

```
bgpd# sh ip bgp 10.0.0.0/16
BGP routing table entry for 10.0.0.0/16
Paths: (1 available, best #1, table Default-IP-Routing-Table)
  Not advertised to any peer
  20
    SRx Information:
      Update ID: 0xB2E8F5E6
      Validation:
        prefix-origin: valid
        path processing disabled!
    192.168.56.104 from 192.168.56.104 (192.168.56.104)
      Origin IGP, metric 0, localpref 100, valid, external, best
      Last update: Wed Dec 31 22:38:17 1969
```

Quagga, BGP check

```
bgpd# sh ip bgg
BGP table version is 0, local router ID is 192.168.56.103
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale, R Removed
Validation:    v - valid, u - unknown, i - invalid, ? - undefined
SRx Status:    I - route ignored, D - SRx evaluation deactivated
SRxVal Format: validation result (origin validation, path validation)
Origin codes: i - IGP, e - EGP, ? - incomplete

   Ident    SRxVal SRxLP Status Network          Next Hop            Metric  LocPrf Weight Path
*> B2E8F5E6 v(v,-)              10.0.0.0/16      192.168.56.104           0              0 20 i
*> 093057FE i(i,-)              10.0.0.0/24      192.168.56.104           0              0 20 i
*  -------- ?(?,-)          I   10.0.1.0/24      0.0.0.0                  0          32768 i
*> D58A50E7 u(u,-)              10.10.0.0/16     192.168.56.104
```

LACNIC

Now, Thanks!!

Questions?