The RPKI & Origin Validation

SANOG / Paro

2010.07.19

Randy Bush <randy@psg.com>

Rob Austein <sra@isc.org>

Steve Bellovin <smb@cs.columbia.edu>

And a cast of thousands! Well, dozens :)

Routing is Very Fragile

- How long can we survive on The Web as Random Acts of Kindness, TED Talk by Jonathan Zittrain?

Routing Mistakes

- Routing errors are significant and have very high customer impact
- We need to fix this before we are crucified in the WSJ a la Toyota
- 99% of mis-announcements are accidental originations of someone else's prefix -- Google, UU, IIJ, …

Why Origin Validation?

- Prevent YouTube accident
- Prevent 7007 accident, UU/Sprint 2 days!
- Prevents most accidental announcements
- Does not prevent malicious path attacks such as the Kapela/Pilosov DefCon attack
- That requires "Path Validation" and locking the data plane to the control plane, the next steps, by my children

This is Not New

- 1986 – Bellovin identifies vulnerability
- 2000 – S-BGP – X.509 PKI to support Secure BGP - Kent, Lynn, et al.
- 2003 – NANOG S-BGP Workshop
- 2006 – ARIN & APNIC start work on RPKI. RIPE starts in 2008.
- 2009 – RPKI Open Testbed and running code in test routers

The Goal

- Keep the Internet working!!!
- Seriously reduce routing damage from mis-configuration, mis-origination

Non-Goals

- Prevent Malicious Attacks
- Keep RIRs in business by selling X.509 Certificates

How Can I Get This

- Remember Route-Maps?
- We used to test AS, Prefix, …
- What if Validity was testable?

Good Dog!

```
RP/0/1/CPU0:r0.dfw#show bgp 192.158.248.0/24
BGP routing table entry for 192.158.248.0/24
Versions:
  Process              bRIB/RIB  SendTblVer
  Speaker                132327      132327
Last Modified: Oct  2 01:06:47.630 for 13:33:12
Paths: (6 available, best #3)
  Advertised to peers (in unique update groups):
    204.69.200.26
  Path #1: Received by speaker 0
  2914 1299 6939 6939 27318
    157.238.224.149 from 157.238.224.149 (129.250.0.85)
      Origin IGP, metric 0, localpref 100, valid, external, \
         origin validity state: valid
      Community: 2914:420 2914:2000 2914:3000 4128:380
 Path #2: Received by speaker 0
...
```

Bad Dog!

```
RP/0/1/CPU0:r0.dfw#sh bgp 64.9.224.0
BGP routing table entry for 64.9.224.0/20
Versions:
  Process              bRIB/RIB  SendTblVer
  Speaker                    0           0
Last Modified: Oct  2 17:38:27.630 for 4d22h
Paths: (6 available, no best path)
  Not advertised to any peer
  Path #1: Received by speaker 0
  2914 3356 36492
    157.238.224.149 from 157.238.224.149 (129.250.0.85)
      Origin IGP, metric 2, localpref 100, valid, external,\
origin validity state: invalid
      Community: 2914:420 2914:2000 2914:3000 4128:380
```

Strange Dog!

```
RP/0/1/CPU0:r0.dfw#sh bgp 147.28.0.0
BGP routing table entry for 147.28.0.0/16
Versions:
  Process              bRIB/RIB  SendTblVer
  Speaker                337691      337691
Last Modified: Oct  2 17:40:16.630 for 4d22h
Paths: (6 available, best #1)
  Advertised to peers (in unique update groups):
    204.69.200.26
  Path #1: Received by speaker 0
  2914 3130
    157.238.224.149 from 157.238.224.149 (129.250.0.85)
      Origin IGP, metric 68, localpref 100, valid, external, \
origin validity state: not found
      Community: 2914:410 2914:2000 2914:3000 4128:380
```

The Solution is to Allow Operator to Test and then Set Local Policy

Secure

```
route-map validity-0
  match rpki-invalid
  drop
route-map validity-1
  match rpki-not-found
  set localpref 50
// valid defaults to 100
```

Paranoid

```
route-map validity-0
  match rpki-valid
  set localpref 110
route-map validity-1
  drop
```

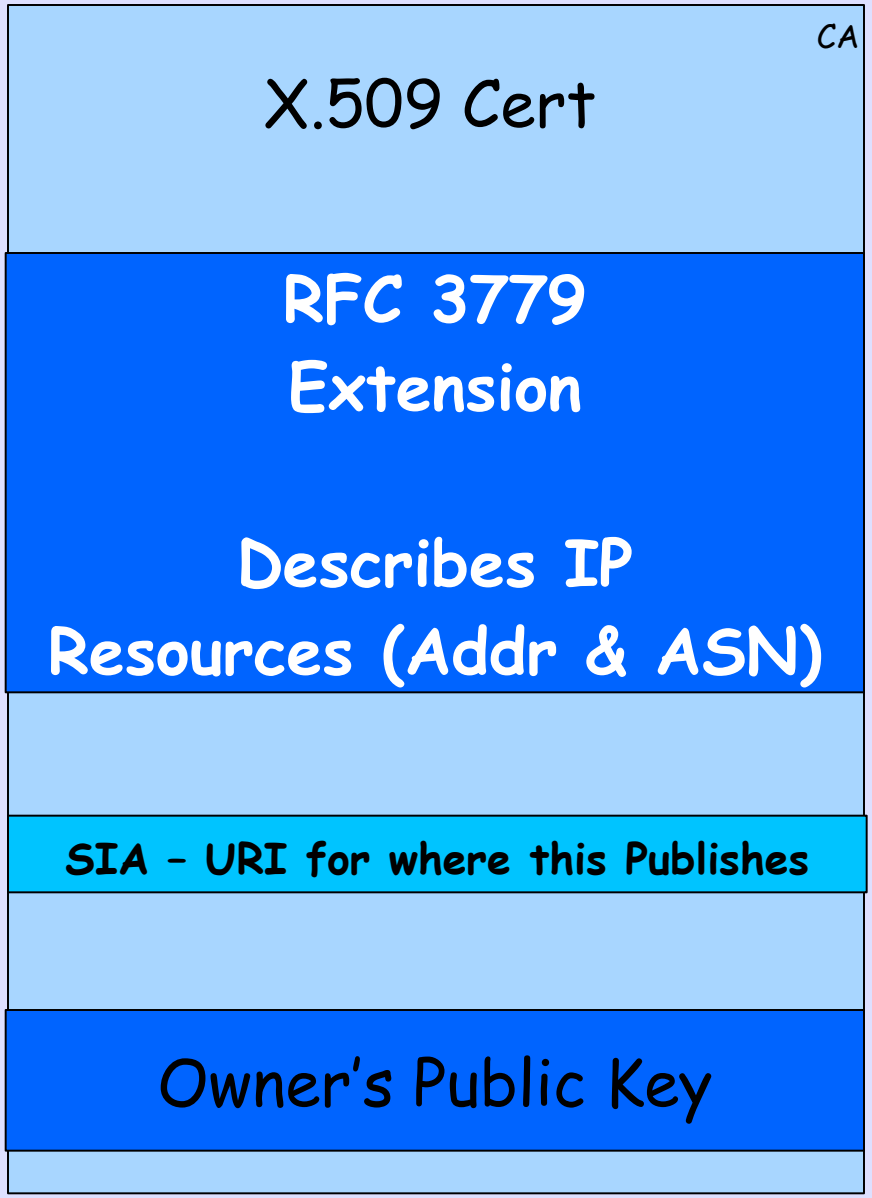
After AS-Path

```
route-map validity-0
 match rpki-unknown
   set metric 50
route-map validity-1
   match rpki-invalid
   set metric 25
// valid defaults to 100
```

But How Do We Decide If An Announcement Is Valid or Not?

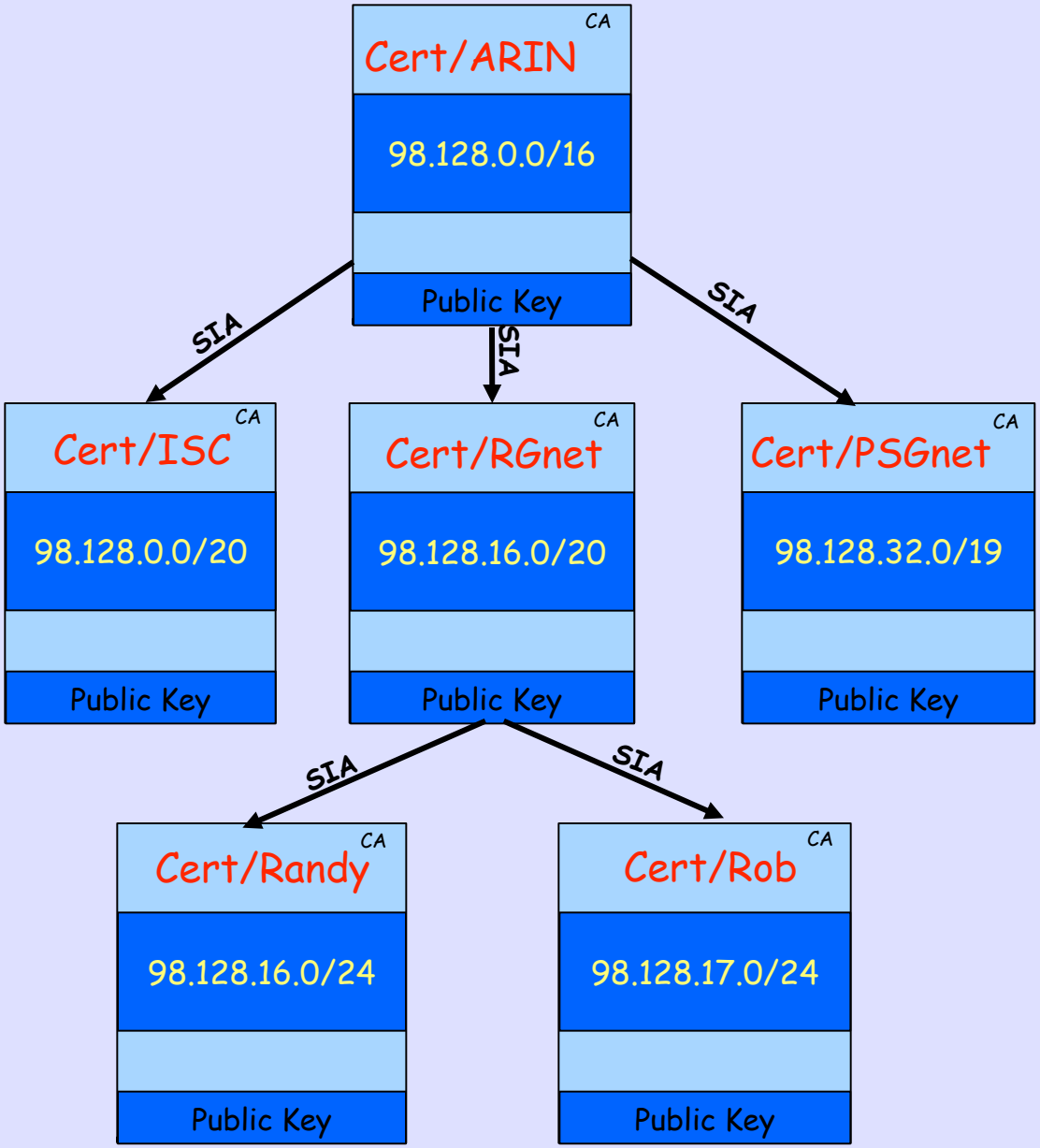
Resource Public Key Infrastructure (RPKI)

X.509 Certificate w/ 3779 Ext



Being Developed & Tested by RIRs and Operators

Certificate Hierarchy follows Allocation Hierarchy

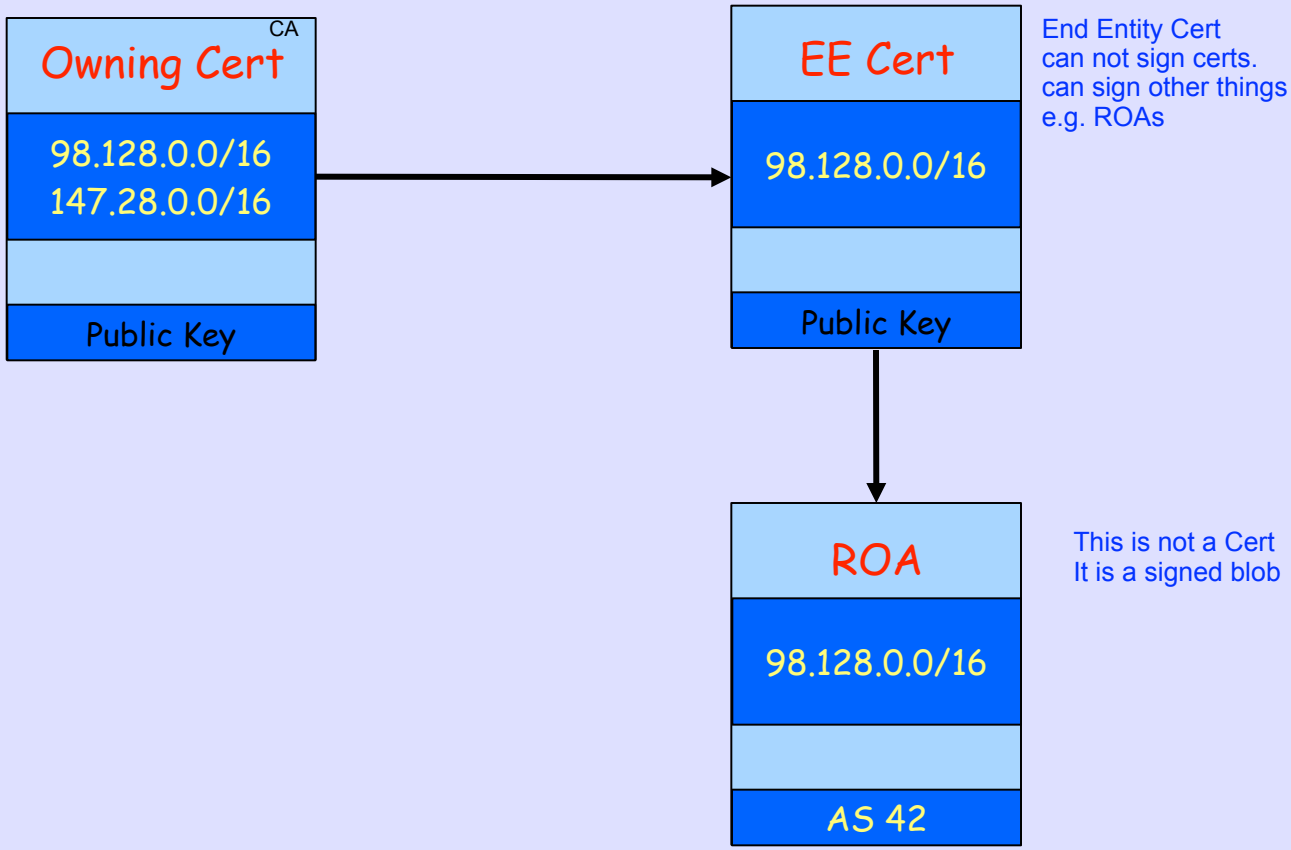


That's Who Owns It

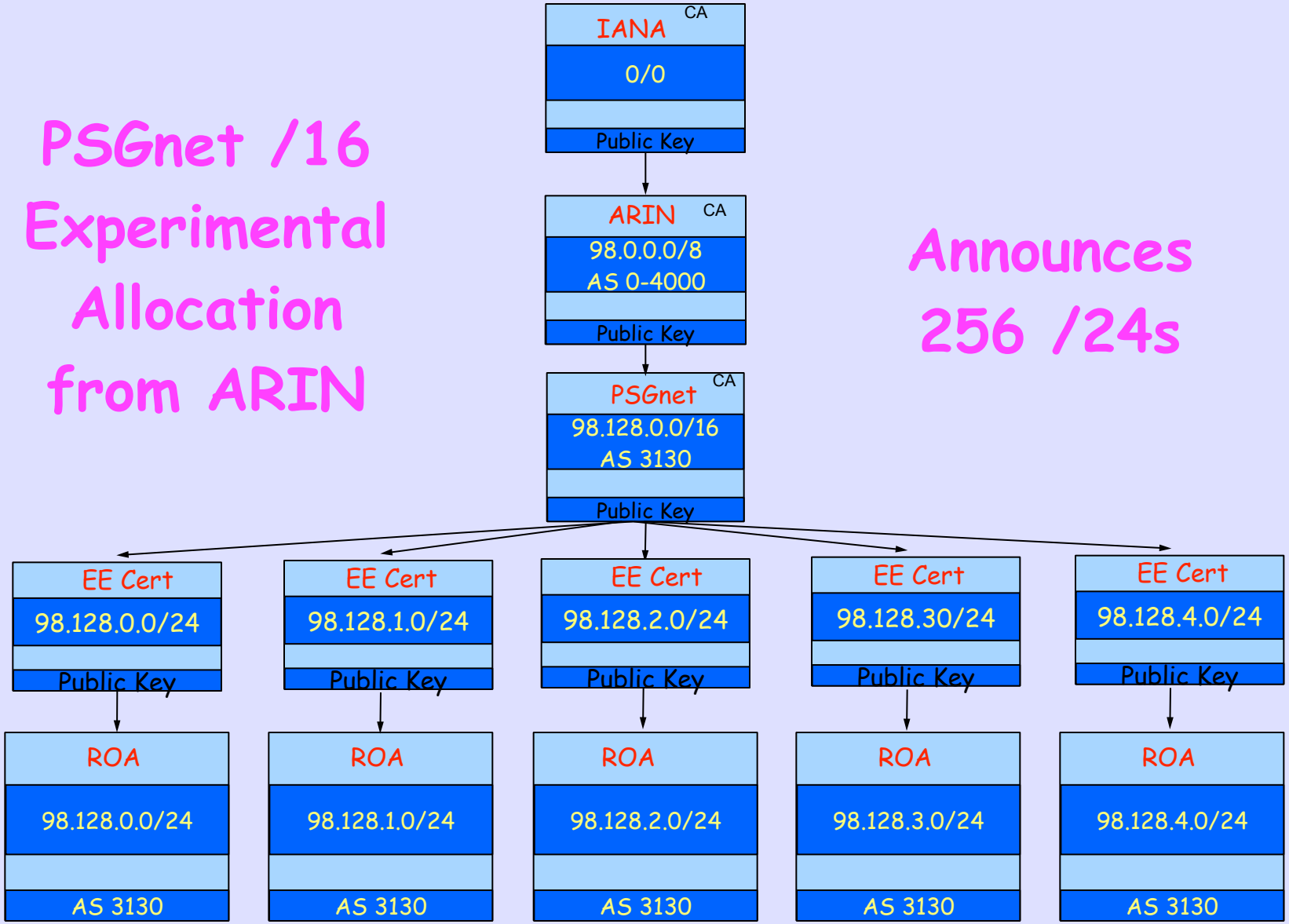
but

Who May Route It?

Route Origin Authorization (ROA)

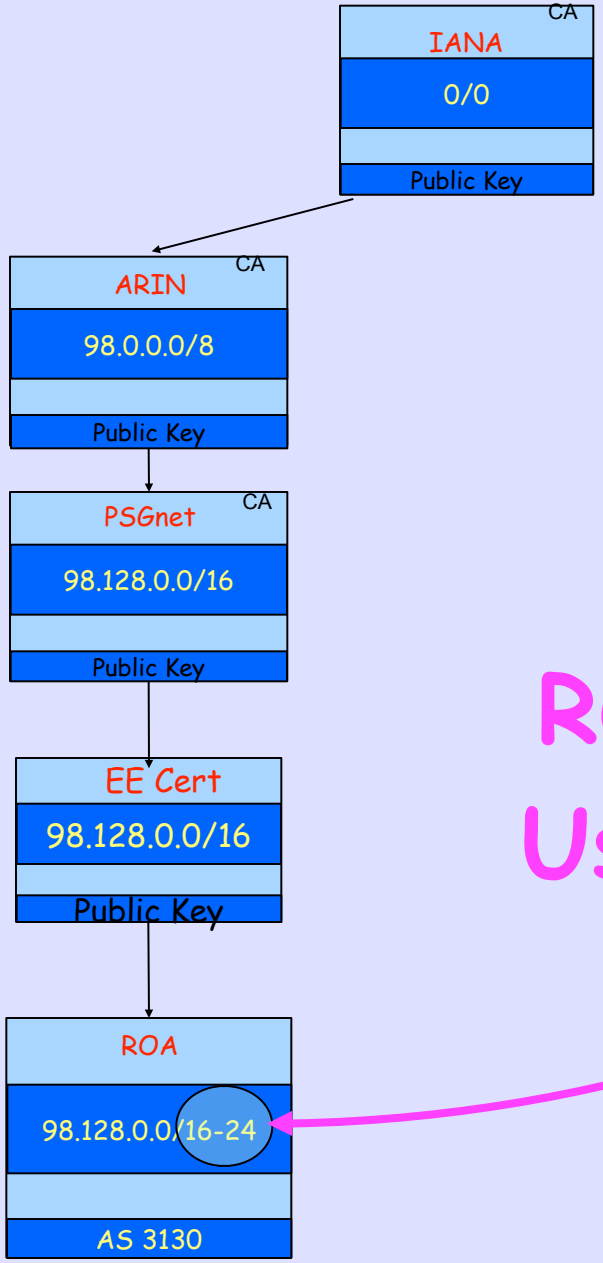


PSGnet /16 Experimental Allocation from ARIN



Announces 256 /24s

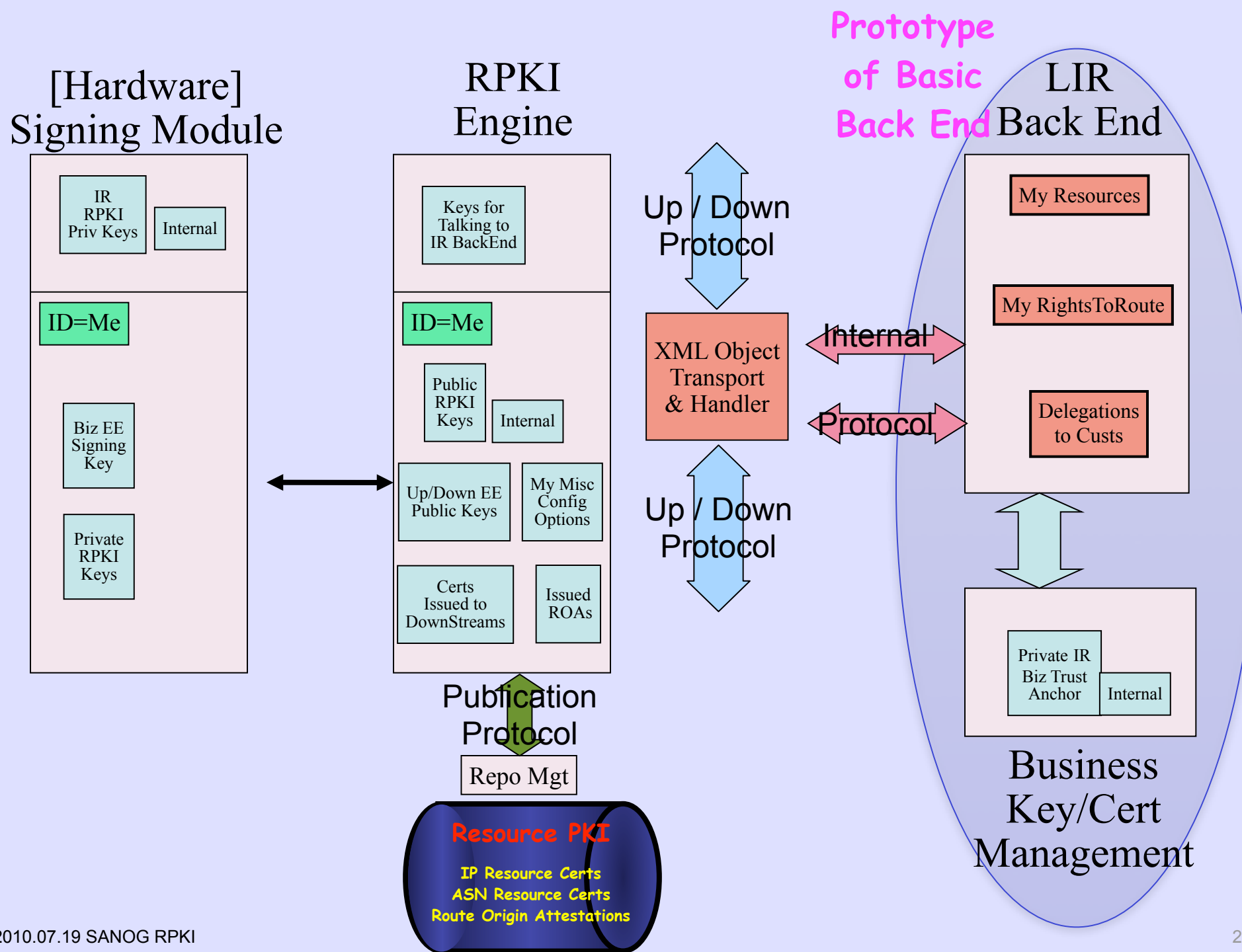
Too Many EE Certs and ROAs, Yucchhy!



ROA Aggregation Using Max Length

Running Code

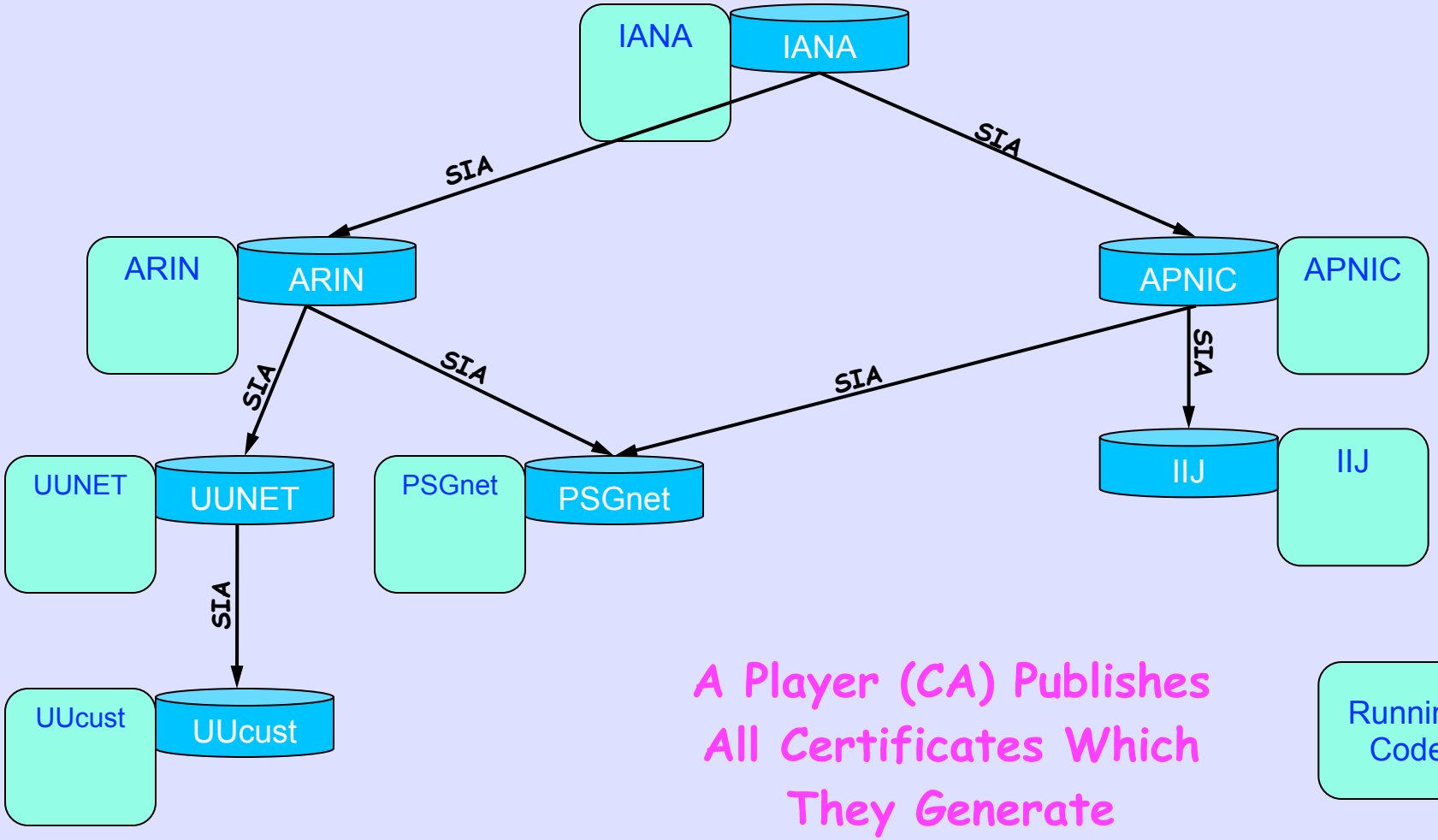
And the Three RPKI Protocols


Prototype of Basic Back End
[Hardware] Signing Module
IR RPKI Priv Keys
Internal
ID=Me
Biz EE Signing Key
Private RPKI Keys
RPKI Engine
Keys for Talking to IR BackEnd
ID=Me
Public RPKI Keys
Internal
Up/Down EE Public Keys
My Misc Config Options
Certs Issued to DownStreams
Issued ROAs
Publication Protocol
Repo Mgt
Resource PKI
IP Resource Certs
ASN Resource Certs
Route Origin Attestations
Up / Down Protocol
XML Object Transport & Handler
Up / Down Protocol
Internal
Protocol
LIR Back End
My Resources
My RightsToRoute
Delegations to Custs
Private IR Biz Trust Anchor
Internal
Business Key/Cert Management

Big, Centralized, & Scary

We Don't Do This

Distributed RPKI DataBase



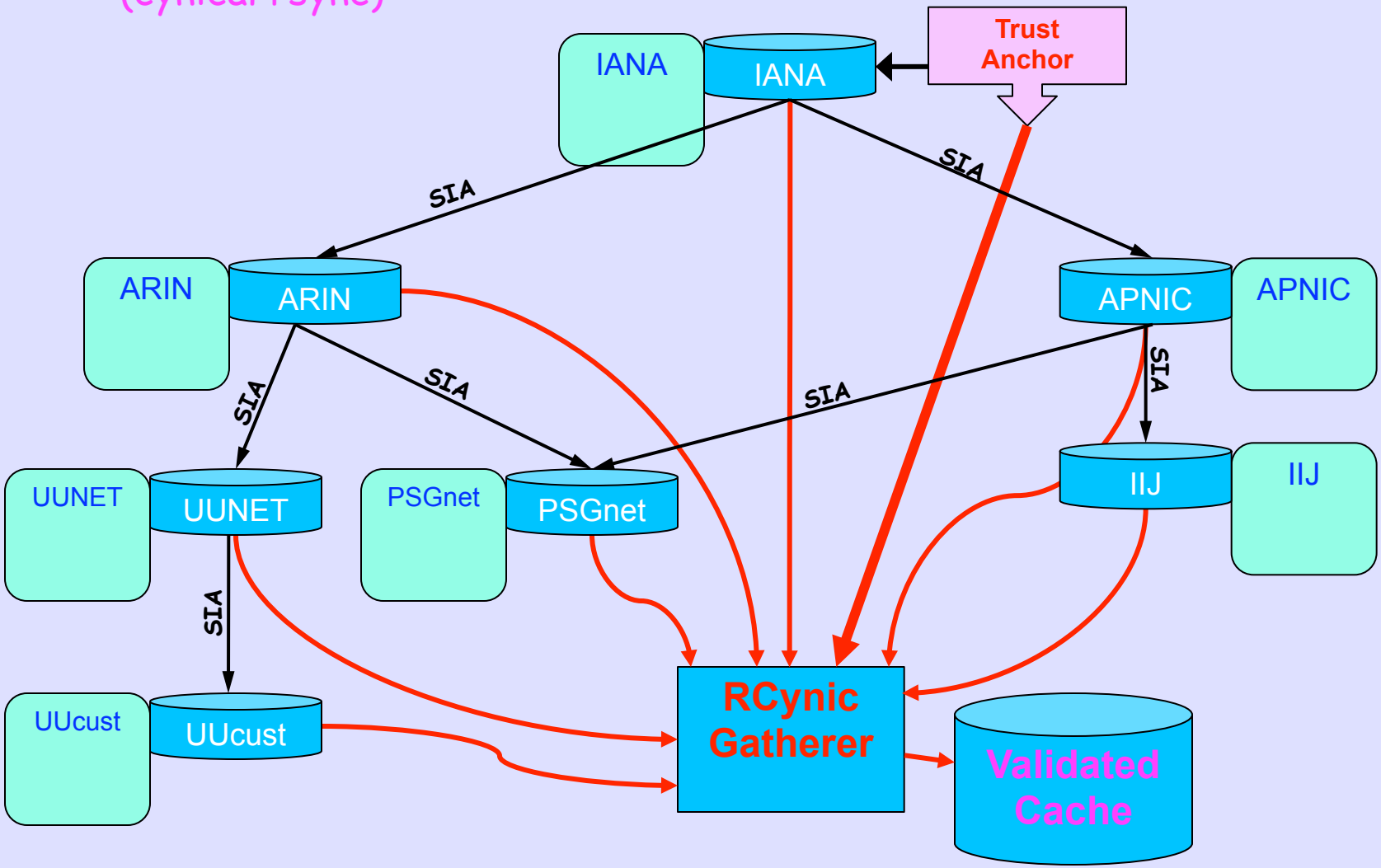
A Player (CA) Publishes
All Certificates Which
They Generate
in Their Own Unique
Publication Point

Running Code

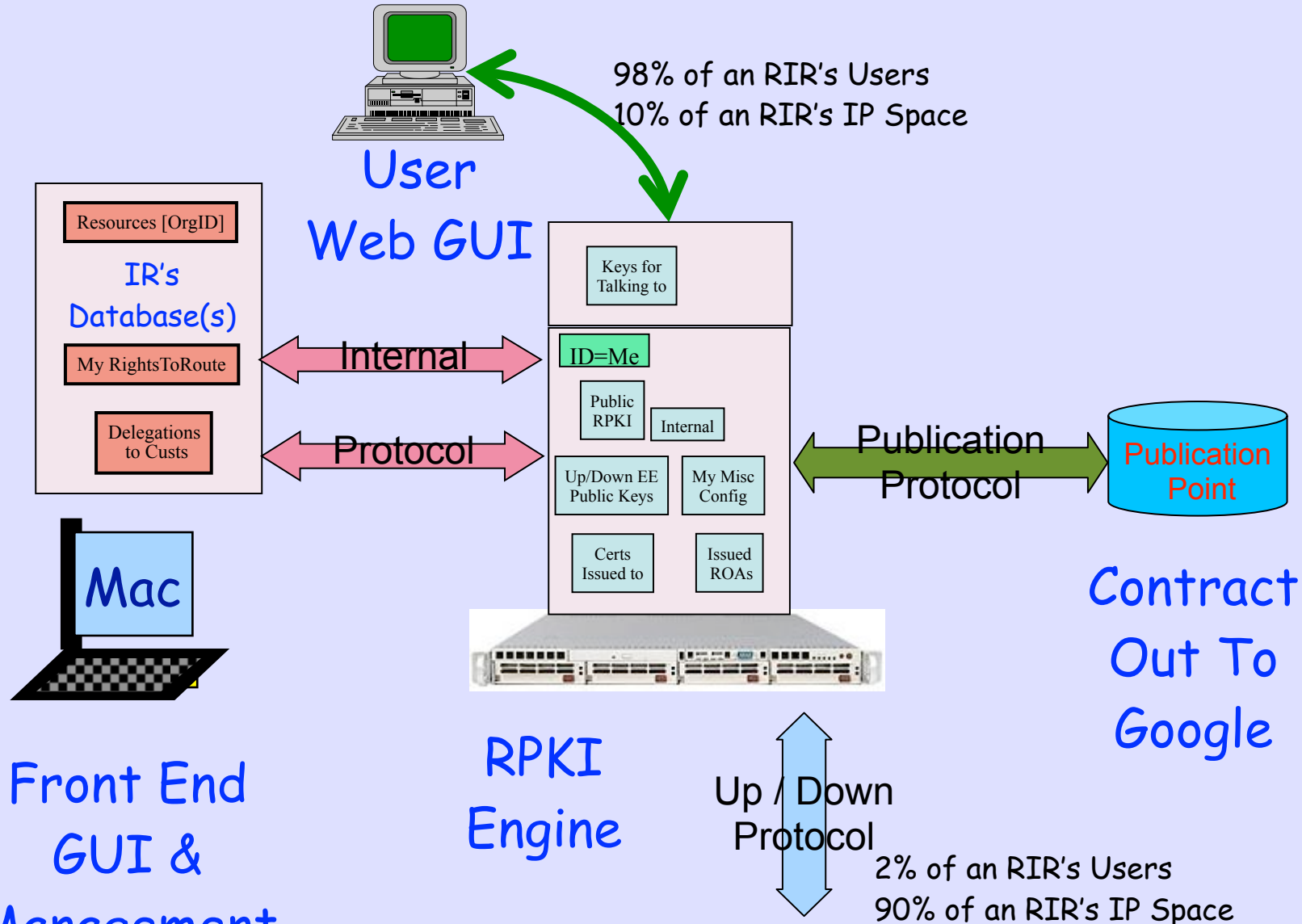
Repository

RCynic Cache Gatherer

(cynical rsync)



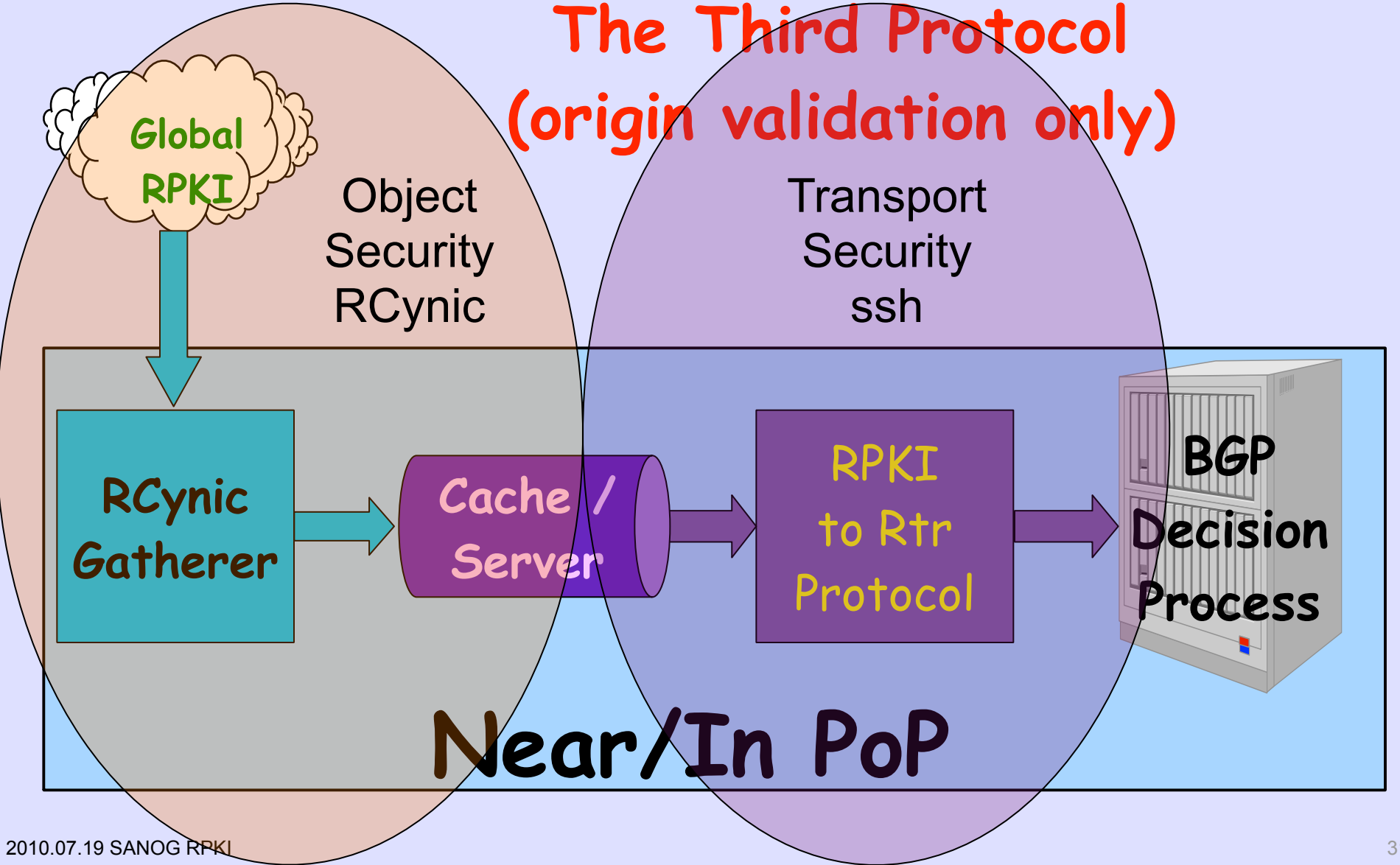
A Usage Scenario



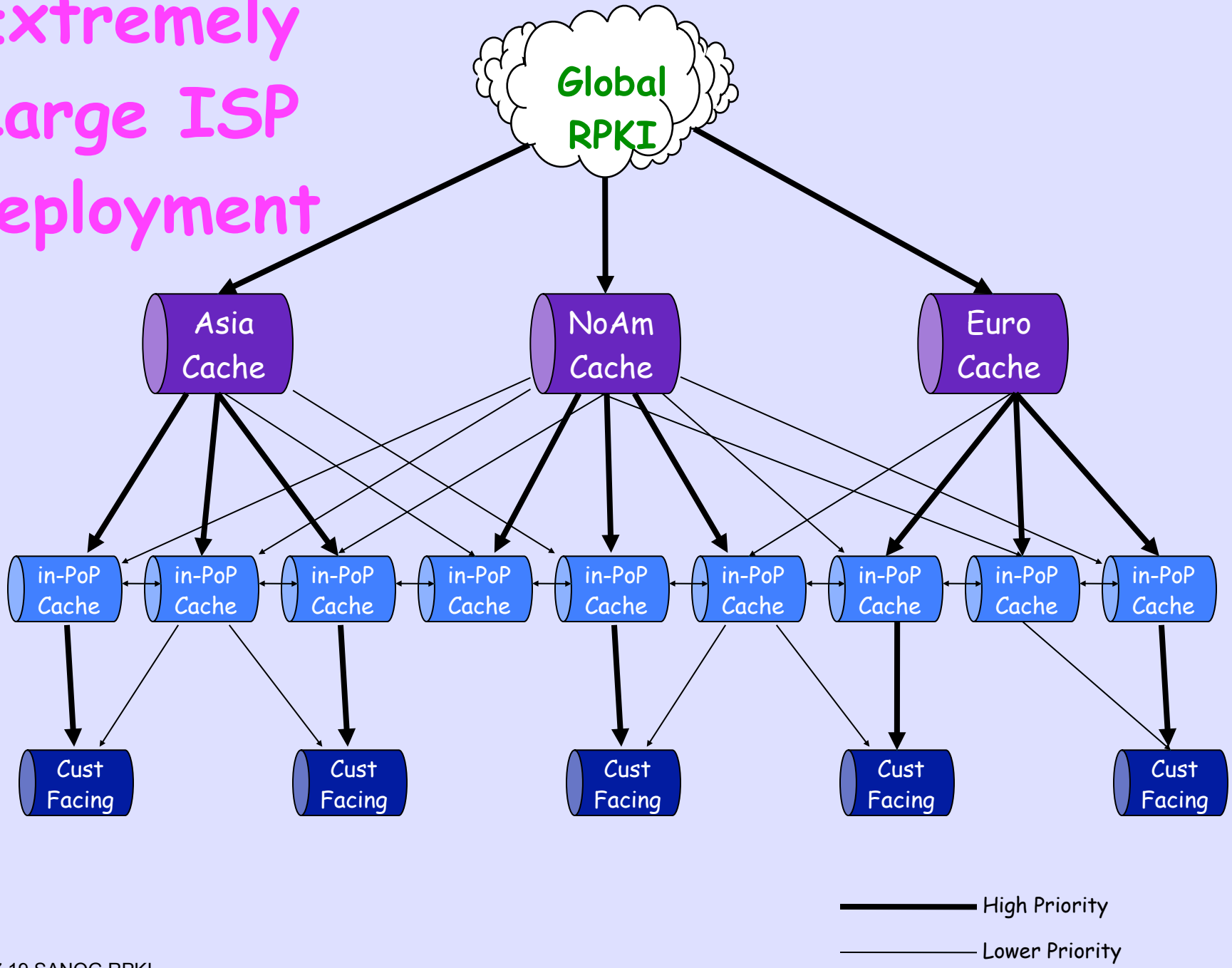
The Router Side

- Cisco IOS and IOS-XR test code have Origin Validation now
- Work continues daily in test routers
- Compute load much less than ACLs from IRR data, **10μsec per update!**
- Expect other vendor soon

RPKI -> Router


The Third Protocol
(origin validation only)
Global RPKI
Object Security RCynic
Transport Security ssh
RCynic Gatherer
Cache / Server
RPKI to Rtr Protocol
BGP Decision Process
Near/In PoP

Extremely Large ISP Deployment



Configure

```
router bgp 4128 bgp router-id 198.180.152.251
   bgp rpki cache 198.180.150.1 42420 refresh-time 600
   address-family ipv4 unicast
   bgp dampening collect-statistics ebgp
   redistribute static route-policy vb-ebgp-out
   ...
```

Result of Check

- **Valid** – A matching/covering ROA was found with a matching AS number
- **Invalid** – A matching or covering ROA was found, but AS number did not match, and there was no valid one
- **Not Found** – No matching or covering ROA was found

Policy Override Knobs

- Disable Validity Check Completely
- Disable Validity Check for a Peer
- Disable Validity Check for Prefixes

When check is disabled, the result is "Not Found," i.e. as if there was no ROA

Show commands

```
RP/0/5/CPU0:ios#show bgp rpki prefix-validation database
Thu Jul 16 15:56:43.805 UTC
Network          Maxlen          Origin-AS     Color        Source
8.0.0.0/4        6               200           0            0
1.1.0.0/16       24              1             0            0
3.0.0.0/24       24              2             0            0
4.0.0.0/8        8               3             0            0
4.0.0.0/24       24              3             0            0
5.0.0.0/24       24              4             0            0
10.0.0.0/6       8               100           0            0
8.0.0.0/8        24              36394         0            0
11.0.0.0/16      24              100           0            0
12.0.0.0/8       8               7018          0            0
20.137.0.0/21    21              4237          0            0
```

RPKI Full Implementation Available as Open Source

https://subvert-rpki.hactrn.net/

and there is a mailing list

Work Supported By

- **US Government**

  THIS PROJECT IS SPONSORED BY THE DEPARTMENT OF HOMELAND SECURITY UNDER AN INTERAGENCY AGREEMENT WITH THE AIR FORCE RESEARCH LABORATORY (AFRL).

- **ARIN**
- **Internet Initiative Japan**
- **Cisco, Google, NTT, Equinix**