The RPKI & Origin Validation

ARIN / San Juan

2011.04.10

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

The Goal

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

Non-Goals

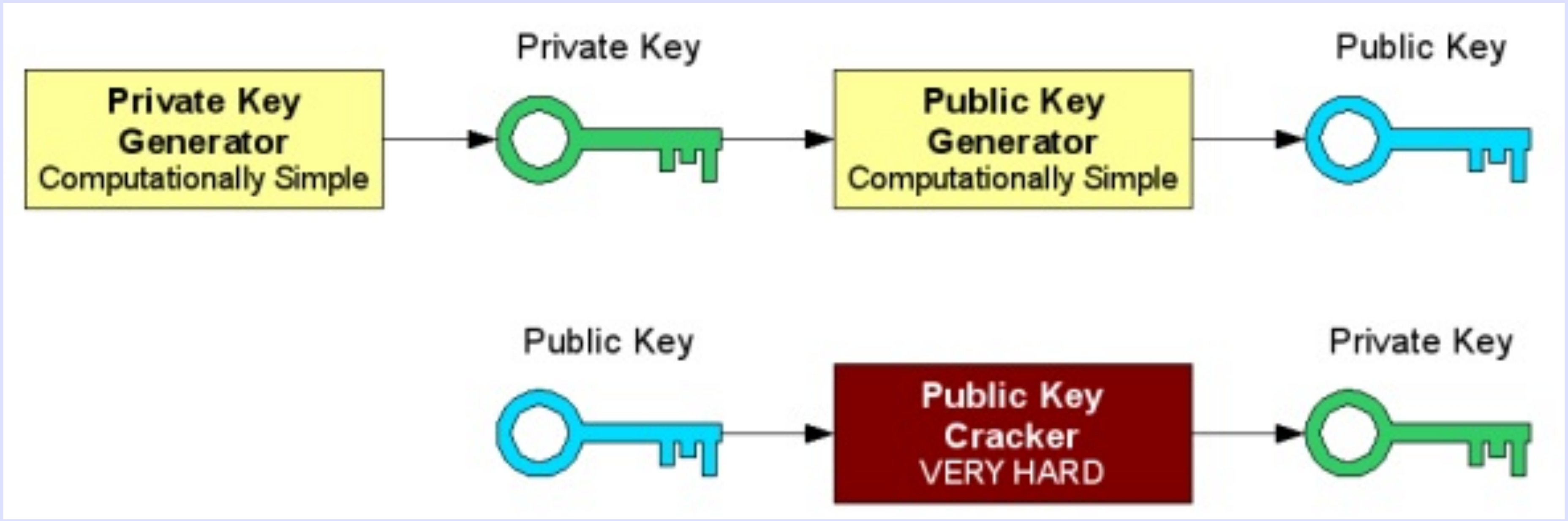
- Prevent Malicious Attacks
- Keep RIRs in business by selling X.509 Certificates

Resource Public Key Infrastructure (RPKI)

Public-Key Concept

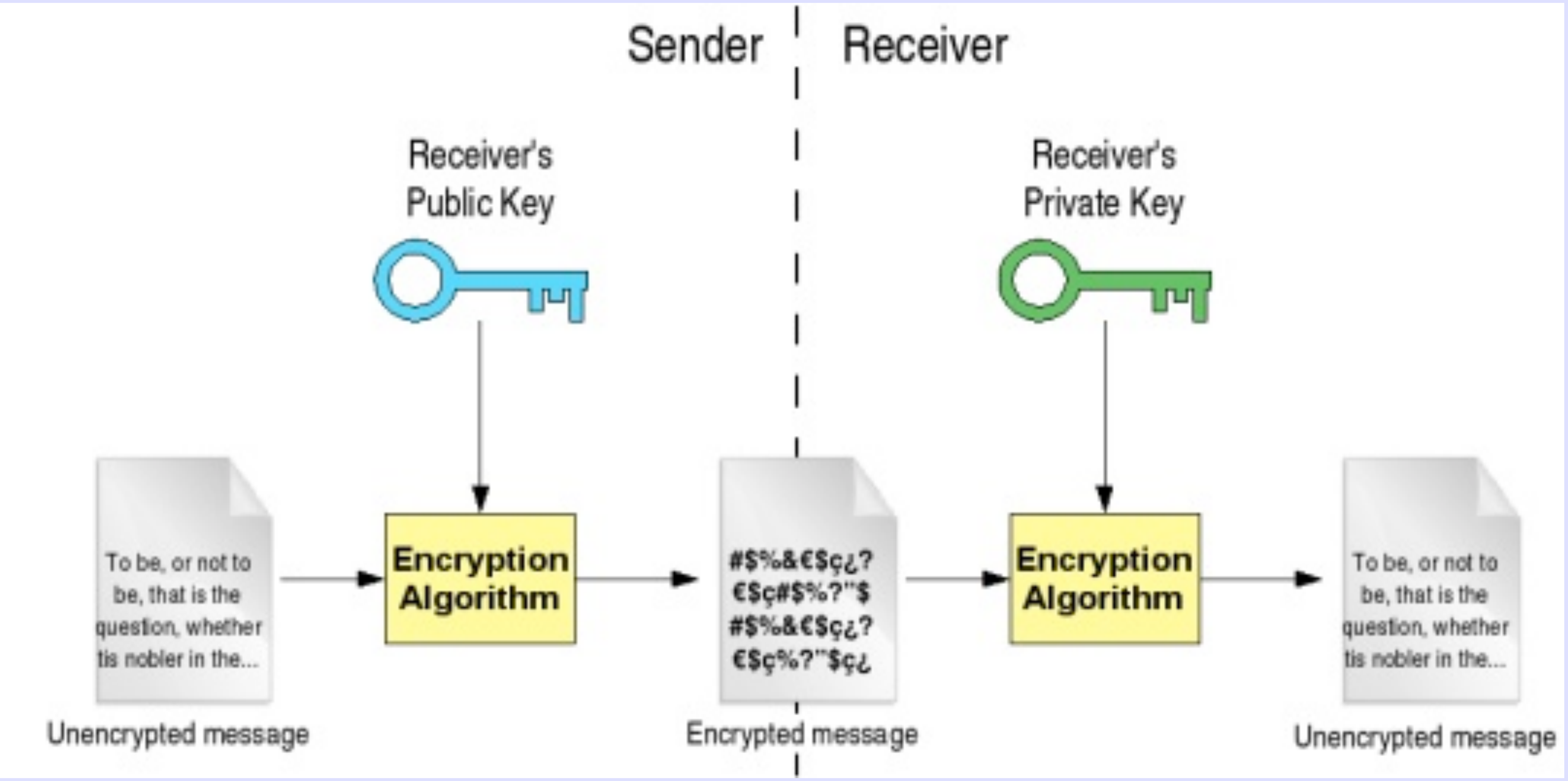
- **Private key**: This key must be known *only* by its owner.
- **Public key**: This key is known to everyone (it is *public*)
- **Relation between both keys**: What one key encrypts, the other one decrypts, and vice versa. That means that if you encrypt something with my public key (which you would know, because it's public :-), I would need my private key to decrypt the message.

Key Generation

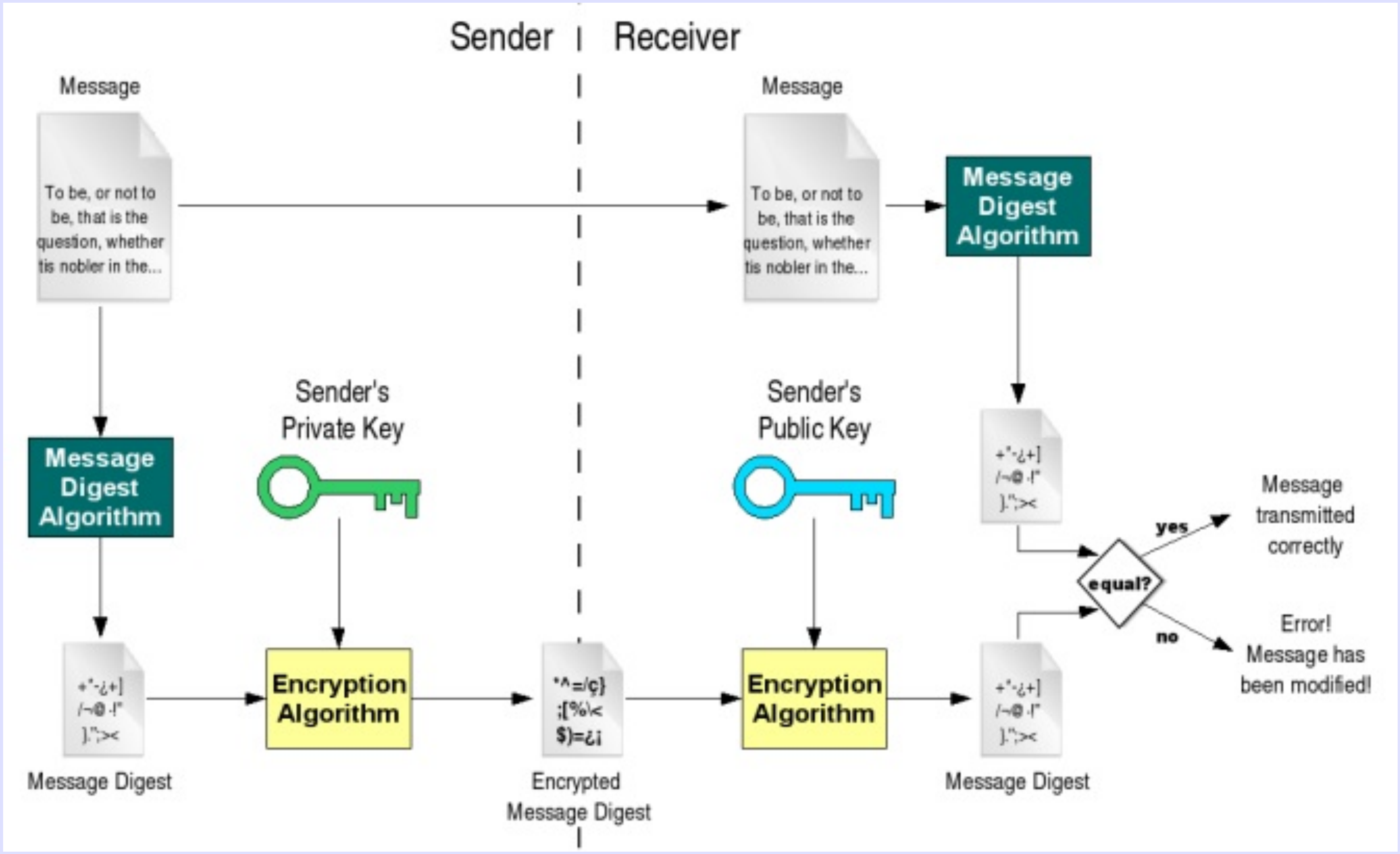


Stolen from - http://gdp.globus.org/gt4-tutorial/multiplehtml/ch09s03.html

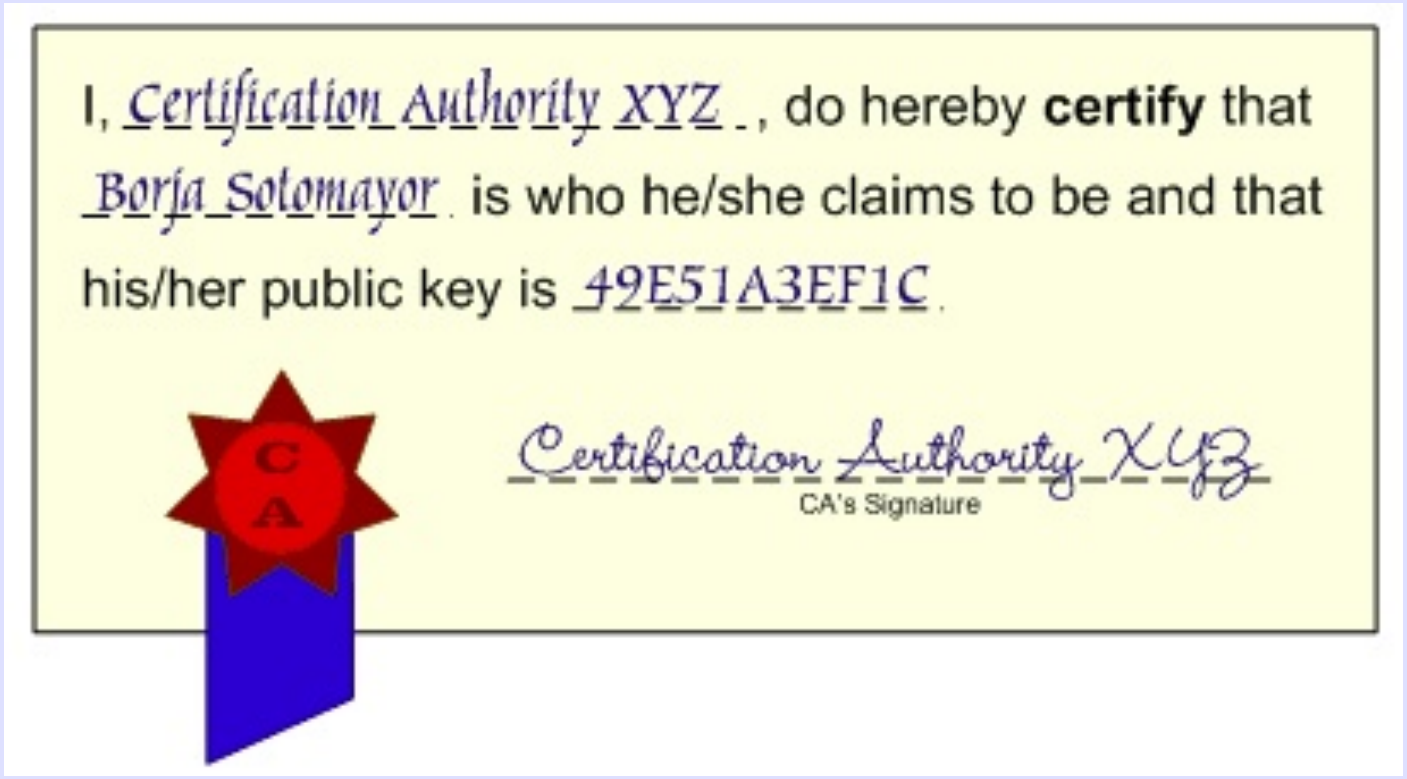
En/DeCryption


Sender
Receiver
Receiver's Public Key
Receiver's Private Key
To be, or not to be, that is the question, whether tis nobler in the...
Encryption Algorithm
#$%&€$ç¿? €$ç#$%?"$ #$%&€$ç¿? €$ç%?"$ç¿
Encryption Algorithm
To be, or not to be, that is the question, whether tis nobler in the...
Unencrypted message
Encrypted message
Unencrypted message

Digital Signature



Certificate



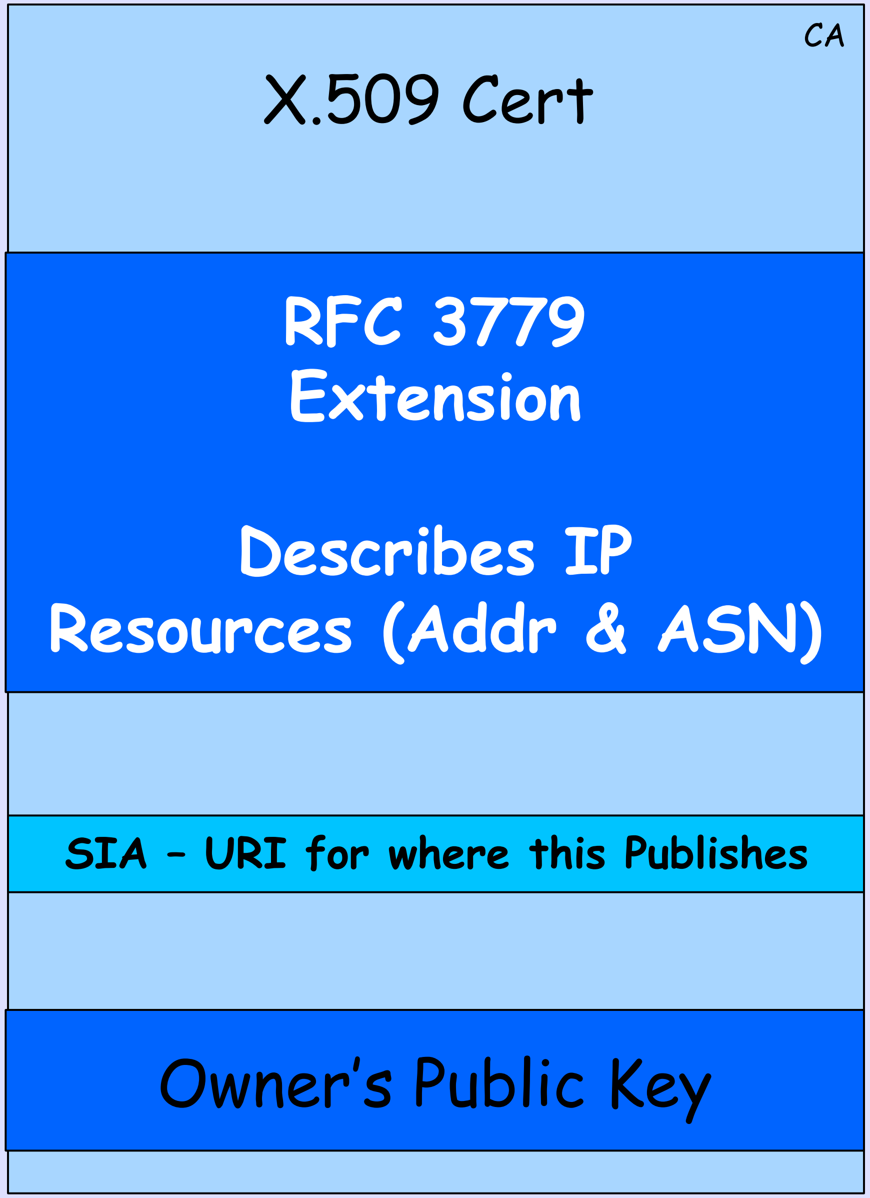
I, Certification Authority XYZ, do hereby **certify** that Borja Sotomayor is who he/she claims to be and that his/her public key is 49E51A3EF1C.

Certification Authority XYZ

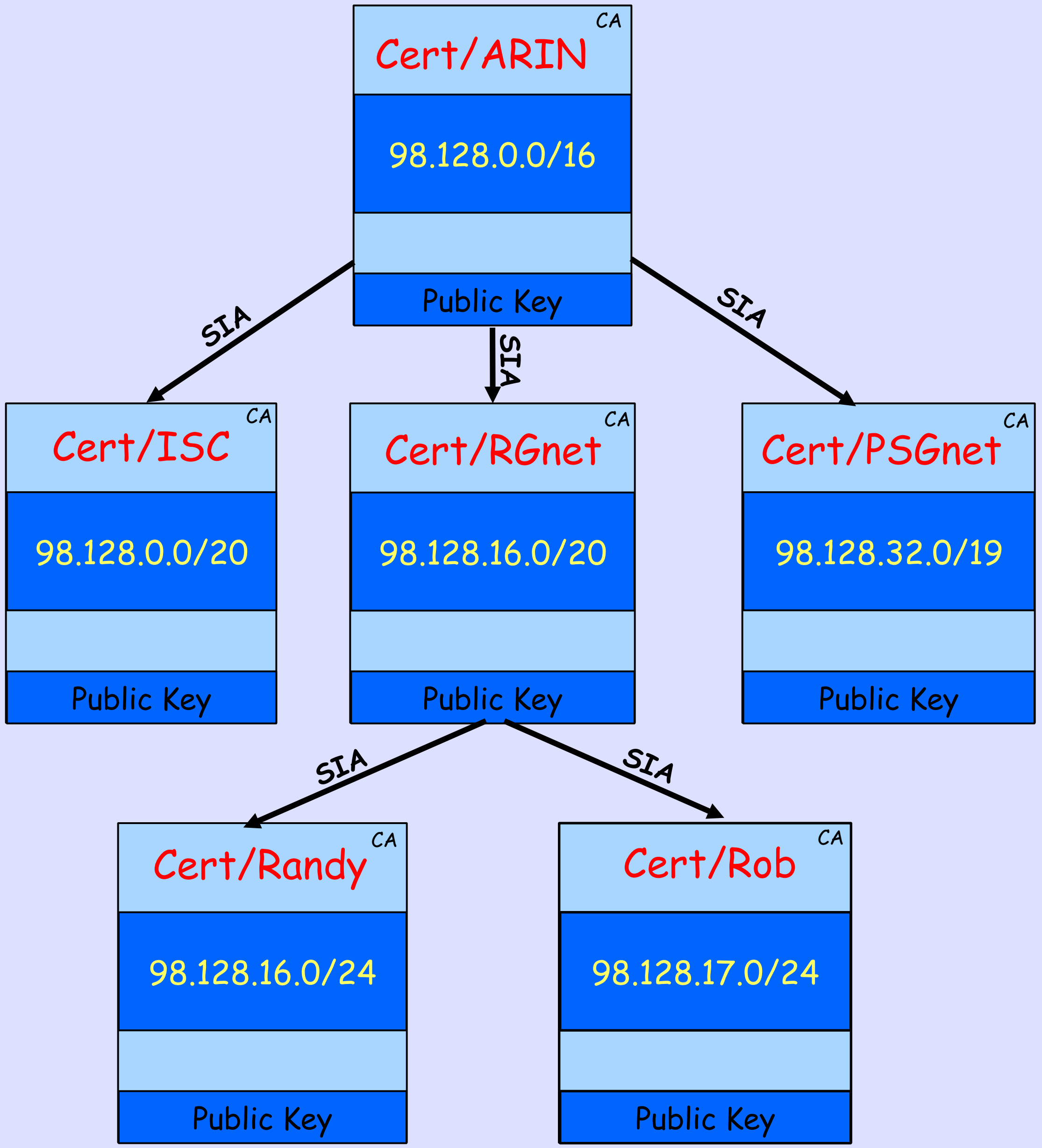
CA's Signature

X.509 RPKI Being Developed & Deployed by IANA, RIRs, and Operators

X.509 Certificate w/ 3779 Ext

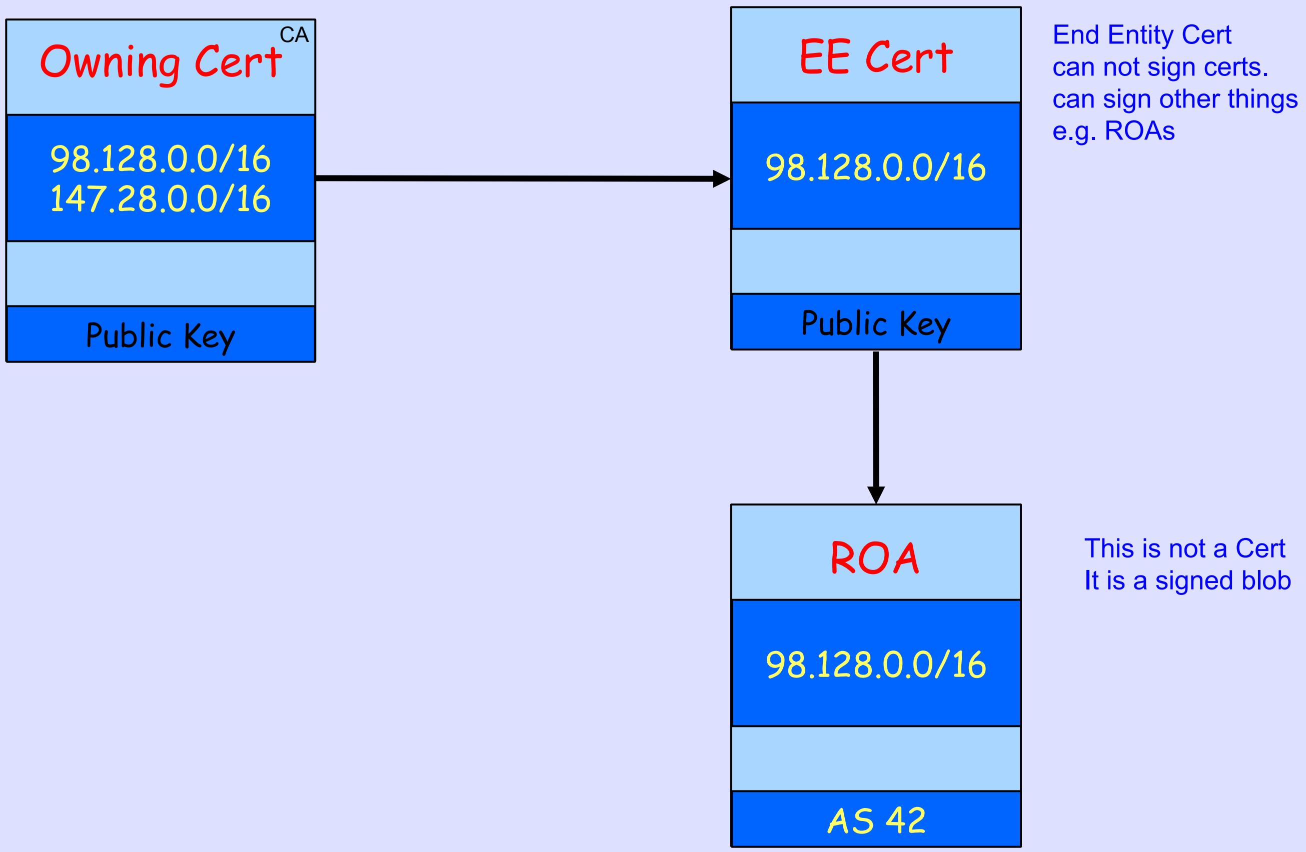

CA
X.509 Cert
RFC 3779 Extension
Describes IP Resources (Addr & ASN)
SIA – URI for where this Publishes
Owner's Public Key

Certificate Hierarchy follows Allocation Hierarchy



That's Who Owns It
but
Who May Route It?

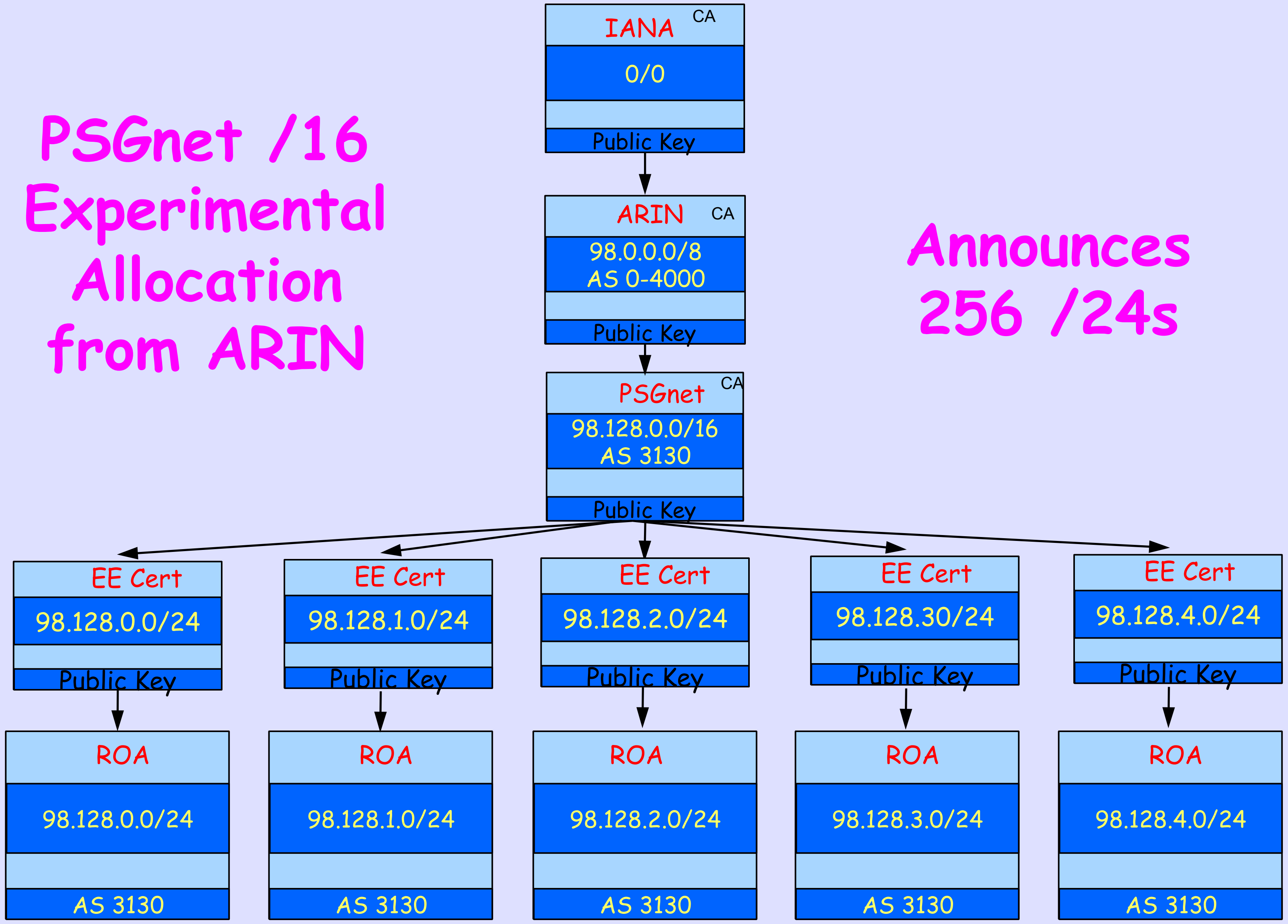
Route Origin Authorization (ROA)



End Entity Cert
can not sign certs.
can sign other things
e.g. ROAs

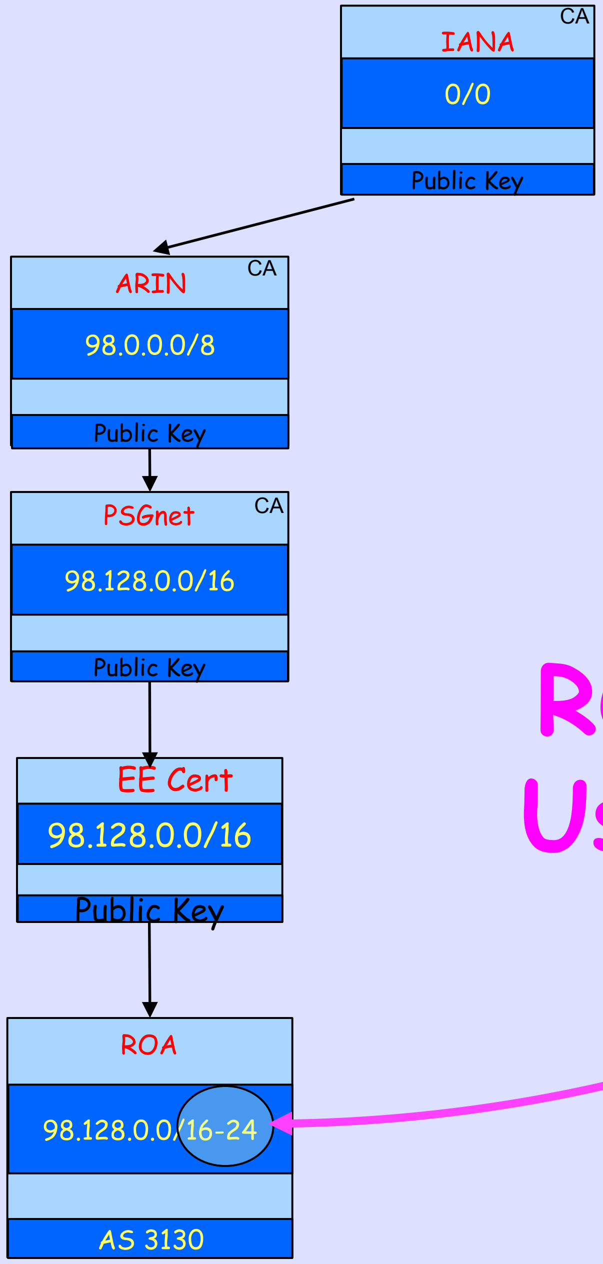
This is not a Cert
It is a signed blob

PSGnet /16 Experimental Allocation from ARIN



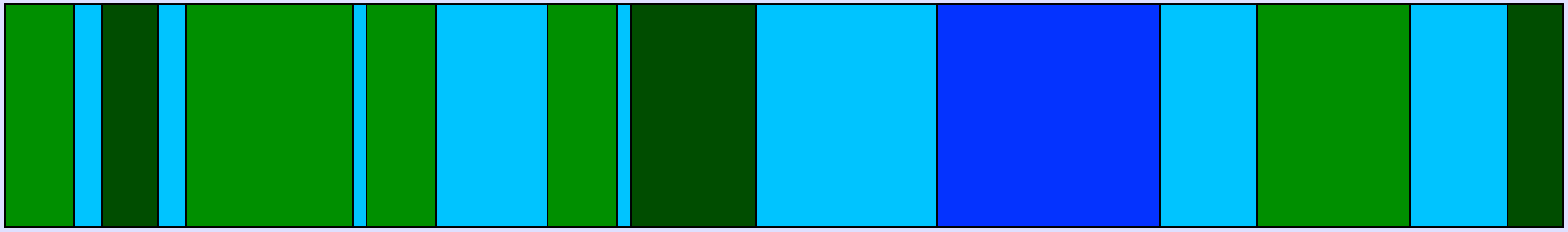
Announces 256 /24s

Too Many EE Certs and ROAs, Yucchhy!



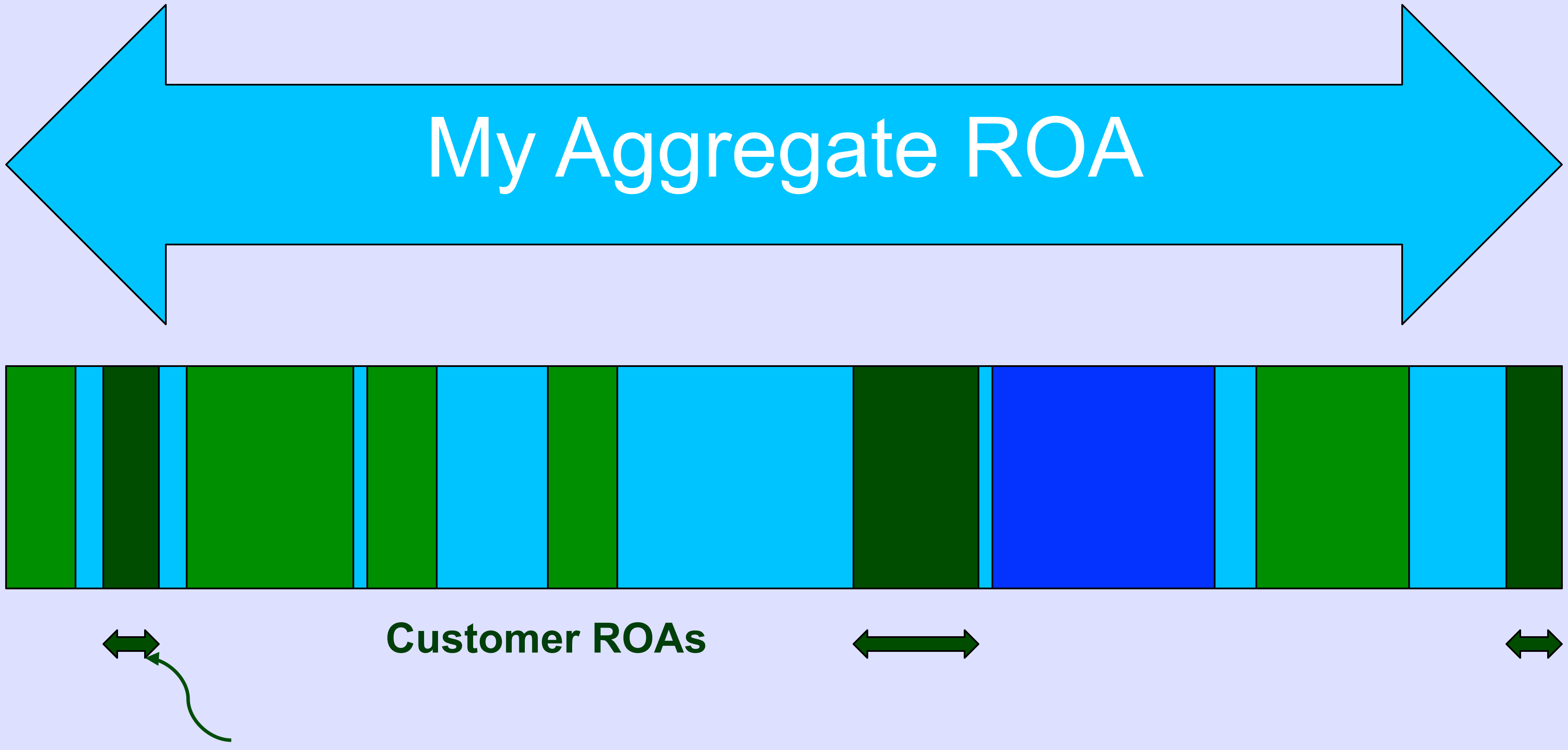
ROA Aggregation Using Max Length

Allocation in Reality



🟦 My Infrastructure
⬛ BGP Cust
🟩 Static (non BGP) Cust
🟦 Unused

ROA Use



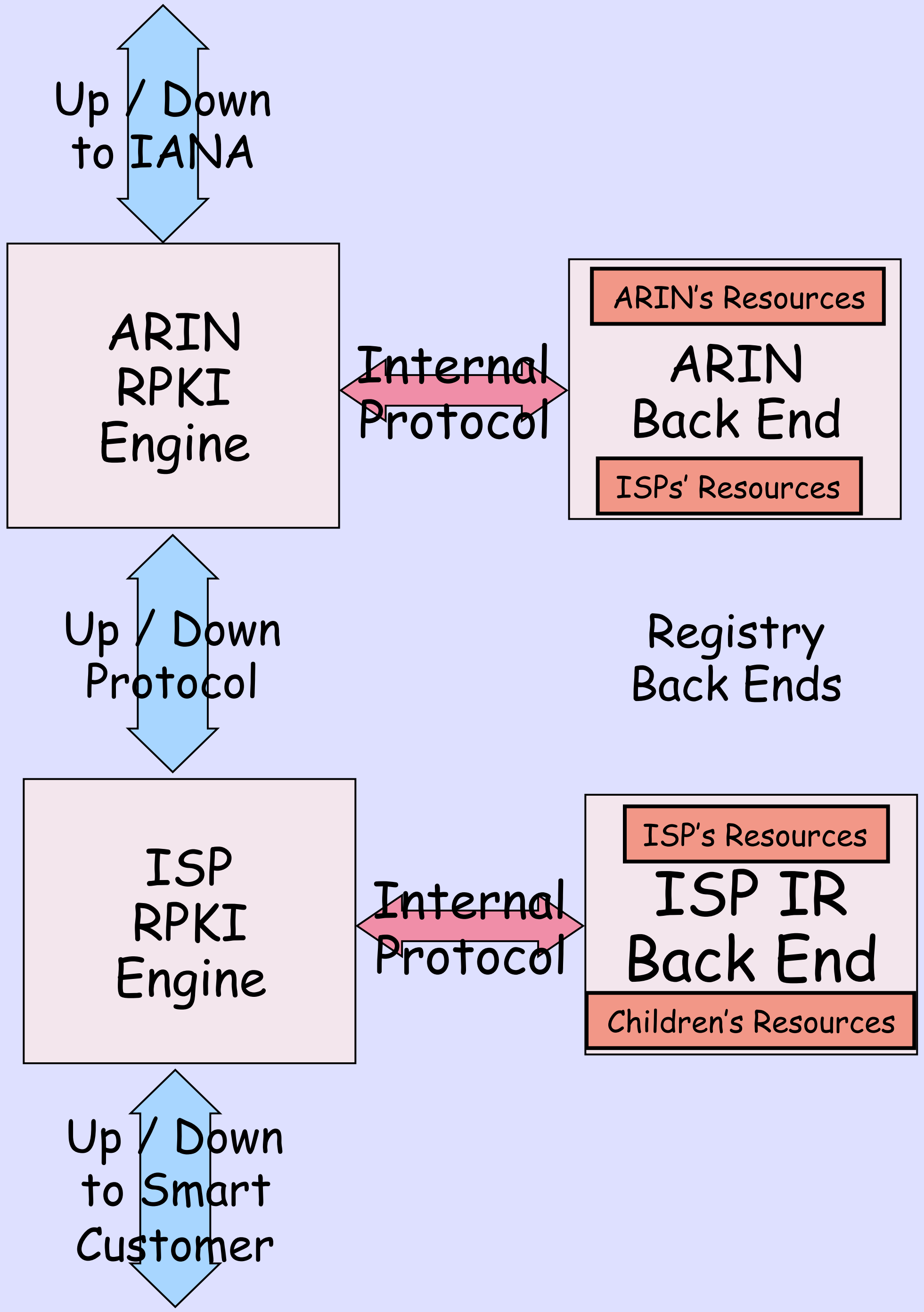
I Generate for
'Lazy' Customer

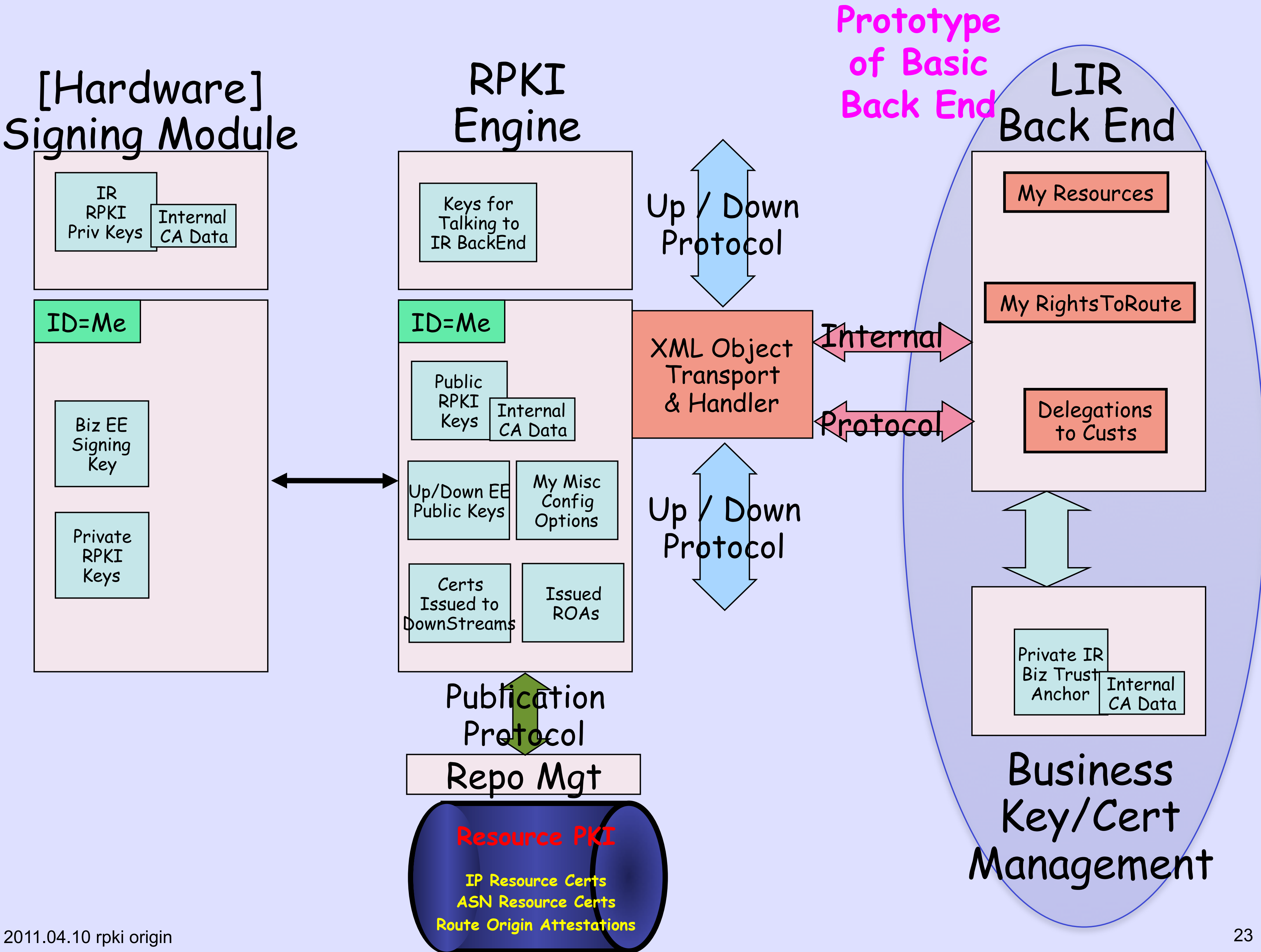
🟦	My Infrastructure	🟩	BGP Cust
🟩	Static (non BGP) Cust	🟦	Unused

Running Code

And the Three RPKI Protocols

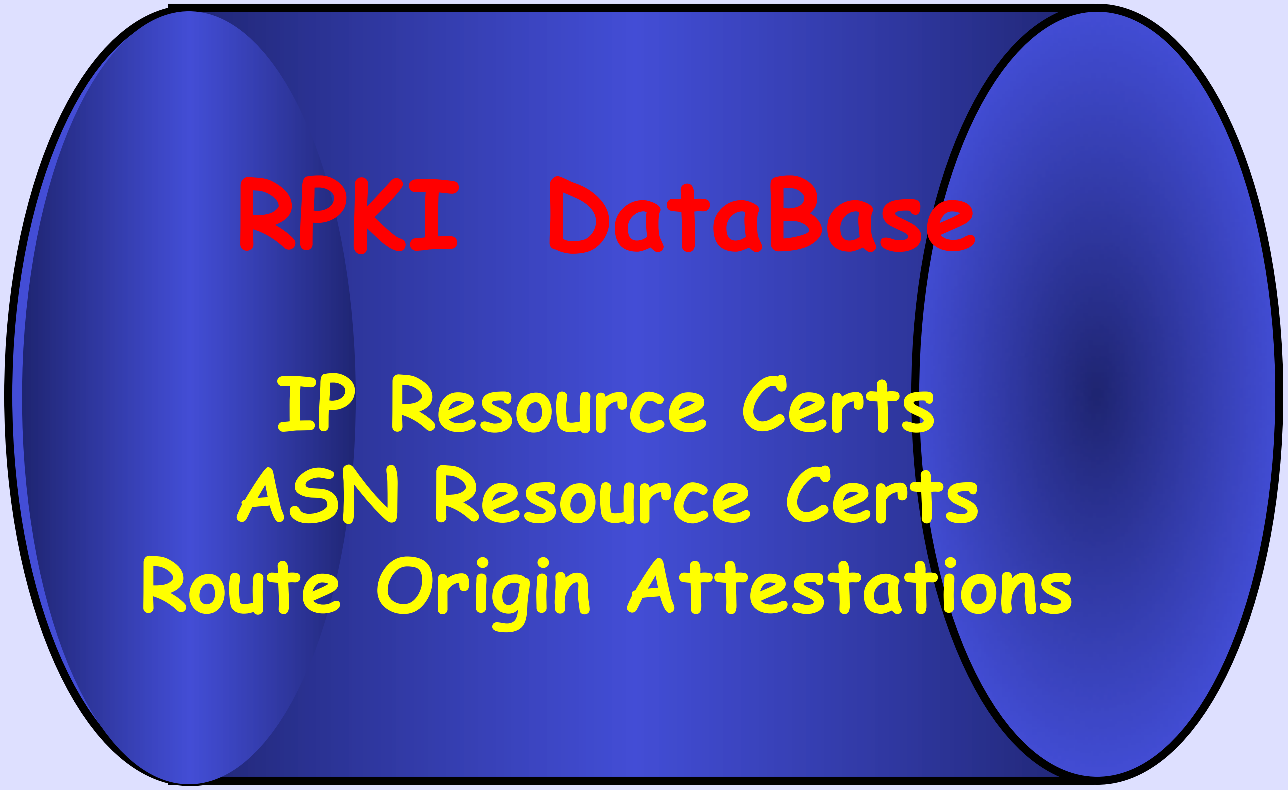
Parent and Child



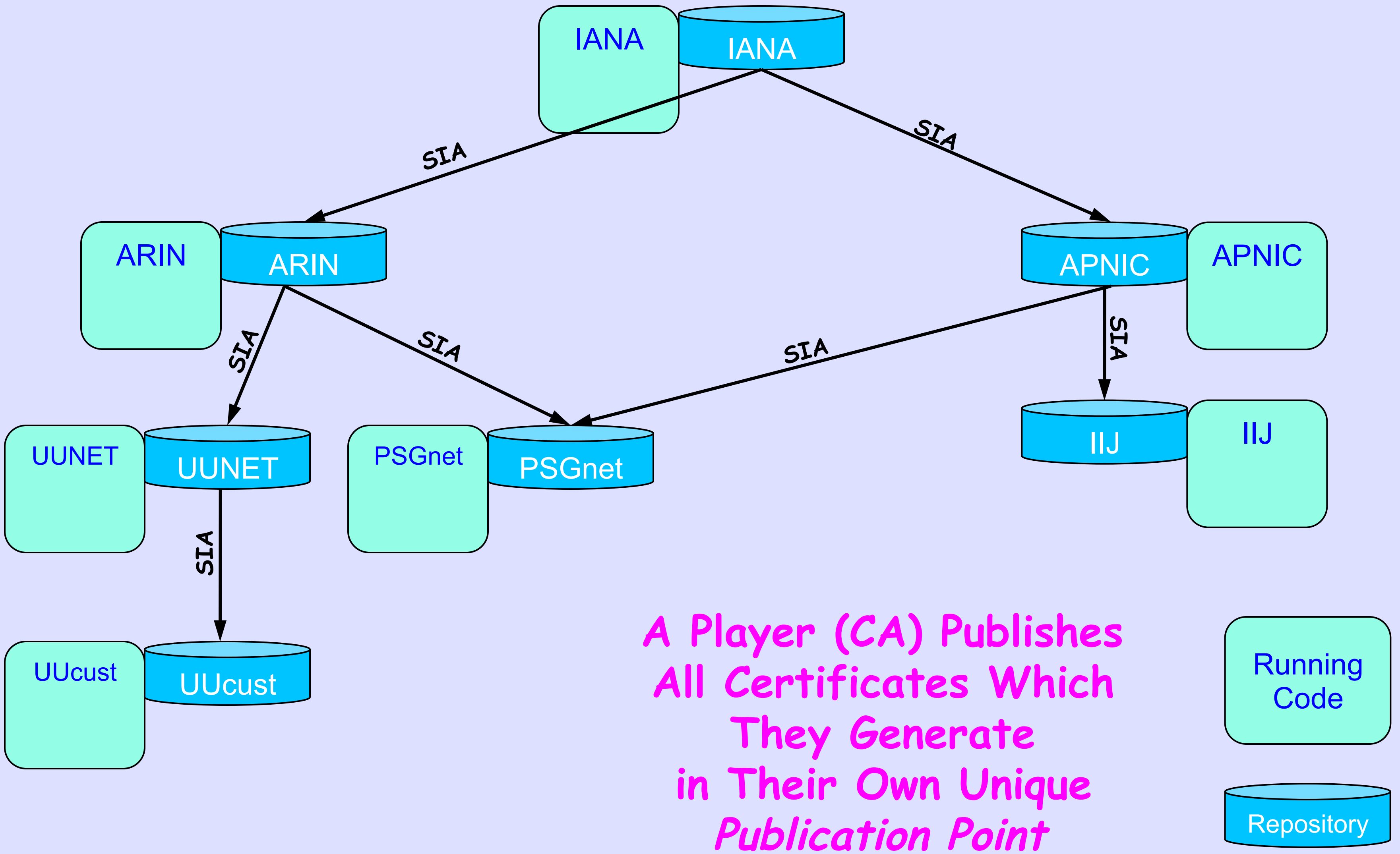


Prototype of Basic Back End
[Hardware] Signing Module
IR RPKI Priv Keys
Internal CA Data
ID=Me
Biz EE Signing Key
Private RPKI Keys
RPKI Engine
Keys for Talking to IR BackEnd
ID=Me
Public RPKI Keys
Internal CA Data
Up/Down EE Public Keys
My Misc Config Options
Certs Issued to DownStreams
Issued ROAs
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 CA Data
Business Key/Cert Management
Publication Protocol
Repo Mgt
Resource PKI
IP Resource Certs
ASN Resource Certs
Route Origin Attestations

Big, Centralized, & Scary
We Don't Do This



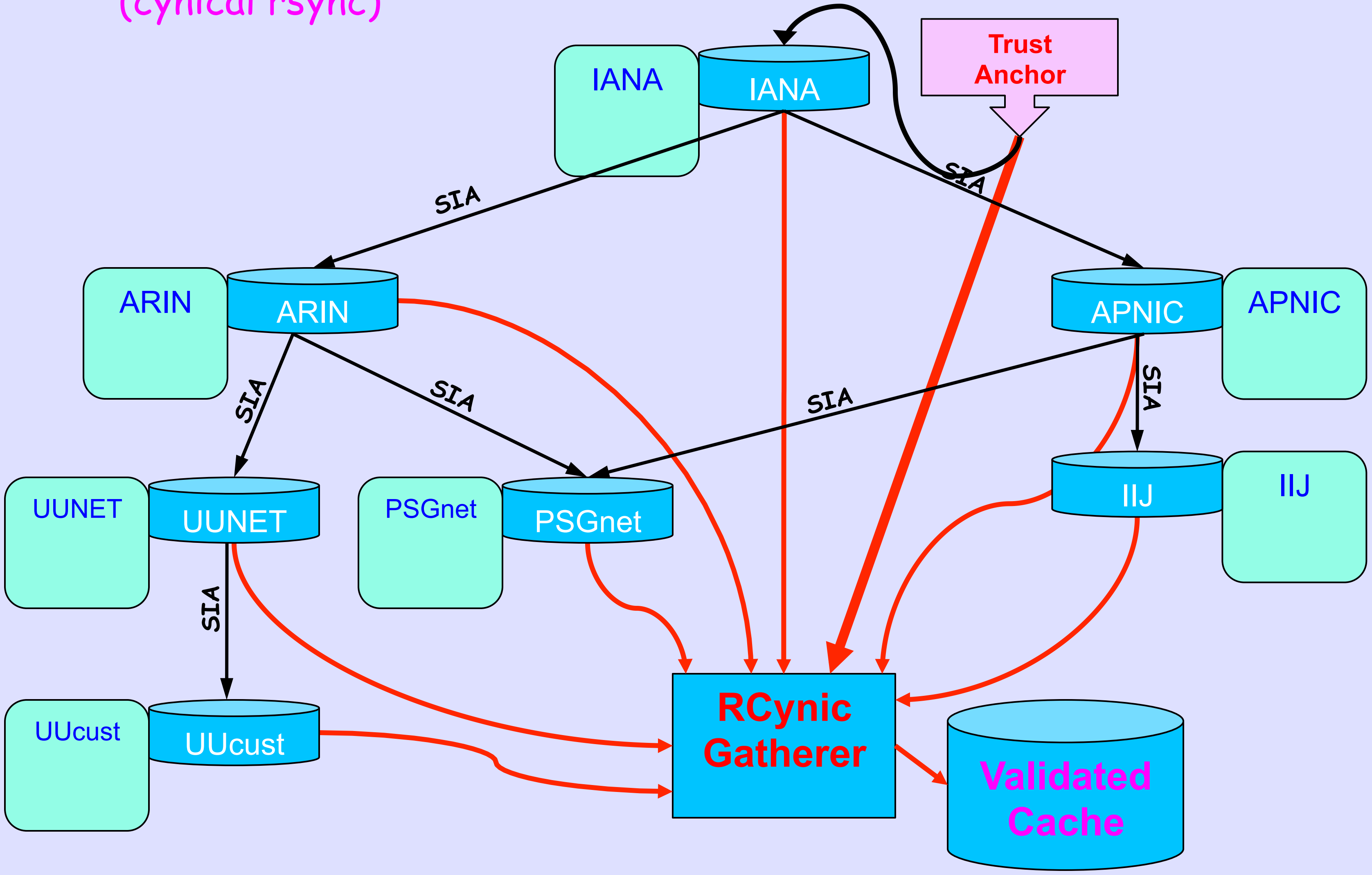
Distributed RPKI DataBase



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

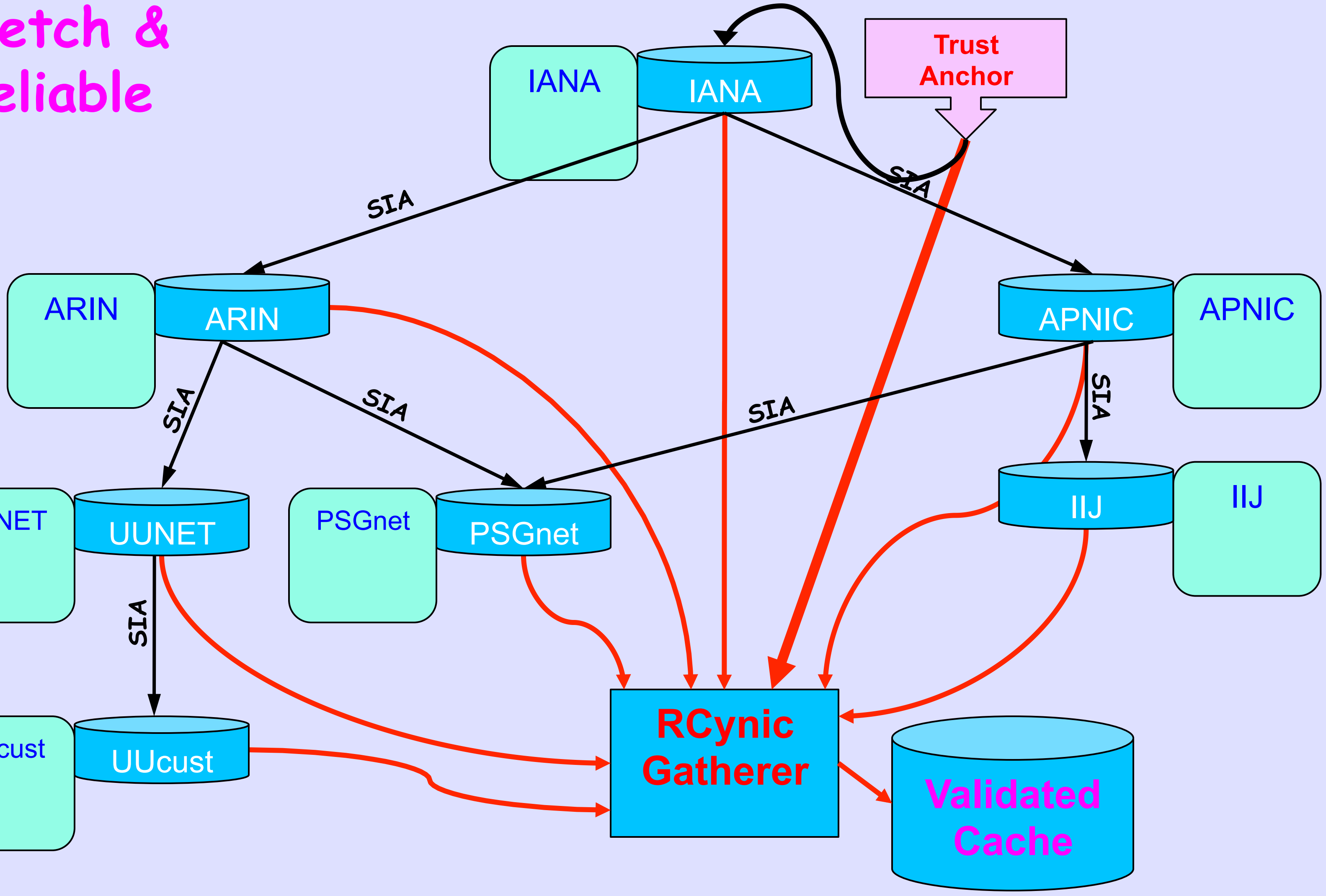
RCynic Cache Gatherer

(cynical rsync)

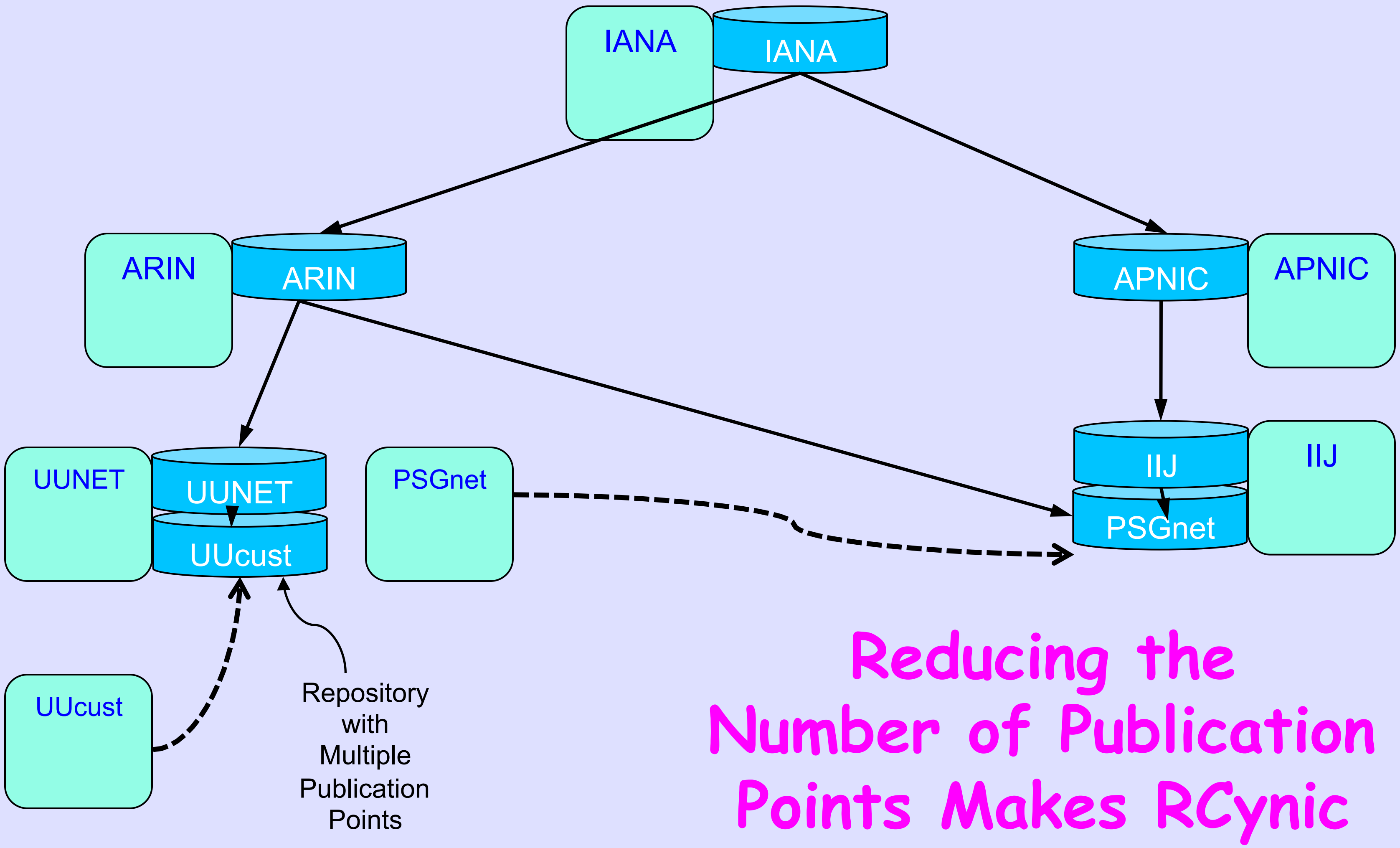


Reliability Issue

Expensive
To Fetch &
Unreliable

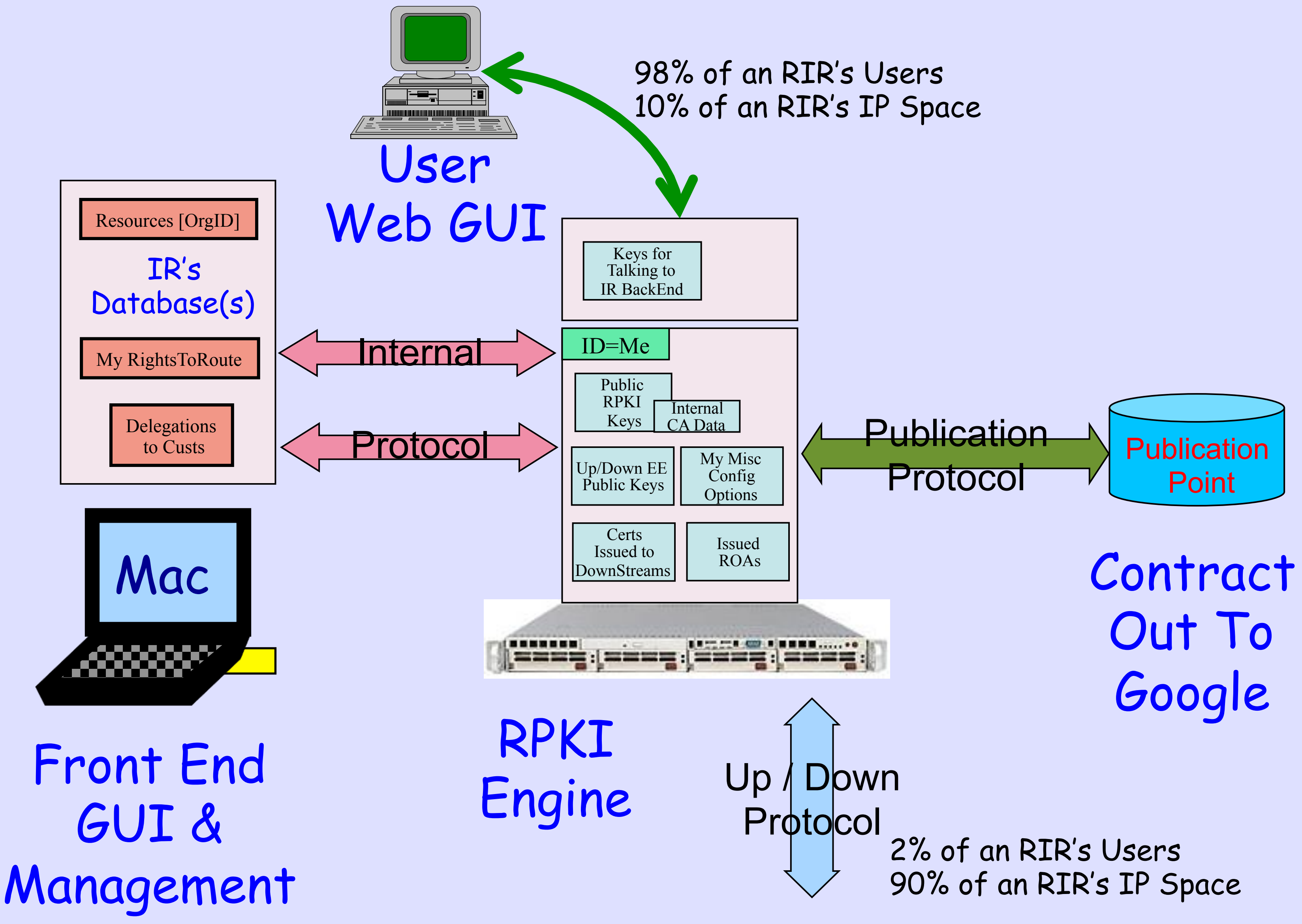


Reliability Via Hosted Publication



Reducing the Number of Publication Points Makes RCynic More Efficient

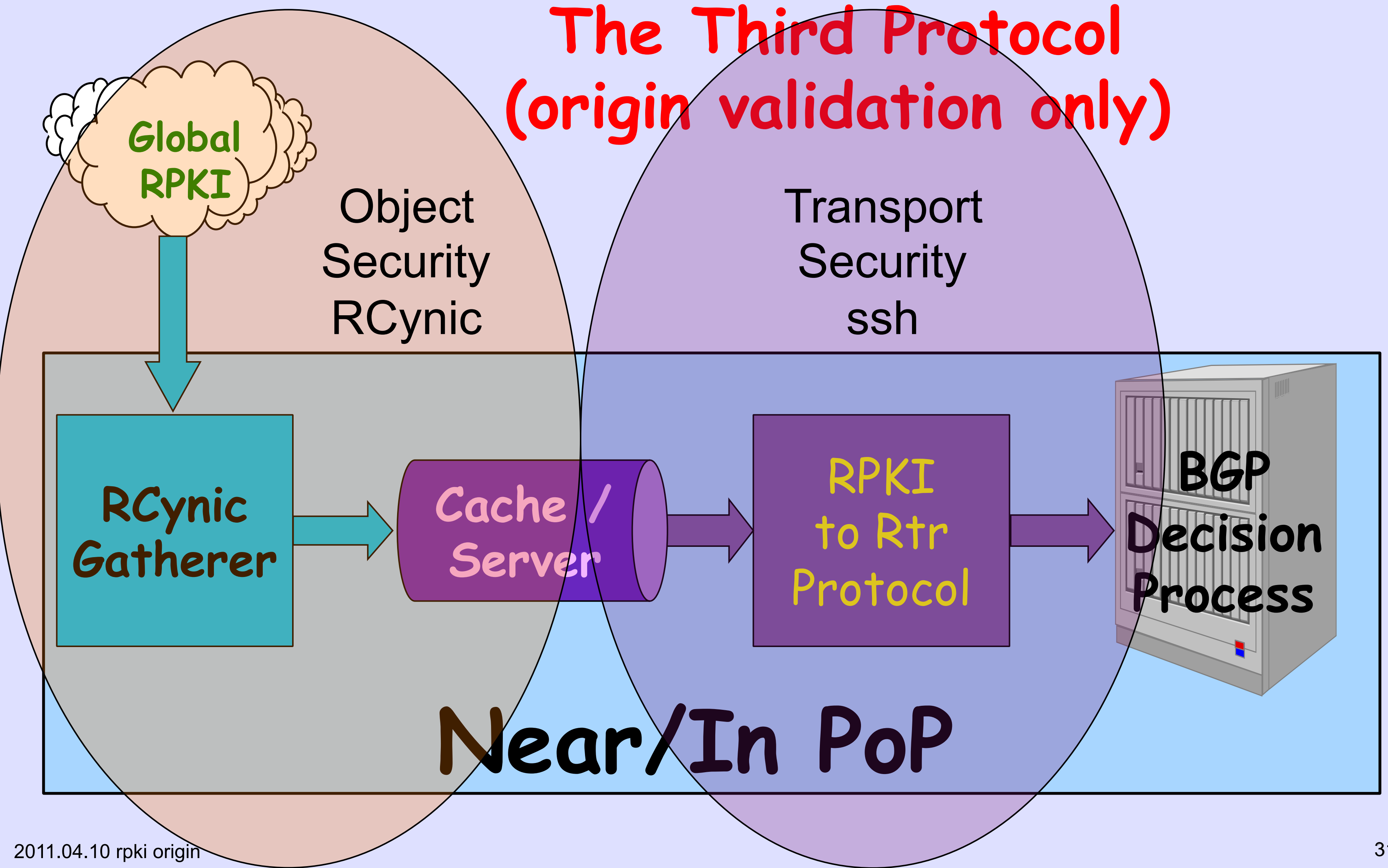
A Usage Scenario



Origin Validation

- 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!**
- Juniper in Alpha this week!

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

Typical Exchange

```
Cache                                Router
    | <----- Reset Query --------- | R requests data
    |                              |
    | ----- Cache Response -----> | C confirms request
    | ------- IPvX Prefix ------> | C sends zero or more
    | ------- IPvX Prefix ------> |   IPv4 and IPv6 Prefix
    | ------- IPvX Prefix ------> |   Payload PDUs
    | ------  End of Data ------> | C sends End of Data
    |                              |   and sends new serial
    ~                              ~
    | -------- Notify ----------> |  (optional)
    |                              |
    | <----- Serial Query -------- | R requests data
    |                              |
    | ----- Cache Response -----> | C confirms request
    | ------- IPvX Prefix ------> | C sends zero or more
    | ------- IPvX Prefix ------> |   IPv4 and IPv6 Prefix
    | ------- IPvX Prefix ------> |   Payload PDUs
    | ------  End of Data ------> | C sends End of Data
    |                              |   and sends new serial
    ~                              ~
```

Reset Query

```
0          8          16          24        31
.-------------------------------------------.
| Protocol |    PDU    |                     |
| Version  |    Type   |   reserved = zero   |
|    0     |     2     |                     |
+-------------------------------------------+
|                                           |
|                 Length=8                  |
|                                           |
`-------------------------------------------'
```

Cache Response

```
0          8          16         24        31
.-------------------------------------------.
| Protocol |   PDU    |                     |
| Version  |   Type   |    reserved = zero  |
|    0     |    3     |                     |
+-------------------------------------------+
|                                           |
|                 Length=8                  |
|                                           |
`-------------------------------------------'
```

IPv4 Prefix

```
0          8          16         24        31
.-------------------------------------------.
| Protocol |   PDU    |                     |
| Version  |   Type   |   reserved = zero   |
|    0     |    4     |                     |
+-------------------------------------------+
|                                           |
|                 Length=20                 |
|                                           |
+-------------------------------------------+
|          |  Prefix  |   Max    |          |
|  Flags   |  Length  |  Length  |   zero   |
|          |   0..32  |   0..32  |          |
+-------------------------------------------+
|                                           |
|                IPv4 prefix                |
|                                           |
+-------------------------------------------+
|                                           |
|         Autonomous System Number          |
|                                           |
`-------------------------------------------'
```

IPv6 Prefix

```
0          8          16         24        31
  .-------------------------------------------.
  | Protocol |    PDU     |                    |
  | Version  |    Type    |  reserved = zero   |
  |    0     |     6      |                    |
  +-------------------------------------------+
  |                                           |
  |                 Length=40                 |
  |                                           |
  +-------------------------------------------+
  |          |   Prefix   |    Max    |        |
  |  Flags   |   Length   |  Length   |  zero  |
  |          |   0..128   |  0..128   |        |
  +-------------------------------------------+
  |                                           |
  +---                                     ---+
  |                                           |
  +---             IPv6 prefix             ---+
  |                                           |
  +---                                     ---+
  |                                           |
  +-------------------------------------------+
  |                                           |
  |         Autonomous System Number          |
  |                                           |
  `-------------------------------------------'
```

End of Data

```
0          8          16         24        31
  .-------------------------------------------.
  | Protocol |   PDU     |                     |
  | Version  |   Type    |   reserved = zero   |
  |    0     |    7      |                     |
  +-------------------------------------------+
  |                                           |
  |                 Length=12                 |
  |                                           |
  +-------------------------------------------+
  |                                           |
  |               Serial Number               |
  |                                           |
  `-------------------------------------------'
```

Notify (Think DNS)

```
0          8          16         24        31
.-------------------------------------------.
| Protocol |   PDU    |                     |
| Version  |   Type   |    reserved = zero  |
|    0     |    0     |                     |
+-------------------------------------------+
|                                           |
|                 Length=12                 |
|                                           |
+-------------------------------------------+
|                                           |
|               Serial Number               |
|                                           |
`-------------------------------------------'
```

Serial Query

```
0          8          16         24        31
  .-------------------------------------------.
  | Protocol |    PDU     |                    |
  | Version  |    Type    |   reserved = zero  |
  |    0     |     1      |                    |
  +-------------------------------------------+
  |                                           |
  |                 Length=12                 |
  |                                           |
  +-------------------------------------------+
  |                                           |
  |               Serial Number               |
  |                                           |
  `-------------------------------------------'
```

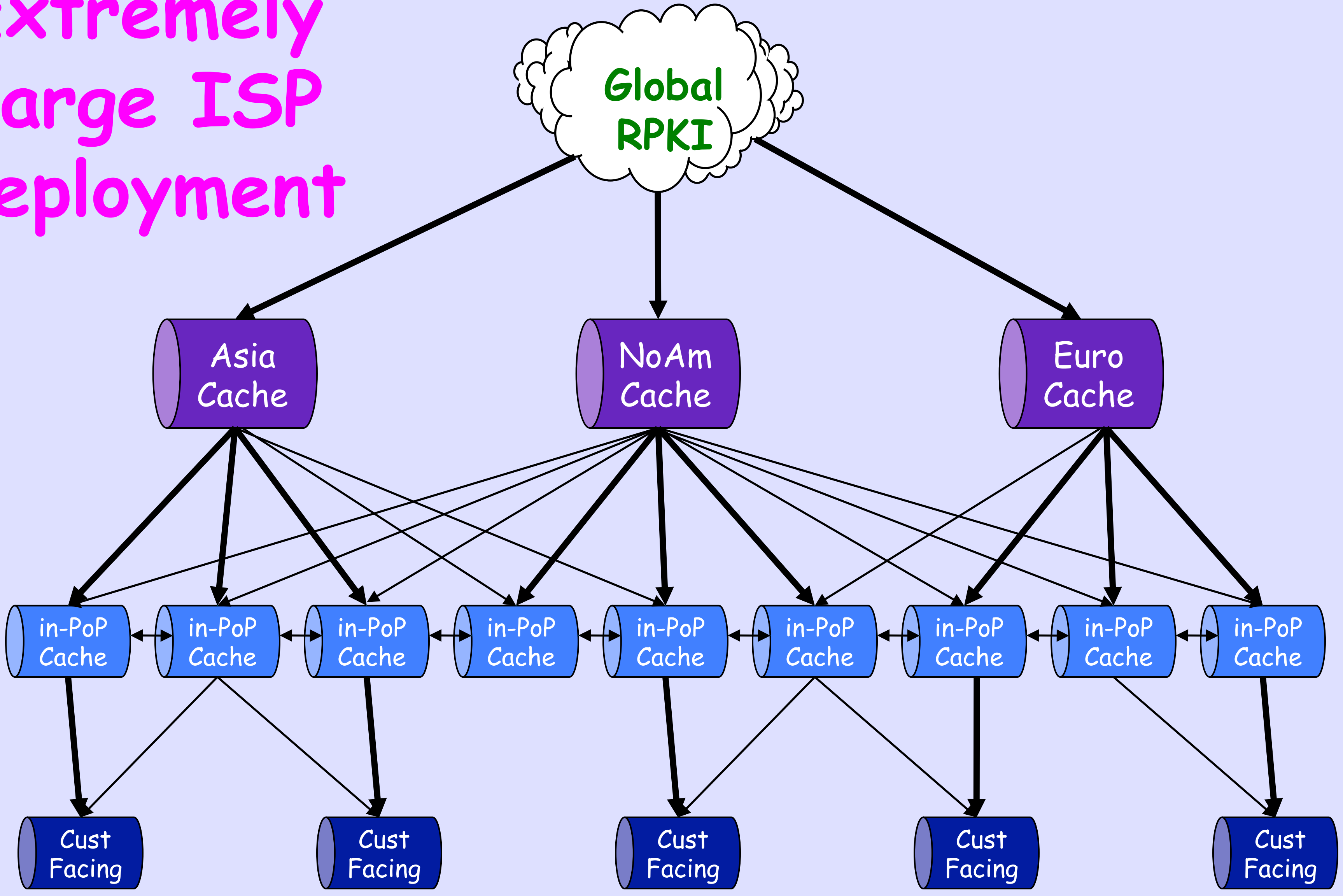
Error Response

```
0          8          16         24        31
.-------------------------------------------.
| Protocol |   PDU    |                     |
| Version  |   Type   |    Error Number     |
|    0     |    10    |                     |
+-------------------------------------------+
|                                           |
|                  Length                   |
|                                           |
+-------------------------------------------+
|                                           |
|       Length of Encapsulated PDU          |
|                                           |
+-------------------------------------------+
|                                           |
~           Copy of Erroneous PDU           ~
|                                           |
+-------------------------------------------+
|                                           |
|           Length of Error Text            |
|                                           |
+-------------------------------------------+
|                                           |
|              Arbitrary Text               |
|                    of                     |
~         Error Diagnostic Message          ~
|                                           |
`-------------------------------------------'
```

Changing Caches

- Running on cache A happily
- A goes bad (A down, sends error, ...)
- Router decides to break off relationship with A
- Router keeps using old data from A
- Router tries other caches in priority order
- Router starts to load from B, in a separate buffer, but still runs on old data from A
- Router finishes loading data from B
- Router flushes all data from A and installs all data from B
- Router reevaluates installed prefixes against new data

Extremely Large ISP Deployment



━━ High Priority

── Lower Priority

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

Prefix validation logic

```
query key = <BGP destination, masklen>, data = origin AS
result = BGP_PFXV_STATE_NOT_FOUND
walk prefix validation table to look for the query key
for each matched “entry” node in prefix validation table,
    prefix_exists = TRUE
    walk all records with different maxLength values
    for each “record” within range (query masklen <= maxLength)
        if query origin AS == record origin AS
            result = BGP_PFXV_STATE_VALID
            return (result)
        endif
    endfor
endfor
if prefix_exists == TRUE,
    result = BGP_PFXV_STATE_INVALID
endif
return (result)
```

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
```

Defaults

- Origin Validation is Enabled if you have configured a cache server peering

- RPKI Poll Interval is 30 Minutes

- No Effect on Policy unless you have configured it

An ISP's ROAs

```
# <prefix>/<length>-<maxlength> <asn> <group>
#
64.9.224.0/19-24      15169    ARIN
74.125.0.0/16-24      15169    ARIN-3
72.14.192.0/18-2415169    ARIN-3
72.14.224.0/24-2436384    ARIN-3
72.14.230.0/24-2436384    ARIN3
64.233.160.0/19-24    15169    ARIN-3
64.9.224.0/19-24      36492    ARIN
66.102.0.0/20-24      15169    ARIN-3
66.249.64.0/19-2415169    ARIN-3
66.249.80.0/20-2415169    ARIN-3
72.14.192.0/18-2415169    ARIN-3
74.125.0.0/16-24      15169    ARIN-3
173.194.0.0/16-2415169    ARIN-3
209.85.128.0/17-24    15169    ARIN-3
216.239.32.0/19-24    15169    ARIN-3
2001:4860::/32-6415169    ARIN-3
```

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
...
```

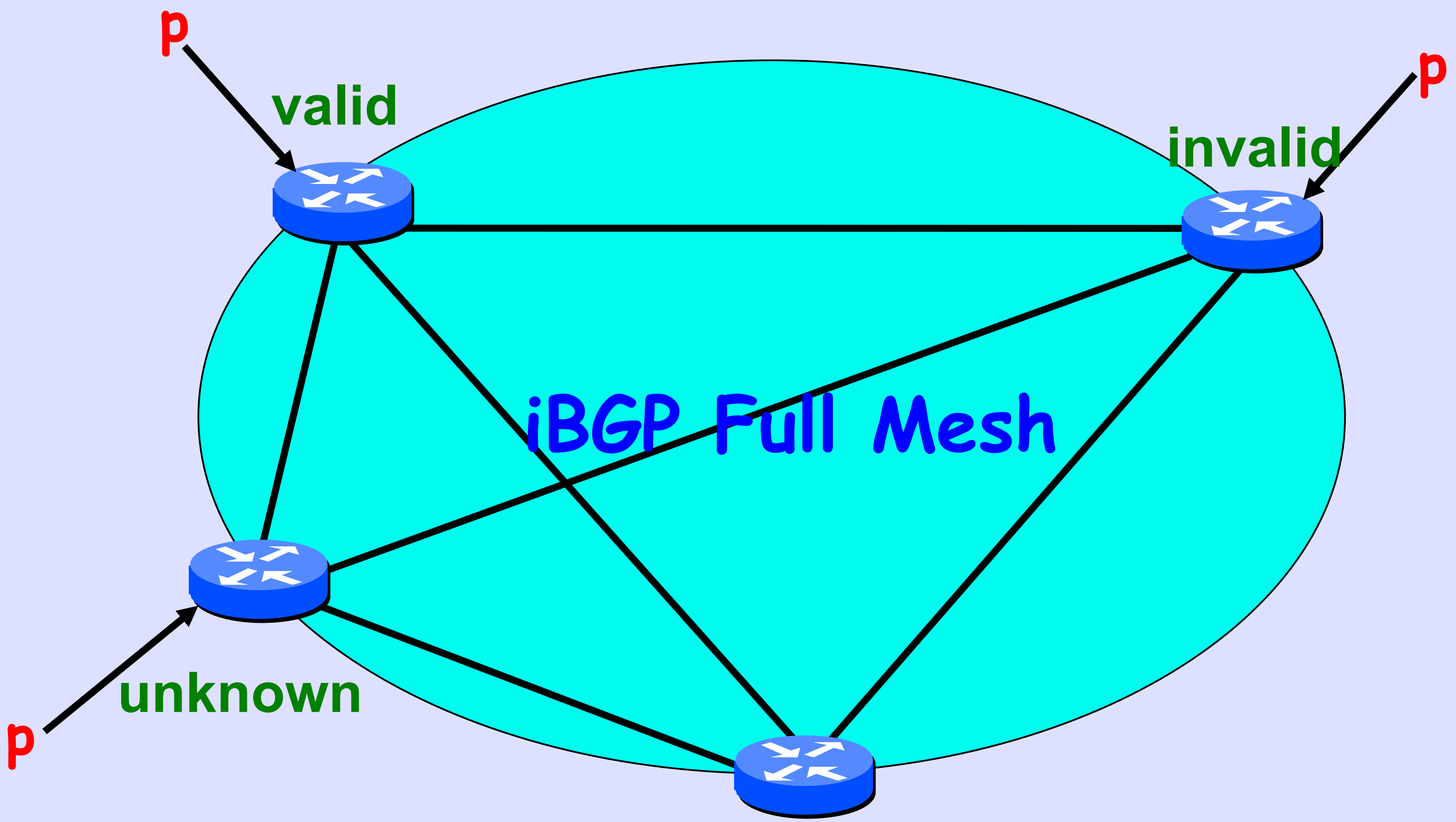
Bad Dog!

```
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
```

iBGP Hides Validity State



which do i choose?
why do i choose it?

Unknown Beat Valid!

```
r1.iad#sh ip bg 198.180.152.0
BGP routing table entry for 198.180.152.0/24, version 324176
Paths: (2 available, best #1, table default)
  Not advertised to any peer
  2914 4128
    129.250.10.157 (metric 1) from 198.180.150.253
  (198.180.150.253)
      Origin IGP, metric 51, localpref 100, valid, internal, best
      Community: 2914:410 2914:2000 2914:3000 3927:380
  1239 2914 4128
    144.232.18.81 from 144.232.18.81 (144.228.241.254)
      Origin IGP, metric 0, localpref 100, valid, external
      Community: 3927:380
      Sovc state valid
```

MED Beat Valid

```
r1.iad#sh ip bg 147.28.0.0
BGP routing table entry for 147.28.0.0/16, version 142233
Paths: (2 available, best #1, table default)
  Not advertised to any peer
  2914 3130
    129.250.10.157 (metric 1) from 198.180.150.253
  (198.180.150.253)
      Origin IGP, metric 105, localpref 100, valid, internal, best
      Community: 2914:410 2914:2000 2914:3000 3927:380
  1239 3130
    144.232.18.81 from 144.232.18.81 (144.228.241.254)
      Origin IGP, metric 653, localpref 100, valid, external
      Community: 3927:380
      Sovc state valid
```

The Solution

is to

Allow Operator to

Test and then

Set Local Policy

Fairly Secure

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

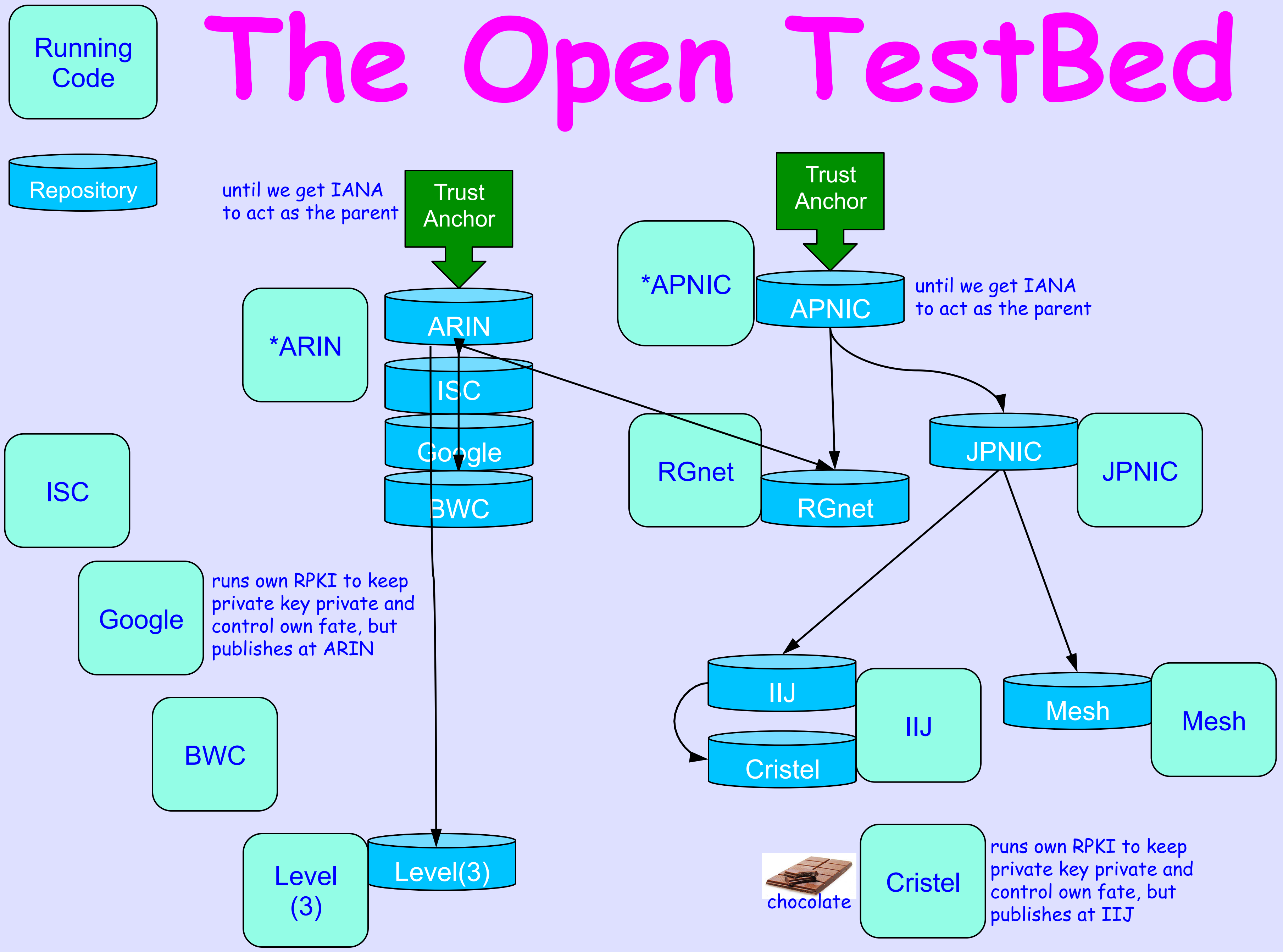
Paranoid

```
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
route-map validity-2
   set metric 100
```

The Open TestBed



until we get IANA to act as the parent

until we get IANA to act as the parent

runs own RPKI to keep private key private and control own fate, but publishes at ARIN

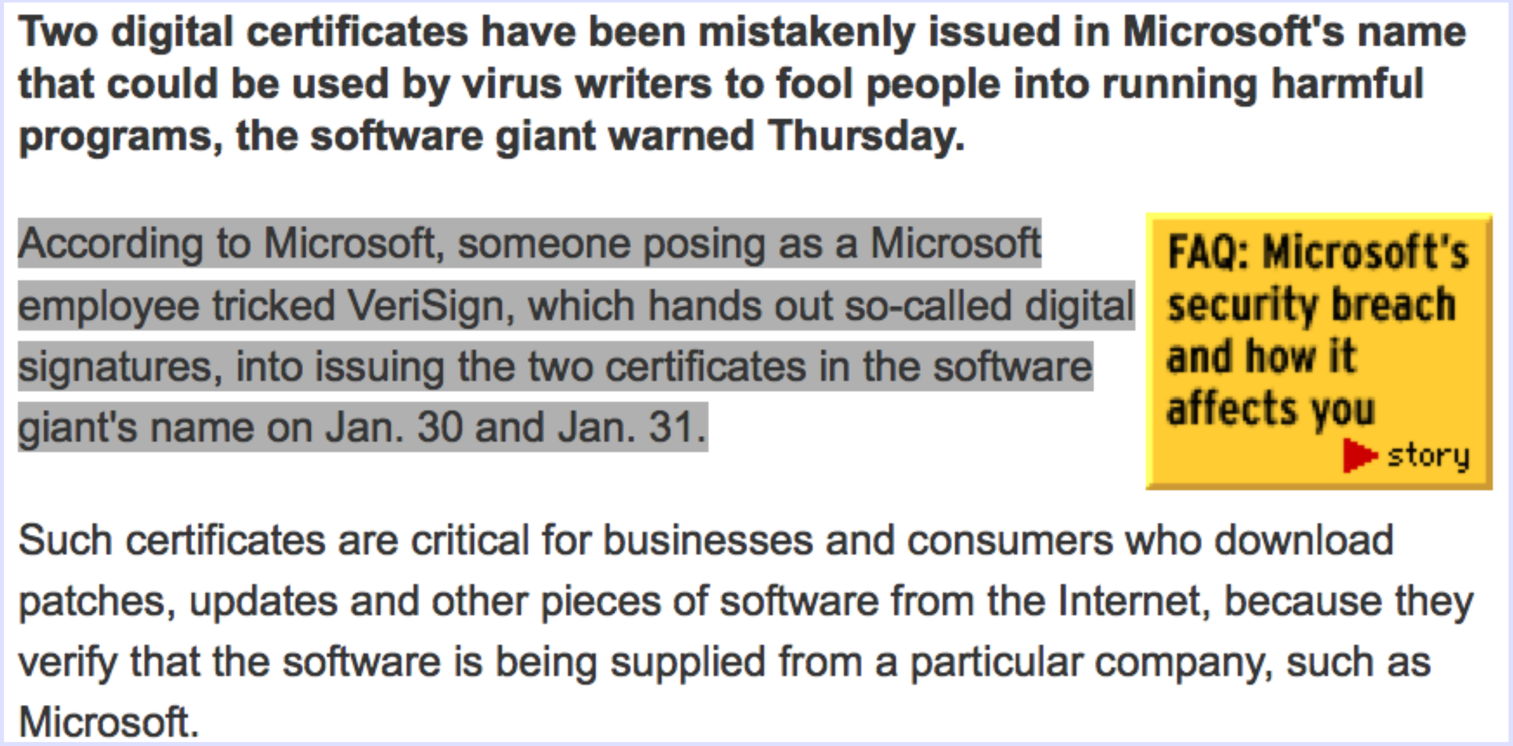
runs own RPKI to keep private key private and control own fate, but publishes at IIJ

* APNIC and ARIN are simulations constructed from public data

The Big Speedbump

"We will never control Internet routing"
-- The RIRs
Certificate Expiration Revocation

But Who Do We Trust?



Two digital certificates have been mistakenly issued in Microsoft's name that could be used by virus writers to fool people into running harmful programs, the software giant warned Thursday.

According to Microsoft, someone posing as a Microsoft employee tricked VeriSign, which hands out so-called digital signatures, into issuing the two certificates in the software giant's name on Jan. 30 and Jan. 31.

Such certificates are critical for businesses and consumers who download patches, updates and other pieces of software from the Internet, because they verify that the software is being supplied from a particular company, such as Microsoft.

http://news.cnet.com/2100-1001-254586.html

Open Source (BSD Lisc) Running Code

https://rpki.net/

Test Code in Routers

Talk to C & J

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). [0]
- ARIN
- Internet Initiative Japan
- Cisco, Google, NTT, Equinix

[0] – we take your scissors away and turn them into plowshares