RPKI Tools From Soup to Nuts

Rob Austein <sra@isc.org>
Randy Bush <randy@psg.com>
Steve Bellovin <smb@cs.columbia.edu>
Michael Elkins <Michael.Elkins@cobham.com>
. . . and a lot of help from our friends

APNIC 30, 24 August 2010

Why We're Doing This

For us, this is all about inter-domain routing security

The tools we build are a means to that end

Right now, routing security consists of

- Knowing your neighbors
- Filtering based on rumors and poorly validated data

Right now, most of the threats are fat fingers

- Some real attacks, but so far they're still rare
- This will change
- Monkey-in-the-middle attacks on TCP used to be considered hard—now we call them "NAT"
- The "Waiting for Kaminsky" deployment model

What We're Trying To Do About It

Use RPKI data to formally validate some of BGP's inputs

- Origin validation code is engineering now, could deploy in next few years but requires production RPKI
- Path validation is still research, but we know some of what it needs

Origin Validation On The Router

What does router really need to know for origin validation?

"Holder of prefix X authorizes origination from ASN Y"

Validation of prefix X on ASN Y can return three states

Valid	Authorization found for X on Y
Invalid	Authorization found for X and Y doesn't have it
Unknown	No authorization data found for X

Origin Validation On The Router

Router just needs a trusted source of these data

Some other owned and trusted box in the POP can do the crypto; router just needs secure channel to that box

This turns out to be really cheap

- Router's part of it runs on current hardware
- The "other box" can be a cheap rackmount PC
- Software for the "other box" is free :)

Origin Validation Implementation Status

Specification

draft-ietf-sidr-rpki-rtr

Implementations

- Open source server and sample client available
- (Test) client running in IOS and IOS-XR
- Other router vendors working on implementations

Caches used to feed servers can be

- Simple stand-alone cache in each POP
- Tree or mesh of caches, rsyncing with each other
- Router doesn't care, server handles all this

Example: “Secure” Configuration

Plan

- Drop routes with status “invalid”
- Downpref routes with status “unknown”
- Default preference for status “valid”

Configuration

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

Example: “Paranoid” Configuration

Plan

- Prefer routes with status “valid”
- Drop routes with status other than “valid”

Configuration

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

Example: “Smaller Hammer” (After AS-Path)

Plan

- Tweak “metric” rather than “localpref”
- Use metric 100 for status “valid,” 50 for status “unknown,” 25 for status “invalid”

Configuration

```
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
```

Some People Like IRR

Some operators have deployed infrastructure using IRR

- We want to upgrade them to RPKI as data source
- This is easy, just feed validated data to irrd

What the heck, let's make it available via WHOIS too

```
$ whois -h whois.rpki.net 198.180.150.1
route:      198.180.150.0/24
descr:      198.180.150.0/24-24
origin:     AS3927
notify:     irr-hack@rpki.net
mnt-by:     MAINT-RPKI
changed:    irr-hack@rpki.net 20100706
source:     RPKI
```

RPKI Validation: “rcynic”

“Cynical rsync”

- Recursive tree walk starting from trust anchor(s)
- Uses rsync to fetch objects, walks manifests to find children
- Checks all the X.509 details including RFC 3779 rules
- Performs object-specific checks on certificates, CRLs, ROAs, manifests

Result is validated cache of current global RPKI state

- Preserves “fall-back cache” from previous runs
- Cache can be seeded from other caches
- All data still validated locally each cycle

rcynic In Action: Summary Listing

rcynic summary 2010-08-13T04:18:03Z

Publication Repository	Current certificates accepted	Current certificates rejected	Current CRLs accepted	Current Manifests accepted	Current Manifests rejected	rsync transfers failed	rsync transfers succeeded	Stale CRLs	Invalid manifest certificates	Stale manifests	Invalid ROA certificates	Current ROAs accepted	Current ROAs rejected
rpki.apnic.net	318	14	317	317	2		319				10	28	10
arin.rpki.net	33		8	8		1	4	2		2		51	
rpki-pilot.arin.net:10873	23	1	24	24			24					4	
rgnet.rpki.net	16		12	12			5	11		11		15	
certrepo.ripe.net	8		9	9			10					8	
rrs.research.icann.org			1		1		1	1	1	1			
apnic.rpki.net	1		1	1			2						
ripe.rpki.net	1		1	1			2						
nets-xserve.lboro.ac.uk						3							
ts2.antd.nist.gov						2							
44-141.antd.nist.gov						1							
rpki.antd.nist.gov						1							
terrain1.antd.nist.gov						1							
ts1.antd.nist.gov						1							
Total	**400**	**15**	**373**	**372**	**3**	**10**	**367**	**14**	**1**	**14**	**10**	**106**	**10**

rcynic In Action: Problem Listing

Problems

Status	URI
certificate has expired	rsync://rpki.apnic.net/repository/A3C38A24D60311DCAB08F31979BDBE39/LrcQPPxKmZ_9X9-eYlg6q9gIraE.cer
certificate has expired	rsync://rpki.apnic.net/repository/A3C38A24D60311DCAB08F31979BDBE39/fqjWBet-TmZtdwvOXrYjox43ZZo.cer
certificate has expired	rsync://rpki.apnic.net/repository/A3C38A24D60311DCAB08F31979BDBE39/tQttmQNdmUubgjcfAZAzJDS-1os.cer
certificate has expired	rsync://rpki.apnic.net/repository/A3C38A24D60311DCAB08F31979BDBE39/J6vZWCVm8ehuxAyxmeWOJiLGvYw.cer
certificate has expired	rsync://rpki.apnic.net/repository/A3C38A24D60311DCAB08F31979BDBE39/8zBriTiNug9ZuNF6oigUMdTJgkk.cer
certificate has expired	rsync://rpki.apnic.net/member_repository/A91893D1/7917259278E611DEADE3059B864992D1/04826FDE7BBB11DEA01C8695864992D1.roa
certificate has expired	rsync://rpki.apnic.net/member_repository/A91893D1/7917259278E611DEADE3059B864992D1/04826FDE7BBB11DEA01C8695864992D1.roa
certificate has expired	rsync://rpki.apnic.net/repository/A3C38A24D60311DCAB08F31979BDBE39/R7Olq0tjrpKnwWPnF_cfrKk97q4.cer
certificate has expired	rsync://rpki.apnic.net/repository/A3C38A24D60311DCAB08F31979BDBE39/F49mvoPEGBpyS3BXori6XFz-1bQ.cer
certificate has expired	rsync://rpki.apnic.net/member_repository/A91872ED/06A83982887911DD813F432B2086D636/49454FBA227411DE9073BEA2864992D1.roa
certificate	

rcynic In Action: Detail Listing

Validation Status

Timestamp	Status	URI
2010-08-13T03:54:50Z	OK	rsync://rpki.apnic.net/repository/8BDFC7DED5FD11DCB14CF4B1A703F9B7/KR8WDankLJ7uq4RBE212svl-C0A.crl
2010-08-13T03:54:50Z	OK	rsync://rpki.apnic.net/repository/8BDFC7DED5FD11DCB14CF4B1A703F9B7/KR8WDankLJ7uq4RBE212svl-C0A.mft
2010-08-13T03:54:50Z	OK	rsync://rpki.apnic.net/repository/8BDFC7DED5FD11DCB14CF4B1A703F9B7/CVPQSgUkLy7pOXdNeVWGvnFX_0s.cer
2010-08-13T03:54:54Z	OK	rsync://rpki.apnic.net/repository/A3C38A24D60311DCAB08F31979BDBE39/CVPQSgUkLy7pOXdNeVWGvnFX_0s.crl
2010-08-13T03:54:54Z	OK	rsync://rpki.apnic.net/repository/A3C38A24D60311DCAB08F31979BDBE39/CVPQSgUkLy7pOXdNeVWGvnFX_0s.mft
2010-08-13T03:54:54Z	OK	rsync://rpki.apnic.net/repository/A3C38A24D60311DCAB08F31979BDBE39/yIOrpoPMPjV3jJeY_aKc9szo7L0.cer
2010-08-13T03:54:57Z	OK	rsync://rpki.apnic.net/member_repository/A9197ADE/6556B78AFEF611DD8CEF7AF7864992D1/yIOrpoPMPjV3jJeY_aKc9szo7L0.crl
2010-08-13T03:54:57Z	OK	rsync://rpki.apnic.net/member_repository/A9197ADE/6556B78AFEF611DD8CEF7AF7864992D1/yIOrpoPMPjV3jJeY_aKc9szo7L0.mft
2010-08-13T03:54:57Z	OK	rsync://rpki.apnic.net/repository/A3C38A24D60311DCAB08F31979BDBE39/7QK71DUarLU1RjuhYSaPx-Q6DhQ.cer
2010-08-13T03:54:59Z	OK	rsync://rpki.apnic.net/member_repository/A91B968F/3EA2C59AE06811DD92C1BDA8864992D1/7QK71DUarLU1RjuhYSaPx-Q6DhQ.crl
2010-08-13T03:54:59Z	OK	rsync://rpki.apnic.net/member_repository/A91B968F/3EA2C59AE06811DD92C1BDA8864992D1/7QK71DUarLU1RjuhYSaPx-Q6DhQ.mft
2010-08-13T03:54:59Z	OK	rsync://rpki.apnic.net/repository/A3C38A24D60311DCAB08F31979BDBE39/mOZ0ark6ICxQQSL4AUgXdb2bet8.cer
2010-08-13T03:55:02Z	OK	rsync://rpki.apnic.net/member_repository/A9155B02/9B0A24E224D511DFA46294EB468B6C5A/mOZ0ark6ICxQQSL4AUgXdb2bet8.crl
2010-08-13T03:55:02Z	OK	rsync://rpki.apnic.net/member_repository/A9155B02/9B0A24E224D511DFA46294EB468B6C5A/mOZ0ark6ICxQQSL4AUgXdb2bet8.mft
2010-08-13T03:55:02Z	OK	rsync://rpki.apnic.net/repository/A3C38A24D60311DCAB08F31979BDBE39/AemlOulb0kicSnMH00zpD4Hu9Gw.cer
2010-08-13T03:55:07Z	OK	rsync://rpki.apnic.net/member_repository/A91E51AD/468BE340E2D411DDA0B1DAD5864992D1/AemlOulb0kicSnMH00zpD4Hu9Gw.crl
2010-08-13T03:55:07Z	OK	rsync://rpki.apnic.net/member_repository/A91E51AD/468BE340E2D411DDA0B1DAD5864992D1/AemlOulb0kicSnMH00zpD4Hu9Gw.mft
2010-08-13T03:55:07Z	OK	rsync://rpki.apnic.net/repository/A3C38A24D60311DCAB08F31979BDBE39/CGdkCJN3JUg7eRO0ZJszQLIc2_g.cer
2010-08-13T03:55:11Z	OK	rsync://rpki.apnic.net/member_repository/A91A7381/8FE4327A5F9F11DE8CAE667A864992D1/CGdkCJN3JUg7eRO0ZJszQLIc2_g.crl
2010-08-13T03:55:11Z	OK	rsync://rpki.apnic.net/member_repository/A91A7381/8FE4327A5F9F11DE8CAE667A864992D1/CGdkCJN3JUg7eRO0ZJszQLIc2_g.mft
2010-08-13T03:55:11Z	OK	rsync://rpki.apnic.net/repository/A3C38A24D60311DCAB08F31979BDBE39/tht7MGwYmoOVo96QKkpO9V1SdBU.cer
2010-08-13T03:55:15Z	OK	rsync://rpki.apnic.net/member_repository/A91F12B4/B6C39E4CC25311DDB54A35C5864992D1/tht7MGwYmoOVo96QKkpO9V1SdBU.crl
2010-08-13T03:55:15Z	OK	rsync://rpki.apnic.net/member_repository/A91F12B4/B6C39E4CC25311DDB54A35C5864992D1/tht7MGwYmoOVo96QKkpO9V1SdBU.mft
2010-08-13T03:55:15Z	certificate has expired	rsync://rpki.apnic.net/repository/A3C38A24D60311DCAB08F31979BDBE39/LrcQPPxKmZ_9X9-eYlg6q9gIraE.cer
2010-08-13T03:55:15Z	OK	rsync://rpki.apnic.net/repository/A3C38A24D60311DCAB08F31979BDBE39/zBNnA5I8y8M03Hlv1D-fJndsw1Q.cer
2010-08-13T03:55:17Z	OK	rsync://rpki.apnic.net/member_repository/A917D3A1/AB4B227A599111DF94E3F5C955AFE7CF/zBNnA5I8y8M03Hlv1D-fJndsw1Q.crl

Need To Roll Your Own Tools?

RFC 3779 support in OpenSSL since 2007

- Development funded by ARIN in 2006
- Contributed to OpenSSL project
- Hooks directly into OpenSSL validation code
- Enabled by default on some platforms
- Easy to enable if not yet the default on yours

But Where Do We Get The RPKI Data?

Our guess at the breakdown of who will do what

- 98% of resource holders will want IANA/RIRs/NIRs to do their RPKI work for them
- But that's probably about 10% of the address space
- The other 2% of resource holders are the big ISPs who account for 90% of the address space
- We think they'll want to hold their own keys and do RPKI for themselves

But Where Do We Get The RPKI Data?

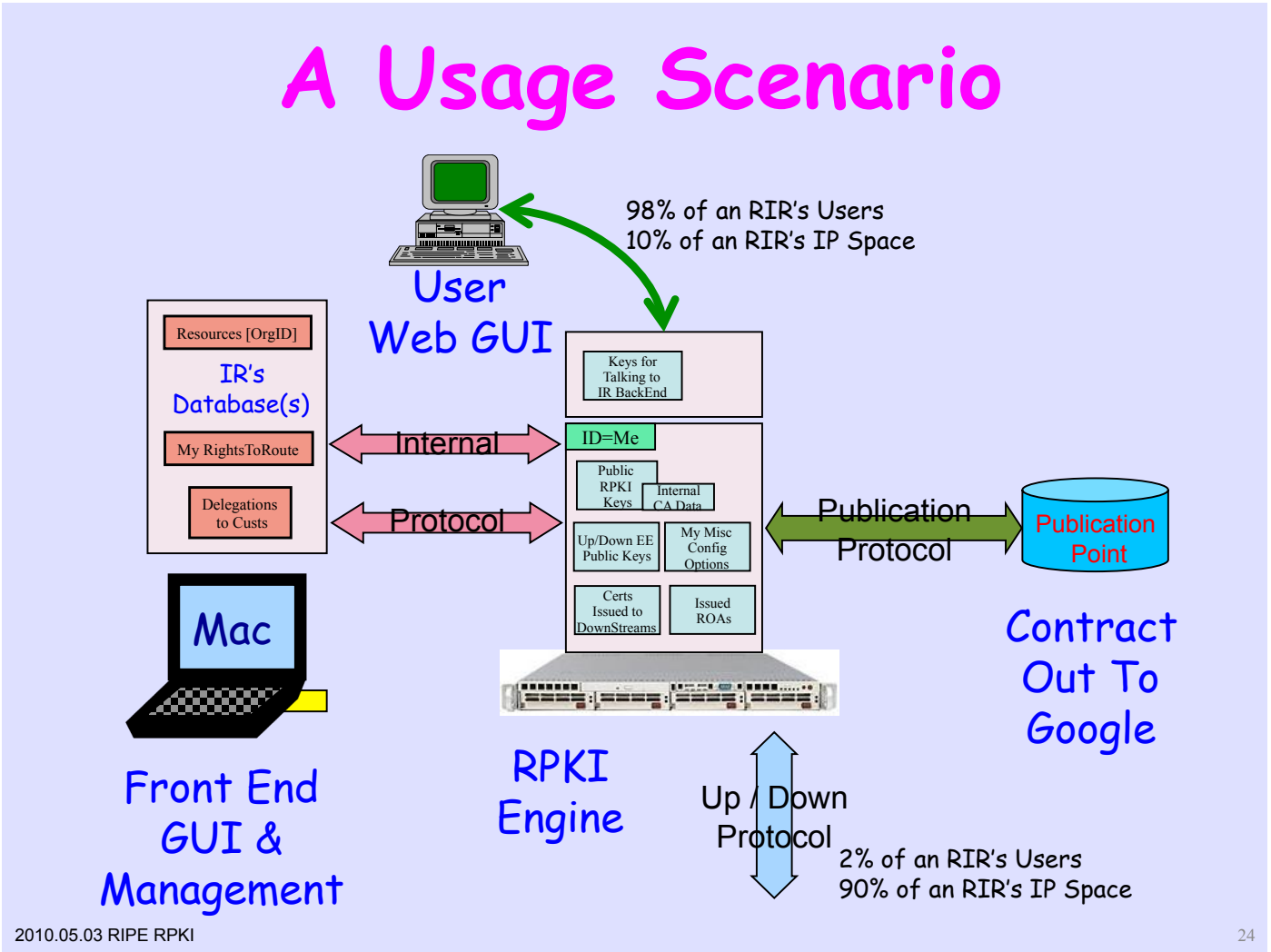
RIRs are mostly focused on serving their own members

- That is, after all, what their members pay them to do
- Which includes serving a lot of those 98% of resource holders directly

We're concentrating on the big ISPs

- Who we think will want to run their own engines
- Who may (or may not) want to run own publication sites

RPKI Production: The Big Picture



RPKI Production: rpkid

Core program for generating and maintaining RPKI objects

- Manages certificates, CRLS, ROAs, manifests
- Client and server for provisioning ("up-down") protocol
- Client for publication protocol
- Supports hosting (single rpkid instance, multiple entities)

Independent of back-end operation (database, BPKI)

- Speaks internal ("left-right") protocol to back-end
- Sample back-end with GUI provided, feel free to roll your own

What rpkid Does

Basic tasks

- Get resource certificates from parent(s)
- Issue resource certificates to children
- Generate ROAs for self
- Generate support objects (CRLs, manifests)
- Regenerate, revoke, or clean up objects as needed
- Publish outputs via publication protocol

pubd, And Why We Need It

Separate program ("pubd") to handle object publication

- Server for publication protocol
- Just does what rpkid asks, after access control checks

Why is this separate from rpkid?

- Different security constraints: rpkid holds RPKI private keys, pubd does not
- Different availability constraints
- Consolidating publication sites is better for everybody
- Outsourcing publication looks like win-win in many cases

Sample Back End

rpkid and pubd are intended to be portable

- Back-end code tends to be highly non-portable
- So rpkid and pubd operate at arms length from back-end code, via defined protocols

We supply a customizable set of back-end tools

- Use them if they fit your operation
- Modify or replace them if they don't

Sample Back End

Command line tool

- To set up parent/child/repository relationships
- To handle CMS authentication keys and certificates
- To support various hosting models

GUI (Django-based web interface)

- Simple interface for resource administration
- Monitor status of received resources

Back-end tools use simple text-based transfer format

- Which you can generate from SQL
- Or spreadsheet
- Or Python (AWK, Perl, TECO, . . .) script

Control Panel

Handle: Alice | export identity | select

Logged in as sra | admin | Log Out

Parents

- RIR

Accepted Resource	Not Before	Not After
ASN 64533	2010-08-13 14:17:21	2011-08-13 14:16:52
10.0.0.0/8	2010-08-13 14:17:21	2011-08-13 14:16:52

Children

- Betty

  Delegated resources:

  - ASN 64533
  - 10.0.0.0/8

My ROA [request]s

Prefix	ASN

Control Panel

Unallocated Resources

-- none --

Allocation View

Handle: Alice

Logged in as sra | admin | Log Out

ASN View

ASN:	ASN 64533
Received from:	RIR
Validity:	2010-08-13 14:17:21 - 2011-08-13 14:16:52
Allocated:	Betty

Edit

Child: Alice's child Betty

Submit Query

Action: give to child

Parent/Child/Repository Relationships

Discovered we needed out-of-band setup protocol

So we wrote yet another little protocol, tried to keep it simple

Two stages

1. Parent/child setup dance
2. Publication repository setup dance

Happens in this order so parent can give hints to child

Parent/Child Setup Dance

Child

- “I call myself . . . ”
- “My public key is . . . ”

Parent

- “I call myself . . . ”
- “I call you . . . ”
- “My public key is . . . ”
- “Your service URL with me is . . . ”
- And one of:
  - “You can publish with me” (offer)
  - “I publish at . . . , maybe you should too” (referral)

Child/Repository Setup Dance

Note

This stage is optional, child might run its own pubd

Child

- "I call myself . . . "
- "My public key is . . . "
- "My parent is . . . "
- And maybe:
  - "Joe sent me"

Repository

- "My public key is . . . "
- "Your service URL with me is . . . "
- "Your publication location is . . . "

Where To Get The Code

Open source implementation

`http://www.rpki.net/`

Thanks to

- ARIN (initial funding)
- DHS (current funding)
- IIJ
- Cisco
- Google
- NTT
- Equinix

Questions?