Adventures in RPKI (non) deployment

Wes George
wesley.george@twcable.com @wesgeorge

Time Warner Cable | ENJOY BETTER

Background

March 2013 FCC CSRIC III WG 6 report on Secure BGP

- Accurate Records, better measurements
- "Cautious, staged deployment of RPKI Route Origin Validation"

It became my job to figure out how to do that at TWC

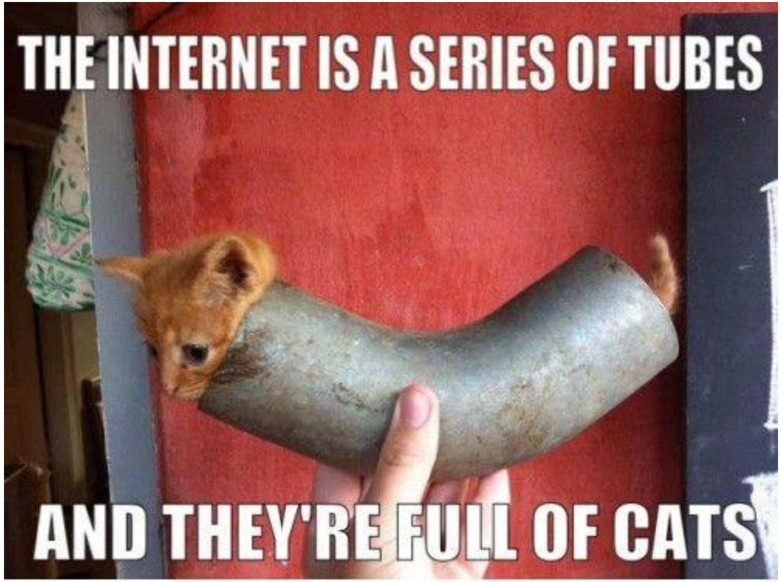
This is not:

- Another "deploying ROV is easy, you should all do it" presentation
- A presentation suggesting ROV is not deployable

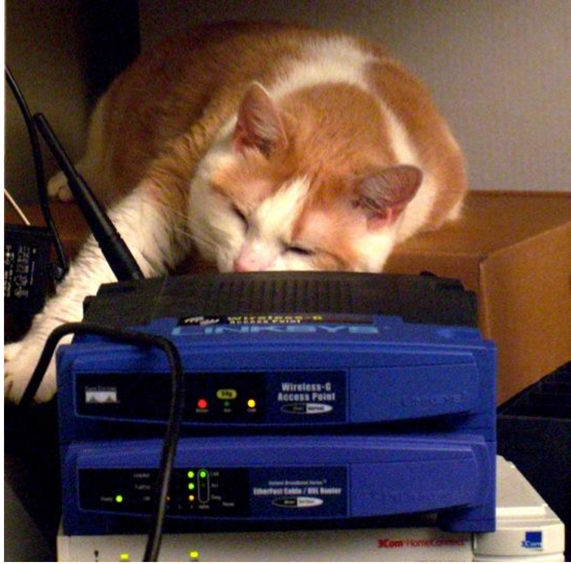
Why NANOG?

This presentation is:

- One guy’s experiences trying to deploy ROV at one ISP
- An attempt to highlight some operational challenges for large scale ROV deployment
  - Internal stuff probably common among big companies
  - External issues, tooling
- A cat (and occasionally dog)-enhanced presentation



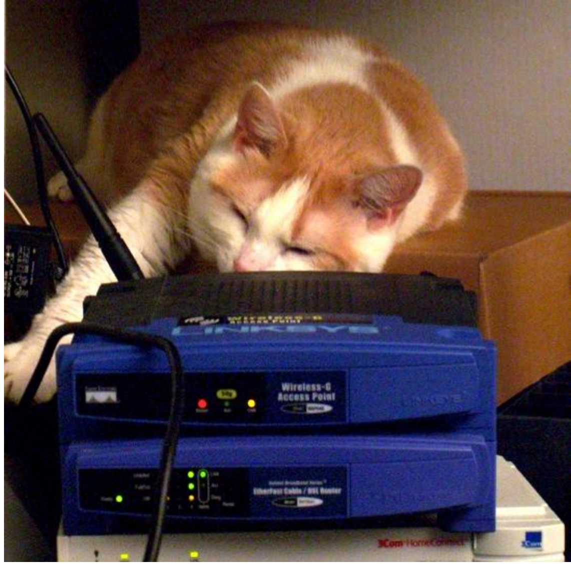
We’ve seen this movie before…



Rolling any new security feature out is hard

- Risk vs reward
  - Compare cost of deploying to cost of doing nothing
    - Cost = liability incurred, money, time, capacity, etc
    - Have I already experienced this attack? Cost?
      - If not, what’s the risk that I will in the near future? Cost?
      - How much risk if I wait {6,12,24} months to deploy?

We’ve seen this movie before…



Rolling any new security feature out is hard

- First Mover problem
  - Without tangible immediate benefit to incremental deployment, it’s a hard sell
    - I gain more benefit and reduce my risk by delaying deployment

ROV is only useful if deployed widely (especially in large networks),

so we need to make it easier to deploy (especially in large networks)

RPKI Route Origin Validation, tl;dr

Signing

- Generate PKI certificates and signed objects called Route Origin Authorizations (ROAs) that link prefix/length(s) to origin ASN(s)
- Publish those certificates and objects in a Certificate Authority publication point

RPKI Route Origin Validation, tl;dr

Validating

- Stand up one or more Relying Parties
  - walk the Trust Anchors to find the CA pub points
  - ingest ROAs (rsync), validate the crypto
  - Push validation info to routers via RPKI-Router protocol
- Configure routing policy on ASBRs to do something with that info
  - Usually increase local pref on valids, drop invalids

Signing Prefixes - Hosted

Hosted – ARIN (or $RIR) as CA

- Generate key, upload to ARIN
- Use their portal to manage ROAs

Issues:

- Have to trust a third party with your private key
- 100% reliant on ARIN’s infrastructure
- PA delegations have to be proxied from downstream customers to ARIN
  – Additional portal/API development to glue things together

Signing Prefixes - Delegated

Delegated – Roll your own

- Install Certificate Authority software, generate keys
- Generate ROAs for all resources you want to sign
- Publish URI for your CA's publication point through ARIN's TA

Issues:

- Careful where you store your keys (not publicly-reachable server)
- TA can only publish one URI per publication point
- Still reliant on ARIN's TA infrastructure

Determining What to Sign

- Need accurate records
  - What prefixes are used where? Purpose? Prefix size(s)?
  - Where are prefixes aggregated/filtered?
  - Which ASN originates?
  - PA customer space
    - Proxy sign
      - At supernet level (static)
      - At subnet level (BGP)
    - Delegate to customer CA
  - Integration to COTS IPAM

Determining What to Sign

Or…

- Over-sign and pollute the database with potentially unnecessary records
  - Every ROA containing a range from supernet down to /24? (/48)
  - Every possible origin ASN
  - Still have to keep track of customer prefix/ASNs

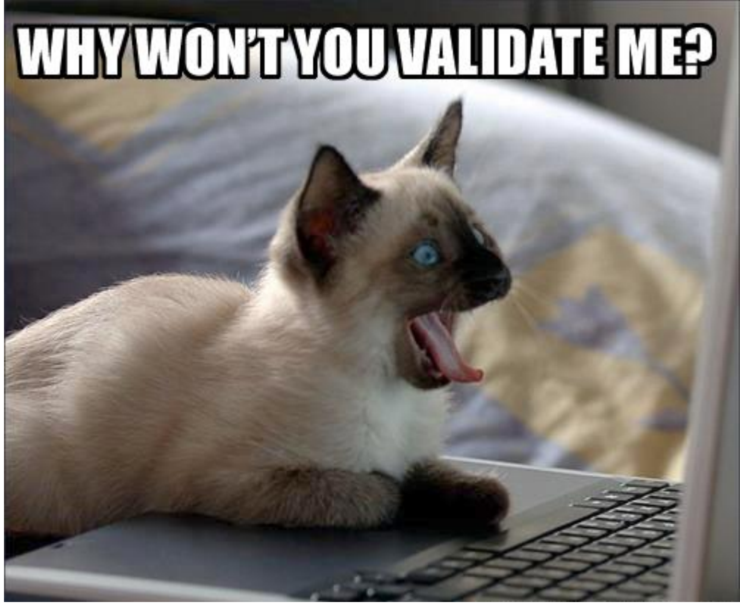
Doing it right means:

- full-scale address audit
- automation to keep records in sync with reality
- customer portal to manage delegation and proxy signing

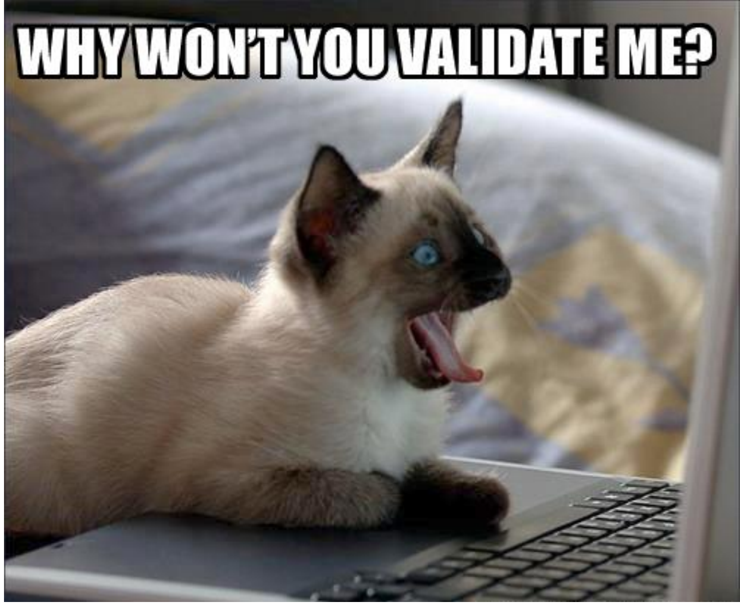
Doing this manually doesn't scale.

Validating Prefixes

- Deploy servers running Relying Party and RPKI ↔ Router software
- Upgrade at least ASBRs to RPKI-capable code
- Point RP software at the TAs
- Build routing policy (usually involves LocalPref)



Validating Prefixes



Issues:

- Adding policy to manipulate local pref without interfering with existing local pref policies can be complex
  - May require some logic to conditionally apply the correct values wherever the LP is set/manipulated
    - LP already exists: pre-existing LP + Validity = new LP
    - LP doesn't exist: Validity + desired LP for a given route type/origin = new LP
- What's an ASBR when you have multiple ASNs?
  - Validation status is a non-transitive community
- Must sign ARIN Relying Party Agreement to use ARIN's TA

Operational Issues - Ownership

- Who owns this set of boxes?

Are they:

- Security devices?
- Routing infrastructure?
- Mission Critical applications/servers?

A different group is often responsible for each

Challenge:

- If it’s Security, how much do the security guys have to know about routing?
- if it’s routing, how much do the router guys have to know about PKI and secure key management?
- If it’s applications, do you have to teach the systems guys about both?

Is the answer different for CA (sign), Publication Point, and RP (validate)?

Operational issues - Failure model

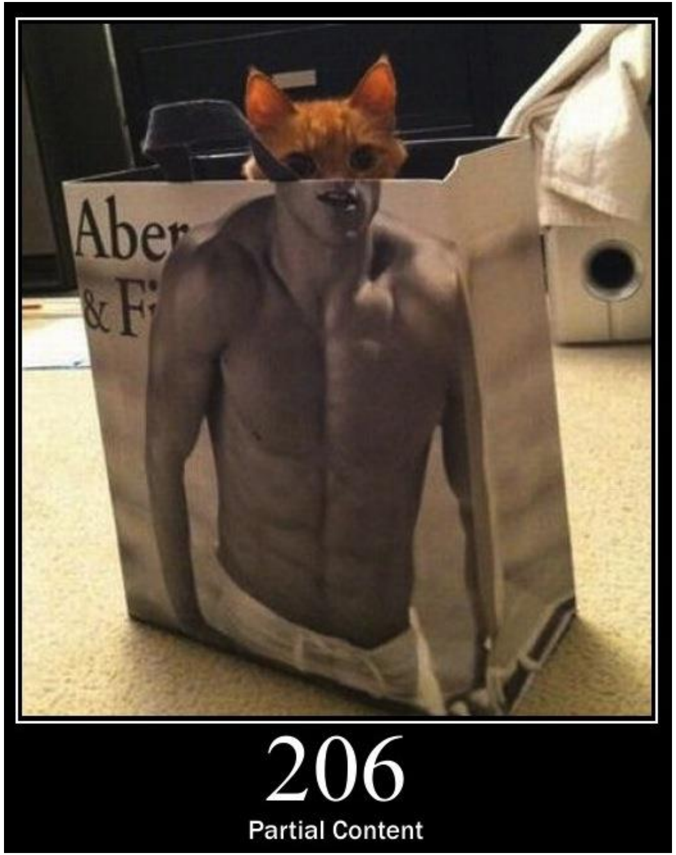
- Current assumption: occasional failures are ok because they mostly fail open
  - Validation failures, other errors fall back to unknown (i.e. unvalidated, unprotected routes)
    - Looks like incremental deployment (not everything is participating yet)
    - How do I tell the difference between broken, not deployed, and actually wrong?
    - How often is too often to fail open and lose the protection I deployed to gain?

What you want out of the system

- Availability
  - Uptime commensurate with the importance to global routing
  - "As long as it's not down when the certs expire"/human time scale isn't really a valid assumption
    - Multiple parts of the system can fail independently (TA, CA pub point)
    - Failures result in routes with no origin validation (exposed to attack)
  - Geo-diverse
    - not just off-site cold standby/DR backup
  - Need something better for resiliency than load-balancers or DNS priority hacks to get around single URI requirement

What you want out of the system

- Consistency
  - Don’t change things out from under rsync (atomicity)
  - Hard to do when you’re synchronizing large filesystem structures instead of single files
    - Scaling considerations -> http://bit.ly/1wejn7f
  - This is a loosely consistent system by design, goal is to reduce the opportunities to be bitten by that fact of life with distributed systems



What you want out of the system

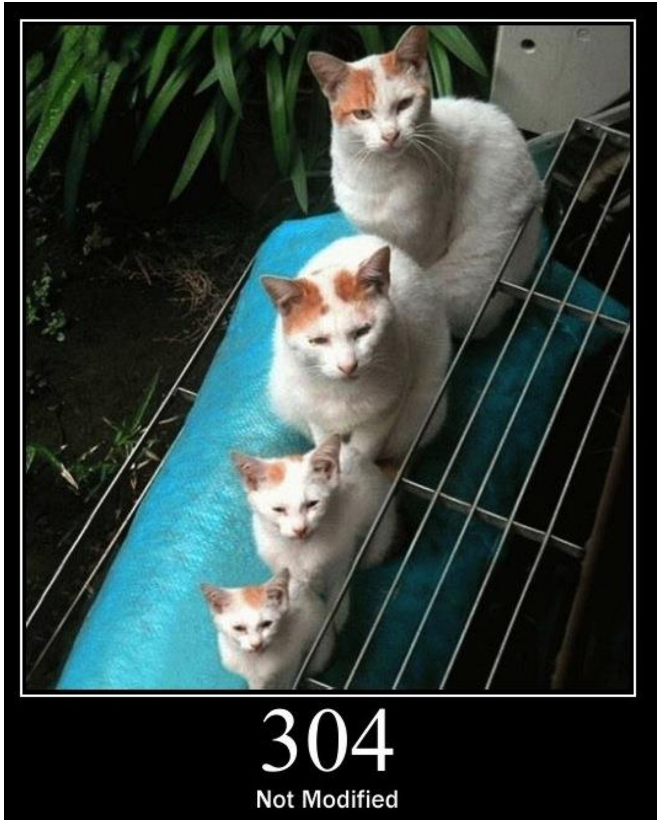
- Data Accuracy
  - Clerical error, system compromise, legal compulsion, fraud
  - Potentially worse since it might result in routes declared invalid and dropped
    - Note: Invalid ROA != Invalid route, invalid ROAs are ignored
  - Bundled/hierarchical nature of certificates mean that if parent cert claims don’t encompass child cert claims completely, child cert (and all of its children) is invalid (see draft-ietf-sidr-rpki-validation-reconsidered)
    - Makes the process for transfers between CAs fragile

How to Fix - Availability

- Support a list of URIs for TAs, CA pub points, try one until you have success
  - Like DNS: more than one place to go for a consistent answer (multiple root servers, multiple auth servers)
    - Still single copy, so no comparison/discrepancy handling needed
- Anycast TA and CA (with rsync?)
- Or ditch rsync? -> `http://bit.ly/1lNYIWR`

How to fix - Consistency

- CA pub point ↔ RP sync
  - Sync tar(s) instead of syncing files? (atomic sync)
  - Serial numbers/TTLs like DNS so that you know when you’re in sync (draft-tbruijnzeels-sidr-delta-protocol)
- Consistency among redundant pub points or TAs
  - Hidden master, push filesystem snapshots or repository tars to one or more public (read-only) servers when data changes
    - Looks a lot like uploading a new DNS zone file

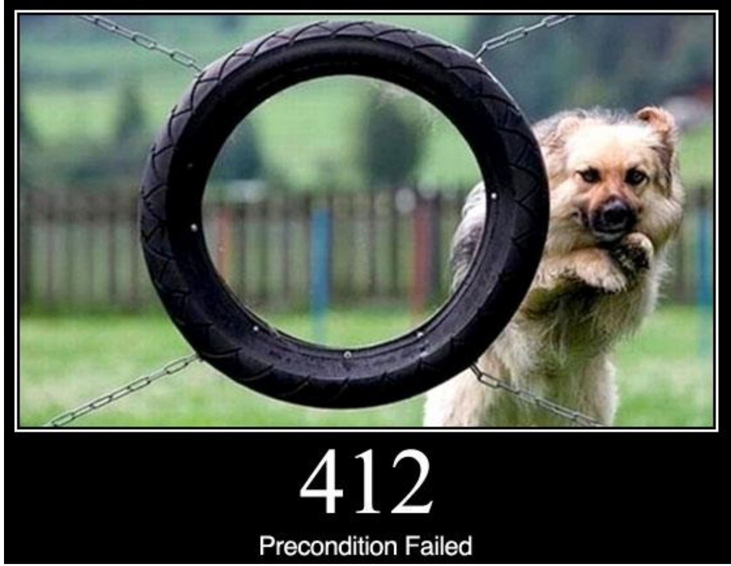


How to fix – Data Accuracy

- Dependent on TA and CA Policies (CPS)
  - Procedural consistency and rigor
  - Authentication and Verification for changes
- PKI bundled hierarchy is an ongoing discussion
- Legal compulsion is an unknown – single root vs. multiple, different jurisdictions
- Your idea here:

How to Fix? - ARIN

- Change ARIN's RPA to fix legal "dealbreakers"
  - Indemnify and hold harmless
    - Clarify that this isn't a requirement to defend ARIN
  - No liability or warranty
    - Change to a FOSS-style no warranty statement
    - Best-effort SLA
      - Availability
      - Process and infrastructure hardening to prevent fraud/clerical errors
      - Notification of externally forced (LE/Judicial/Legislative) changes before they are made
- Stop requiring non-ARIN members to sign RPA to access ARIN's TAL
  - Current situation means that ARIN region's routes may remain unvalidated outside of ARIN region
  - We seem to be unique among RIRs in enforcing such a requirement

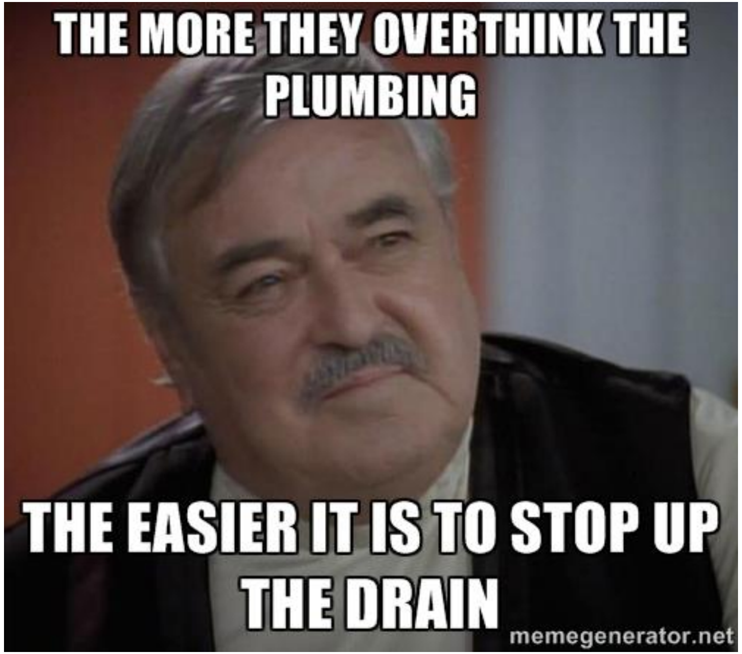


ARIN - Policy

- Are the RIRs the right host point for mission-critical applications like this?
  - Resource commitment from members
  - SLA commitment to customers/members
  - Experience with mission-critical hosting
  - Policy/governance
- RIRs often say that they do not set routing policy
  - ROV can fundamentally alter traffic flow/global routing, how do we guide implementation?
  - Fix via ARIN Policy Development Process (PDP)?
    - Board will likely see this as contractual/operational issue, out of scope for the PDP
    - Already removing ops-focused stuff from NRPM (ARIN-2014-5, ARIN-2014-6)

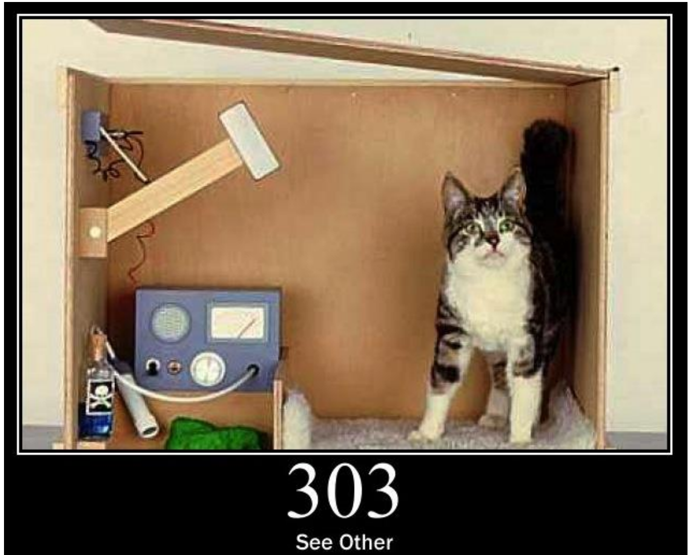
#include Pithy_words_for_summary

“RPKI ROV will succeed where others have failed because it replaces complex things like email templates, web forms, and router config with simple, easy-to-understand public key infrastructure”
– Rob Seastrom


THE MORE THEY OVERTHINK THE PLUMBING
THE EASIER IT IS TO STOP UP THE DRAIN
memegenerator.net

Alternatives

- ROV depends on a critical mass of deployment to provide the expected benefit
  - People signing routes to protect against origin hijacks need large networks to drop invalid routes
  - People validating routes need originators signing their routes so that they can detect invalid ones
- Sounds a lot like other recommendations that we need “everyone” to do:
  - Keep your data accurate in RADB, IRR
  - Filter your customers’ BGP announcements inbound
  - draft-ietf-opsec-bgp-security
  - MANRS (routingmanifesto.org)



Questions? Flames?

- Thanks to:
  - HTTP Status Cats, HTTP Status dogs, meme sites everywhere
  - Rob Austein, Rob Seastrom, Michael Abejuela, Geoff Huston, Sandy Murphy, Chris Morrow