SCTE
a subsidiary of CableLabs®

Creating Infinite Possibilities.

# Improve Routing Security by validating BGP (Border Gateway Protocol) with RPKI (Resource Public Key Infrastructure)

Tony Tauber

Distinguished Engineer
Comcast
tony_tauber@comcast.com

SCTE CABLE-TEC EXPO®22
SEPTEMBER 19-22 • PHILADELPHIA, PA

2022 Fall Technical Forum
SCTE® • CABLELABS® • NCTA

SCTE
a subsidiary of CableLabs®

Creating Infinite
Possibilities.

# BGP and RPKI Background

# Network Operators advertise IP address space reachability

Internet is composed of tens of thousands of Autonomous Systems (ASes)

- Access networks (Wireline, Mobile, etc.)
- Content Delivery Networks (CDNs)
  - Cloud providers and Web hosting
  - Search engines
  - Entertainment
  - Social networks
- Enterprises
  - Companies
  - Colleges and Universities
  - Banks

# BGP Security Risks

## Misconfigurations and Malicious Actors

- Traffic doesn’t go to the right place and maybe goes to the wrong place

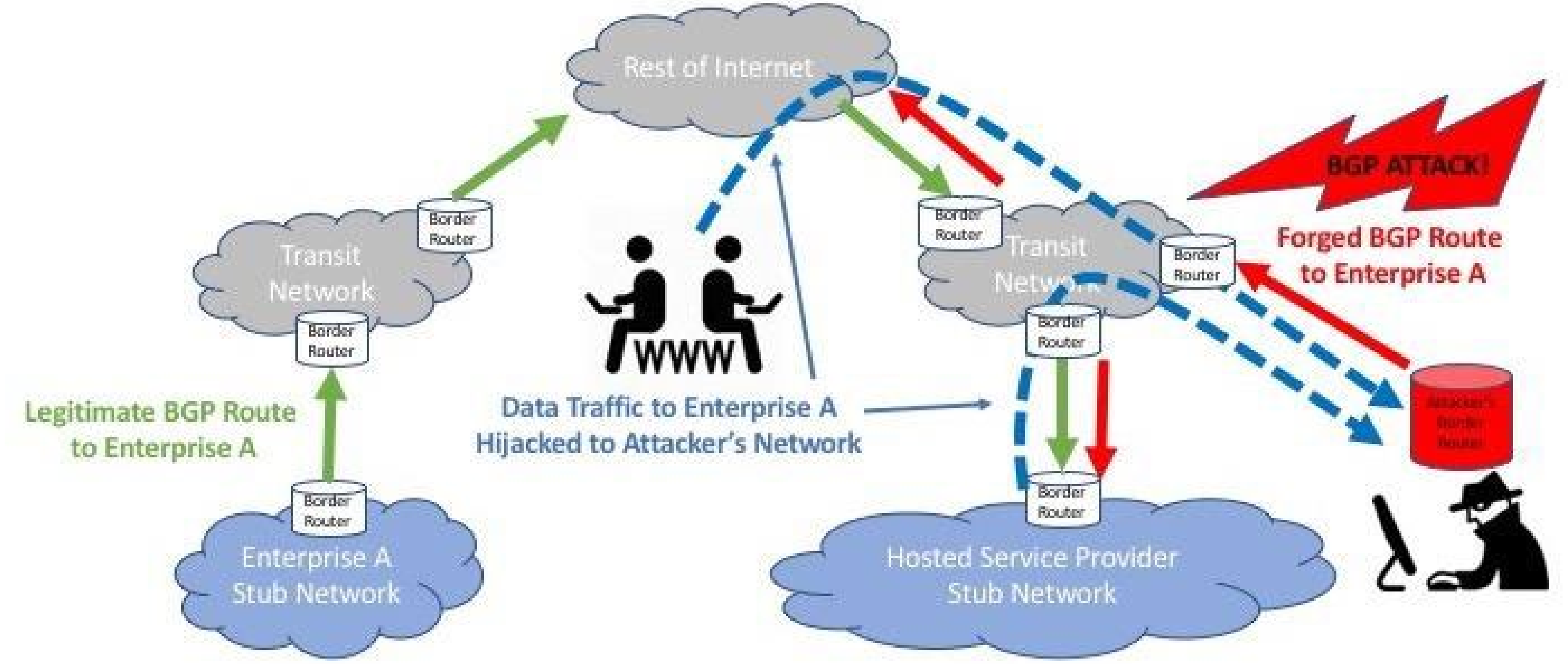
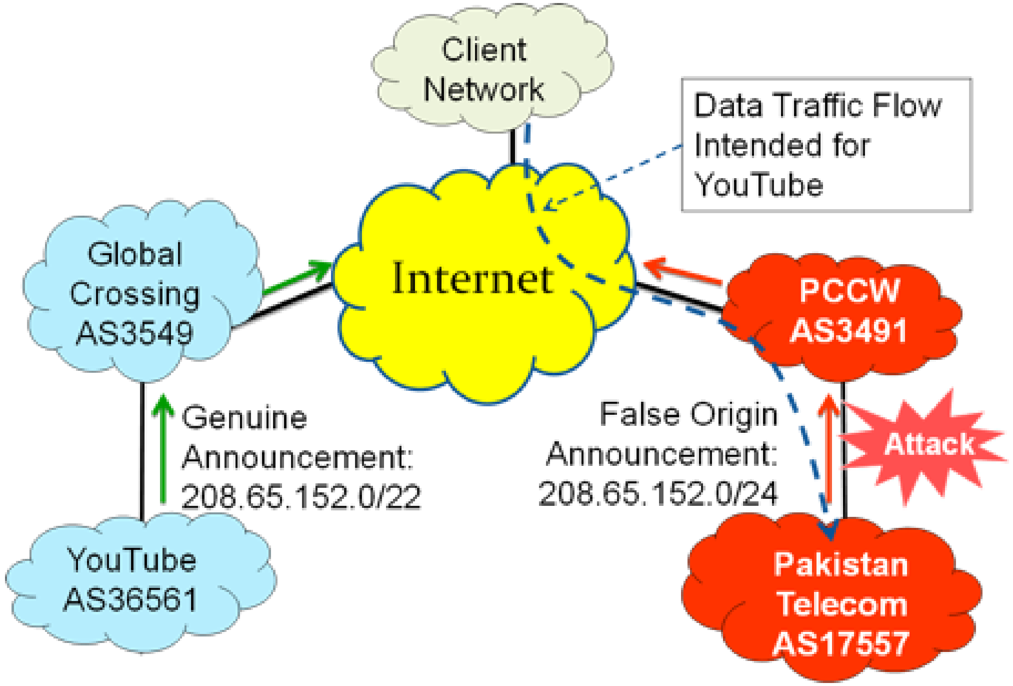

Rest of Internet
Transit Network
Border Router
Border Router
Transit Network
Border Router
Border Router
Border Router
BGP ATTACK!
Forged BGP Route to Enterprise A
WWW
Legitimate BGP Route to Enterprise A
Data Traffic to Enterprise A Hijacked to Attacker’s Network
Attacker's Border Router
Border Router
Enterprise A Stub Network
Border Router
Hosted Service Provider Stub Network

Diagram courtesy of NIST (US National Institute of Standards and Technologies

# Canonical BGP “hijack” incident

## Pakistan Telecom vs. Large Streaming Video Provider

- Attempted in-country censorship
- Accidentally leaked externally
- Global outage for users



Client Network
Data Traffic Flow Intended for YouTube
Internet
Global Crossing AS3549
PCCW AS3491
Genuine Announcement: 208.65.152.0/22
False Origin Announcement: 208.65.152.0/24
Attack
YouTube AS36561
Pakistan Telecom AS17557

Diagram courtesy of NIST (US National Institute of Standards and Technologies

# RPKI Route Origin Validation (ROV)

## Acronyms

RPKI = Resource Public Key Infrastructure

- The system

ROA = Route Origin Authorization

- The main item of interest

ROV = Route Origin Validation

- How it gets used – the process it enables

AS = Autonomous System

- A network speaking BGP

# RPKI Route Origin Validation (ROV)

## Function

- A method for the ”owner” (registered user) of a prefix to assert which AS(es) are the correct originator(s) for that prefix
- Asserts (implicitly) that other originators are not valid

# What is in a ROA?

A digitally signed statement consisting of

- prefix
- maximum prefix length
- originating ASN

RPKI also has other types of objects to make it work:

- Certificates
- Manifests
- CRLs (Certificate Revocation Lists)

# BGP ROV States

## Routers compare BGP routes received to VRPs

Three possible states

- **NotFound (a.k.a. Unknown)**
  - BGP route doesn’t match any ROA
- **Valid**
  - BGP route matches a ROA – same Origin AS and same length or w/in ”maxlen”
- **Invalid**
  - The ROA and route announcement differ either of these ways:
    - Originating ASN
    - Maximum length (”maxlen”)

BGP Routes which are “invalid” are dropped; others are kept

# RPKI Authority and Delegation Structure

## RPKI structure follows IP addressing Structure

The “root” assigner of all IP space (IPv4+IPv6) is IANA

Delegated to 5 “continental” RIRs (Regional Internet Registries)

- ARIN, RIPE NCC, APNIC, LACNIC, AFRINIC

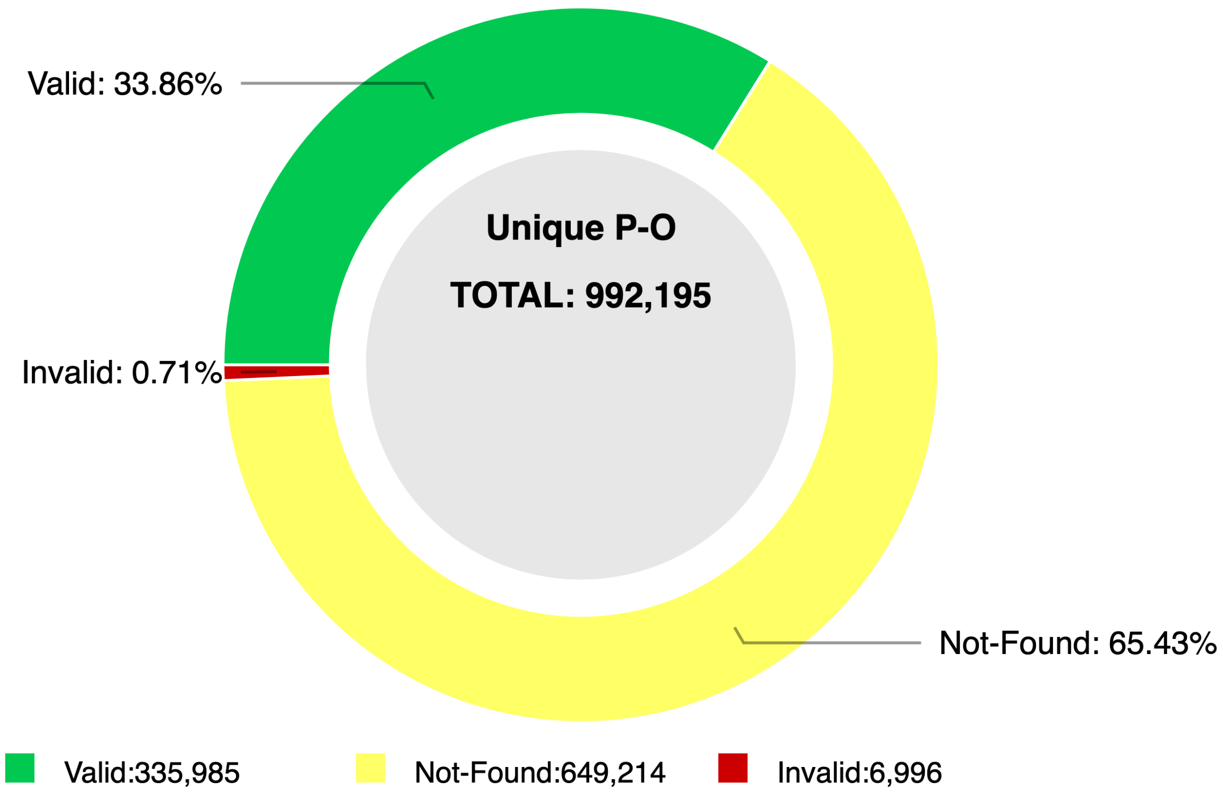
They assign further to

- LIRs (Local Internet Registries)
- Service Providers
- Enterprises

RPKI is a X.509 Digital Certificate architecture aligned similarly

# ROA Coverage – IPv4 Address Space

RPKI-ROV Analysis of Unique Prefix-Origin Pairs (IPv4)



**NIST RPKI Monitor:** RPKI-ROV Analysis **Protocol:** IPv4 **RIR:** All **Date:** 2022-01-22 06:00

Courtesy: NIST (National Institute of Standards and Technologies
https://rpki-monitor.antd.nist.gov/ROV

Creating Infinite Possibilities.

SCTE
a subsidiary of CableLabs®

# Validating

# How does it work?

Repositories

- ROAs are published on servers operated by RIRs and their delegates

Validating Caches (VC) running Relying Party (RP) software

- Servers running validator software which fetches ROAs and other data using
  - rsync – TCP protocol for synchronizing files on servers (TCP port 873)
  - RRDP – RPKI Repository Delta Protocol which uses HTTPS as transport (newer, preferred)
- Run cryptographic integrity checks to produce VRPs (Validated ROA Payload)
- RPKI-to-Router (RPKI-RTR) protocol (TCP port 323 or 8323)
  - Allows for fetching VRP data by routers
  - Routers cache the data locally and refresh at intervals
    - Retain local cache for a configurable time in case connection to cache is lost

# ROV Deployment Considerations

## Risks

Fail-open model

- Given that most of the prefixes are still not covered (i.e., “not-found”)
- Hence absence of a covering ROA will still allow for route propagation
- Same for failure or loss of VC infrastructure
- Same for complete loss of RTR connections/data at router level
  - (more later)

Already some large ISPs doing ROV

- Hence invalid announcements are already getting dropped

## Risk Analysis

Low barrier of entry

- No new gear (features on existing routers)
- Some VMs running freely available open-source software

Risk of doing nothing

- Vulnerability of mis-origination by others

Risks of doing something

- Collateral damage, increased complexity, new troubleshooting

Management buy-in

- Can’t make the case as an individual just in case something goes wrong
  - “Whose idea was this anyway?”
- Different recent incidents in the trade press helped the case

# Reading (Validation) / Writing (Publishing)

## Can do one without the other, not necessary to do together

Need to work as incrementally as possible

- Can’t enable things globally
- Always have a backout plan of each (sub-)step

Publishing

- Hosted model: RIR publishes the data that members enter in the portal
    - e.g., ARIN Online
- Delegated model: RIR delegates to LIR (Local Internet Registry)
    - Run own CA (Certificate Authority) and PP (Publication Point) servers

Validating... (covered in later slides)

# Reading (Validation) / Writing (Publishing)

## High-Level Deployment and Implementation Plan

Reading – Route Origin Validation using published ROAs

- Add inbound route-policy to ”drop invalid” after dropping bogons
- Field trial with subset of interconnection partners in August 2020
- Broader rollout through remainder of 2020 and early 2021

Writing – Publishing ROAs for our own address space

- Start with one or small number of prefixes
- Gradually expand

# Reading (Validation) / Writing (Publishing)

## Environment

Validation – Cisco/Juniper edge routers

- Incremental rollout

Publication – ROA generation

- 100 + prefixes
- Two dozen internal ASNs
- Thousands of more-specifics

## Rollout

Easier to do with small risk

- Luckily, it "fails open" – in absence of a ROA, BGP route is accepted

Only external eBGP sessions

- Not on sessions among our different regional ASes for instance
- No iBGP (doesn't even make sense)
- Key reason: we carry many more-specifics internally

Config per router, per neighbor

- Easier to see if something goes wrong and back out if necessary

Pairwise coordination with all partners is not the goal, notification is

# Validating Cache (VC) Relying Party (RP) Software/Servers

## Infrastructure Design Goals

Geographic diversity

- Deploy to two different data centers in case one has an outage

Software diversity

- Deploy two different codebases in case one has a problem

Several freely available open-source options

## Infrastructure Design Goals

Initial choices

- Routinator – NLnet Labs – written in Rust programming language
- RIPE RPKI Validator v.3 – RIPE NCC –written in Java language
  - Later replaced with rpki-client (with web wrapper) and StayRTR

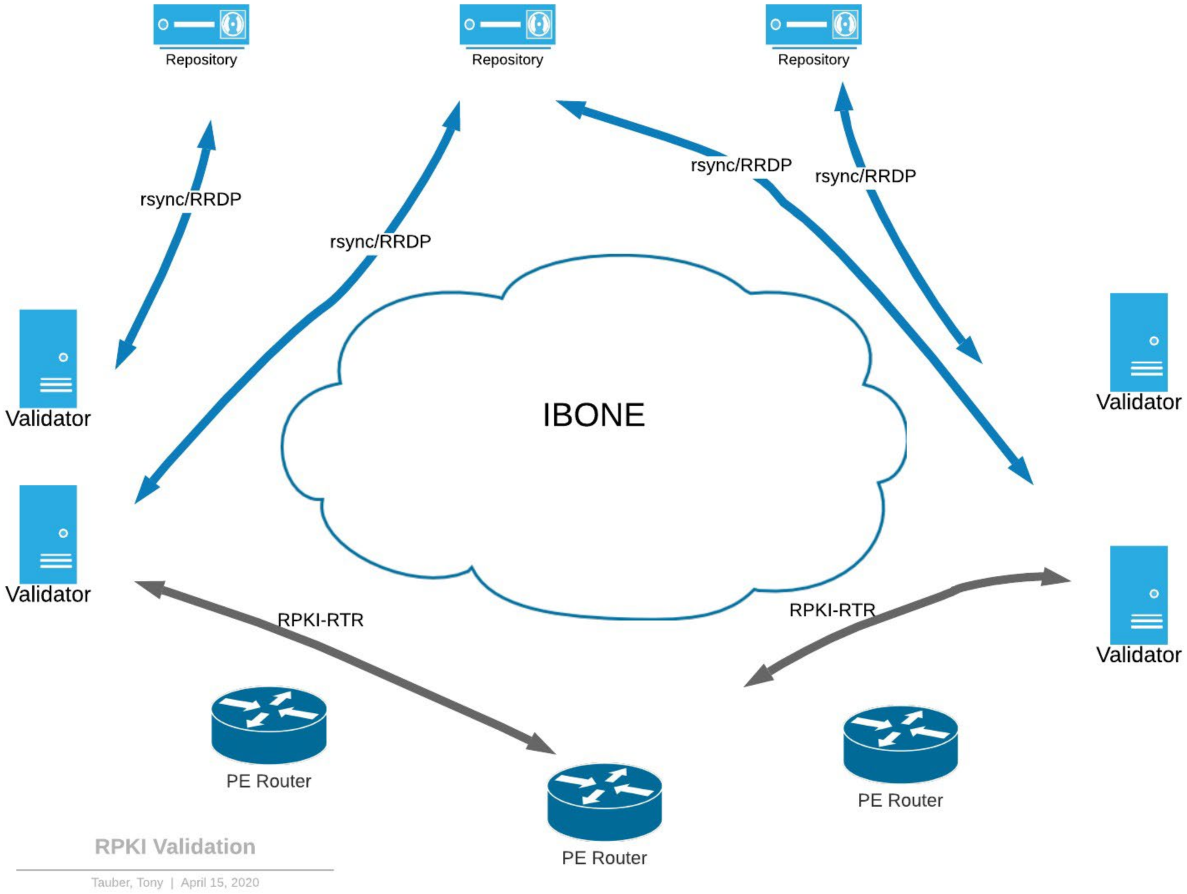
Hence, each router will have 4 different RTR servers configured

- Deployed and managed by our DNS staff

All have packages now, easy to install and keep updated

Can produce metrics also for consumption

# ROV Components and Data flows



# Findings

## Bugs?

Router vendor software had some bugs

- Made sure to patch to the recommended versions

RP Software has had some bugs

- Mostly bounds-checking and the like
- Installed fixed packages as they were released

SCTE
a subsidiary of CableLabs®

# Creating Infinite Possibilities.

## Publishing

# Signing and Publishing ROAs

## Hosted Model

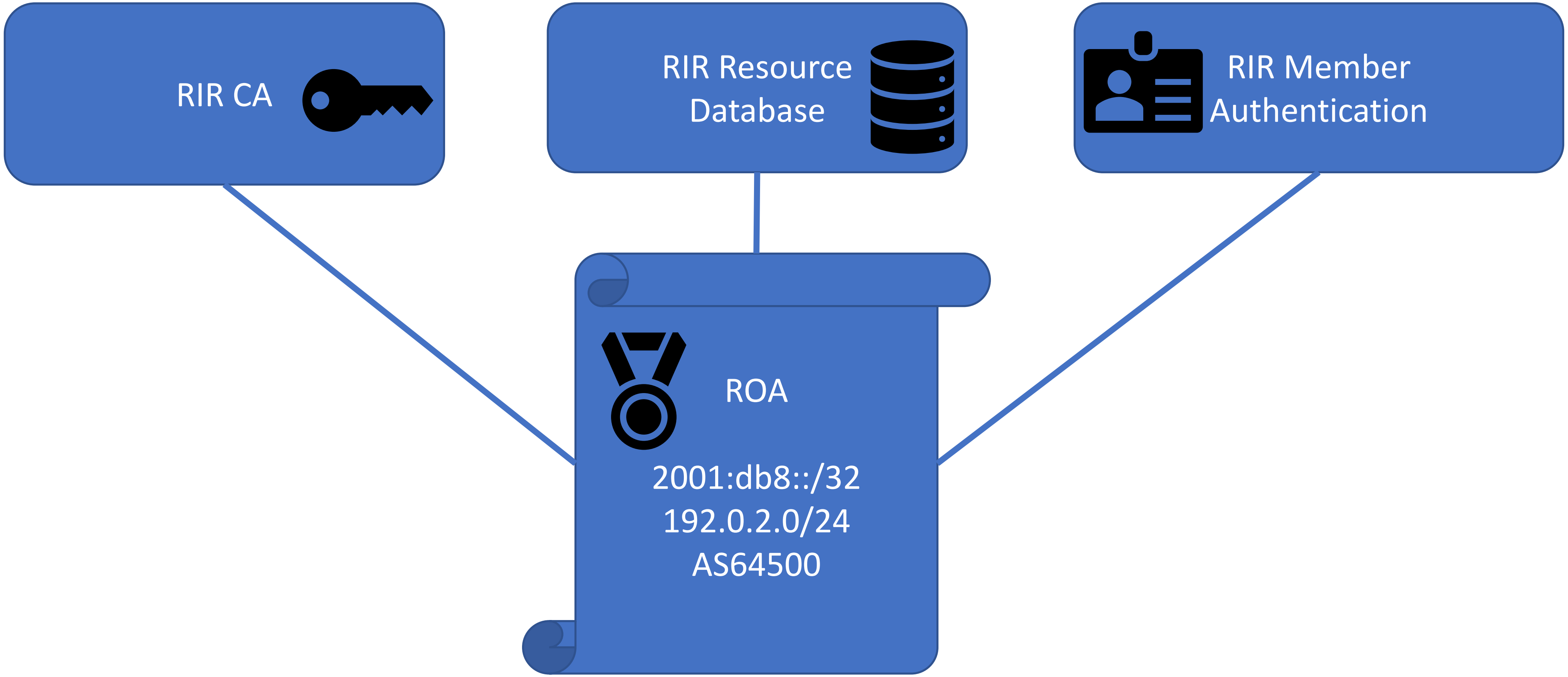
Via RIR portals

- Varying degrees of ease and integration
- For example, publish ROAs to match existing BGP announcements

APIs available

- ARIN API script – Rich Compton from Charter Communications
- Not polished but wouldn't be possible without it!

# ROA Creation Components – RIR Hosted Model



RIR CA

RIR Resource Database

RIR Member Authentication

ROA

2001:db8::/32
192.0.2.0/24
AS64500

# Signing and Publishing ROAs

## Delegated Model

Address-issuing authority delegates to you

- RIRs in our case, could be more layers down
- Issues a Certificate which is used to sign ROAs and other artifacts
- Hosts a record with URL to Publication Point (PP)

Certificate Authority (CA) and Publisher Software:

- Krill – NLnet Labs
- rpki.net – Dragon Labs

Publication point (PP) needs to be globally reachable

Info about running own RPKI CA

- https://www.slideshare.net/apnic/should-i-run-my-own-rpki-certificate-authority

# Decision Points

Delegated

- Extra servers and software to run
- Availability profile a bit unknown

Hosted

- Less of these risks….

Went with Hosted at this point

- Share fate with thousands of others
- Consider revisiting at a later date
- Hybrid model (CA internal, PP hosted elsewhere) has some appeal

# Considerations

Larger risk

- Can create connectivity issues if something goes unreachable
- Can take time to back out or correct
  - ROA distribution is on order of minutes to hours
- Make sure to do it carefully

Our complexity

- Something over 100 address blocks
  - Almost all ARIN, a few from other RIRs
- Distributed unevenly across more than 20 different ASes
  - Backbone, Regional, Data Center, Enterprise

# Signing and Publishing ROAs

## Process

Issuing ROA for largest blocks makes ROAs underneath “invalid”

- Unless there’s a matching ROA for the more-specific already

Gradually roll out

- Sign few non-intrusive prefixes
- Start from “bottom” (more-specific prefixes)
- Once all filled in, issue ROAs for top-level blocks

Integrate with IP management software in a later phase

Ended up publishing several thousand ROAs (mostly IPv6)

- Fewer blocks but so much more to break apart

SCTE
a subsidiary of CableLabs®

# Creating Infinite Possibilities.

# Thank You!

Tony Tauber

Distinguished Engineer
Comcast
tony_tauber@comcast.com

SCTE CABLE-TEC EXPO®22
SEPTEMBER 19-22 • PHILADELPHIA, PA

2022 Fall Technical Forum
SCTE® • CABLELABS® • NCTA

COMCAST