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

A Technical Paper prepared for SCTE by

Tony Tauber
Distinguished Engineer
Comcast
Cambridge, MA
tony_tauber@comcast.com

Courtney Smith
Principal Engineer
Comcast
Mt. Laurel, NJ
courtney_smith@comcast.com

Table of Contents

Title **Page Number**

List of Figures

Title **Page Number**

1. Introduction

In this paper, we will discuss the basic operation of BGP and inter-provider Internet routing including some vulnerabilities of the system. We will then describe RPKI, a set of technologies developed by the IETF (Internet Engineering Task Force) to help address a sub-set of these vulnerabilities. Deployment of these tools is not without risk and complication, and we will describe how we went about assessing and enabling RPKI in a large MSO network including design tradeoffs and lessons learned.

This is a complex set of technologies which have matured over the course of a decade or more of implementation and deployment experience. It operates across many operational and software domains with many actors and interacting systems. Hence decisions whether to deploy it and exactly how to do so while minimizing risk and disruption must be balanced carefully. These risks and complexities are magnified as function of the number of resources, interactions, and systems involved.

Making use of RPKI to improve security for an MSO (Multi-System Operator) service provider network is just such a complicated scenario and special care must be taken. In this paper, we describe the various considerations we assessed and decisions we made in the course of this enablement.

2. BGP Background

In order to communicate with given resource on the Internet, it's necessary to send packets to the destination IP (Internet Protocol) address which can service that request. BGP (Border Gateway Protocol) is the method different networks that make up the Internet use to communicate what endpoints (IP addresses) can be reached on their infrastructure. As put in RFC 4271, "[t]he classic definition of an Autonomous System is a set of routers under a single technical administration". BGP speakers (routers configured with BGP) within an AS can communicate reachability information for IP prefixes which are reachable within the infrastructure of that AS to neighboring ASes.

As a BGP message (sometimes also called an "advertisement" or "announcement"), leaves an AS border router to an neighboring AS, if the first AS originated the announcement for a given IP address block, that AS puts its ASN (Autonomous System Number) in the BGP message as the "origin AS". Each AS that conveys that reachability information onward via BGP appends its own ASN in a chain known as the "AS-Path". The AS-Path has a few functions including preventing loops in the topology and also operationally tracing responsibility for handing routing and traffic for a given destination.

To take an example, if AS150 originated a BGP route for 2001:db8::/32 which was then received by AS280 and passed along from there to AS320, we would expect an AS-Path of "320 280 150" read from left to right as "nearest" to "furthest", ultimately to "origin" AS.

[Include references and pictures.]

3. BGP "Hijack" Risks

The basic BGP protocol was specified almost three decades ago (RFC 1654 superseded by later RFCs) and for more than two decades the operational community has recognized some security risks with the protocol.

One important aspect of the Internet routing system is that routing information flows not only between directly adjacent ASes but also, transitively, to distant ASes. The complexity and dynamic nature of the global Internet routing system historically has not had any "ground truth" declaration of the proper topology and relationships among the various ASes and IP address blocks connected to them.

BGP mis-originations, sometimes called "hijacks", refer to cases where a network other than the legitimate address-holder incorrectly advertises reachability for a given IP address block. The term "hijack" suggests malicious intent however in most cases the intent of the actors involved is not disclosed or definitively known. Some cases appear to be the result of a configuration error where some characters in the address block may have been transposed or another typographical error has occurred. In some other cases, a bug in a router or other piece of network equipment seems to have caused the problem. The risk of such events can be visualized by this diagram:

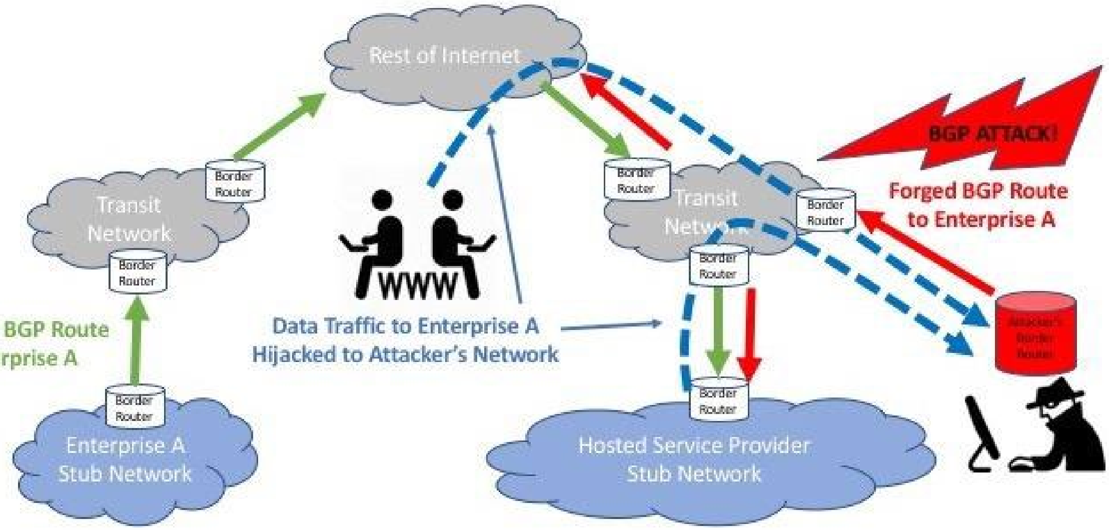
Rest of Internet
Transit Network
Border Router
Border Router
Border Router
Transit Network
Border Router
Border Router
BGP ATTACK!
Forged BGP Route to Enterprise A
Attacker's Border Router
WWW
Legitimate BGP Route to Enterprise A
Data Traffic to Enterprise A Hijacked to Attacker's Network
Border Router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

Figure 1: BGP Hijacks steal and divert Internet traffic to attackers.

(Courtesy USA National Institute of Standards and Technology)

Perhaps the most well-known case is the 2008 Pakistan Telecom incident which appeared to involve a BGP redirection within the Pakistan Telecom network to possibly enable censorship of a popular Internet video provider in country. The bogus BGP information leaked out to the broader Internet and ended up disrupting legitimate traffic intended for said provider as illustrated in the diagram below

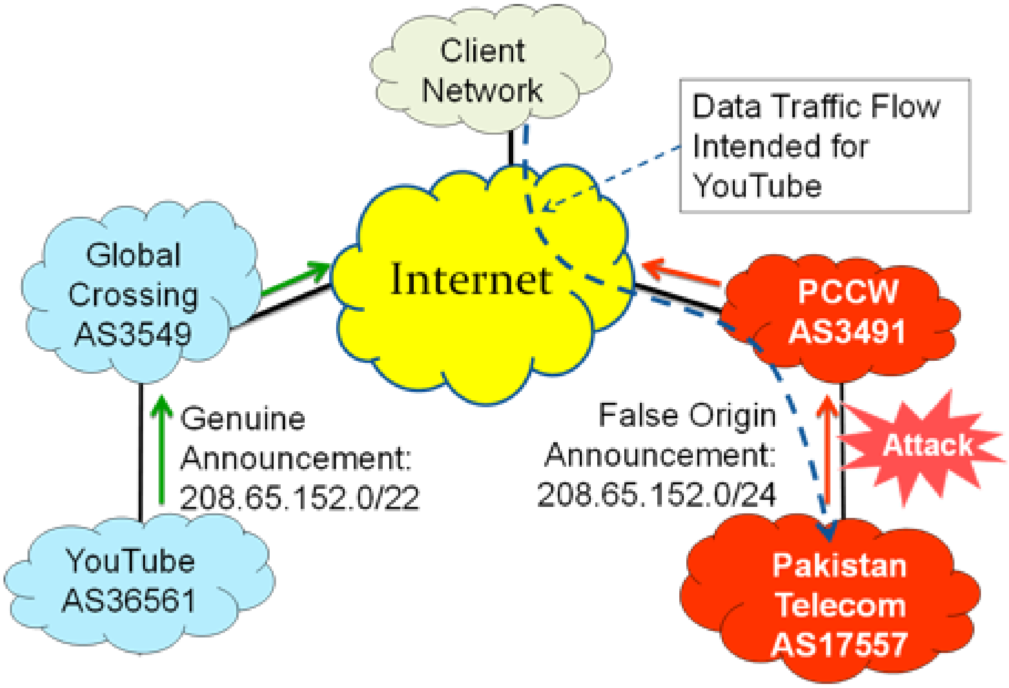
Client Network
Data Traffic Flow Intended for YouTube
Global Crossing AS3549
Internet
PCCW AS3491
Genuine Announcement: 208.65.152.0/22
False Origin Announcement: 208.65.152.0/24
Attack
YouTube AS36561
Pakistan Telecom AS17557

Figure 2: 2008 Pakistan Telecom BGP incident

(Courtesy USA National Institute of Standards and Technology)

A catalogue of public BGP hijack incidents on Wikipedia includes approximately twenty episodes spanning many years and that list is not exhaustive.
To address this particular weakness in the routing system, the IETF SIDR (Secure Inter-Domain Routing) Working Group worked to develop new enabling technologies. We will discuss the first one of those below and what Comcast did to utilize them.

4. RPKI and ROV Overview

RPKI (Resource Public Key Infrastructure) defined in RFCs 6480-6493, is a method which follows the IP address assignment hierarchy. The root of Internet addressing rests with IANA (the Internet Assigned Number Authority) which delegates addresses to the 5 Regional Internet Registries (RIRs) assign IP address resources to different network operators or delegate further to LIRs (Local Internet Registries which can be on a country (e.g., China, Brazil) or Service Provider level.

The structure of the RPKI uses X.509 digital certificates to enable cryptographic verification of the chain of authority and particularly to issue ROA (Route Origin Authorization) objects which describe a mapping of an IP address range and the origin AS (Autonomous System) which has authority to announce reachability via BGP.

The figures below illustrate the delegation hierarchy manifested in the RPKI as well as the components needed to create a ROA (Route Origin Authorization) object.

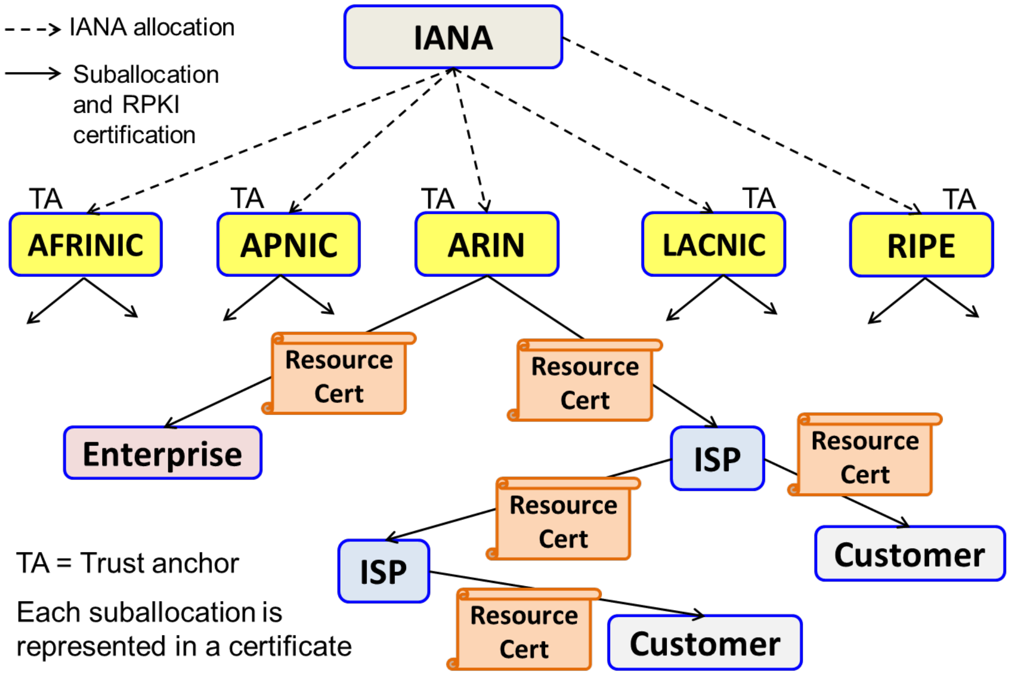

Figure 3: RPKI (Reseource Public Key Infrastucture) Delegation Structure

(Courtesy USA National Institute of Standards and Technology)

A ROA consists primarily of three elements:

- IP prefix (IPv4 or IPv6) or range
- Maximum valid length for a prefix within that range
- The AS authorized to announce ("originate") that prefix in BGP

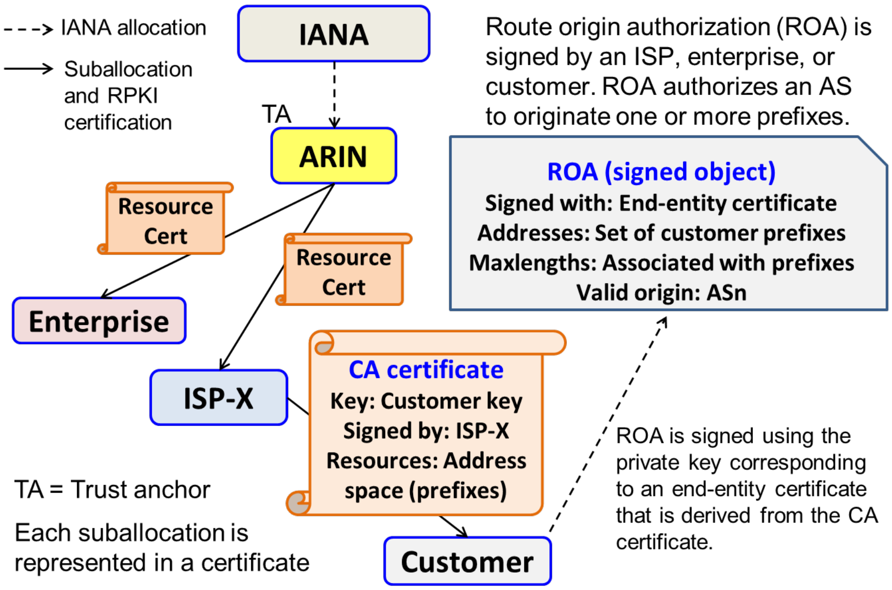
IANA allocation
Suballocation and RPKI certification
IANA
TA
ARIN
Resource Cert
Resource Cert
Enterprise
ISP-X
CA certificate
Key: Customer key
Signed by: ISP-X
Resources: Address space (prefixes)
Customer
Route origin authorization (ROA) is signed by an ISP, enterprise, or customer. ROA authorizes an AS to originate one or more prefixes.
ROA (signed object)
Signed with: End-entity certificate
Addresses: Set of customer prefixes
Maxlengths: Associated with prefixes
Valid origin: ASn
ROA is signed using the private key corresponding to an end-entity certificate that is derived from the CA certificate.
TA = Trust anchor
Each suballocation is represented in a certificate

Figure 4: ROA (Route Origin Authorization) Creation

(Courtesy USA National Institute of Standards and Technology)

These objects and associated artifacts, e.g., manifests and CRLs (Certificate Revocation Lists) are published so that they may be verified using any of several compliant RP (Relying Party) software packages. Once the verification is complete, the VC (Validating Cache) system renders the output mappings in a form that can be consumed by routers using the RTR (RPKI To Router) protocol.

A compliant router can then be configured to use the information to validate incoming BGP announcements and take action based on comparing them to this set of authorizations. This process is known as ROV (Route Origin Validation).

5. Deployment - Reading and Writing (Vaildating and Publishing)

There are two aspects to RPKI ROV; validating and publishing RPKI data which can be thought of as “reading” and “writing”. It is not required to deploy these at the same time, in a given order, or for a given operator to do both, necessarily. We will discuss each below.

5.1. Validating

In order to perform validation of BGP information, the source data must be collected and loaded into the routers. In this section, we will outline the components and process.

5.1.1. Relying Party software

A number of freely available open source software implementations of the RP (Relying Party) function have been developed over the spread of several years. Some of them have been maintained more vigorously than others.

RP software is run on VC (Validating Cache) servers and must have access to reach any Internet destination via at least the *rsync* protocol (TCP port 873) and RRDP (RPKI Repository Delta Protocol) in order to collect the RPKI data from the various publication points which cannot be known a priori and are communicated via URIs, including DNS (Domain Name System) names, embedded within RPKI data; hence, DNS-based and not IP-address based and not amenable to IP-based firewall rules or access controls.

After fetching all the RPKI data and performing cryptographic and other consistency checks against it, the VC produces a list of VRPs (Validated ROA Payloads) consisting of an IP prefix (IPv4 or IPv6), a maximum length (often referred as “maxlen”), and an origin AS authorized to originate such a prefix up to the “maxlen” value.

Additionally, RFC 8416 defines SLURM (Simplified Local Internet Number Resource Management) by which the operator of a VC can inject “local overrides” which will augment the public RPKI data as may suit the network operator’s needs.

5.1.2. Router configuration

The routers consume VRP data from the VC servers via the RTR (RPKI To Router) protocol, defined in RFC 8210, which runs via a TCP connection by default port 323 but can be some other user-defined port. Multiple RTR servers can be defined, and, for most router vendors, the router considers the union of all VRPs received via RTR. Refer to the diagram below for illustration of the data exchanges involved.

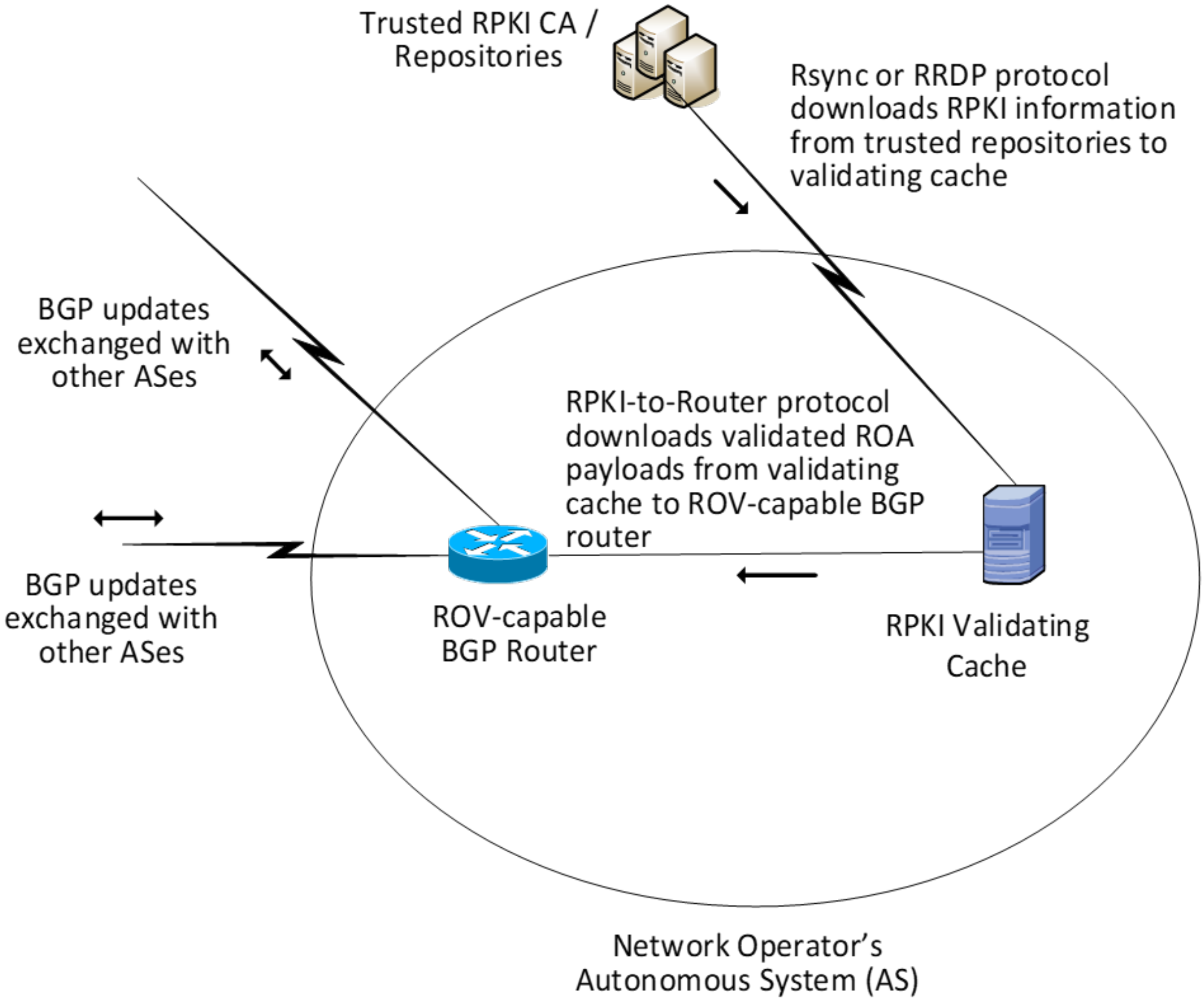
Trusted RPKI CA / Repositories
Rsync or RRDP protocol downloads RPKI information from trusted repositories to validating cache
BGP updates exchanged with other ASes
RPKI-to-Router protocol downloads validated ROA payloads from validating cache to ROV-capable BGP router
BGP updates exchanged with other ASes
ROV-capable BGP Router
RPKI Validating Cache
Network Operator's Autonomous System (AS)

Figure 5: RPKI ROV (Route Origin Validation) Data Flows

(Courtesy USA National Institute of Standards and Technology)

Then the router is configured to perform ROV on the BGP table. The configuration options and implementation details among the vendors may vary somewhat but for our purposes, we decided that the most important risk of erroneous BGP announcements was from other networks; hence we would focus on eBGP sessions to other networks (in our case we have eBGP sessions between different ASes that are under our administration so not truly “external”). With the configuration we selected, each BGP announcement received over these eBGP sessions would be evaluated against a local table of VRPs (Validated ROA Payloads) received from the VCs.
The validation of a BGP route against a ROA has one of three possible outcomes:

- **NotFound (a.k.a. Unknown)**
  - BGP route does not 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”)

It is very important to note that the operational model for RPKI ROV is that if a route is in the “not-found” category (no matching ROA in the routers local cache), the route should be accepted in the routing table and traffic forwarded to that destination. Routes which are “valid” should, of course, be accepted and “invalid” routes should be dropped. In this sense ROV is said to follow a “fail open” model which means that if the connection to the VC server is lost or the VC server loses its feed of RPKI information, traffic will continue to flow and not cause an outage.

At the time of this writing, (June 2022), below is a view of the global IPv4 BGP table and what routes are covered by ROAs. As this graph shows most of the Internet IPv4 routing table is not covered by any ROA but an ever-growing slice is covered, and a very small amount seems covered but not valid per the logic above. At the time of our deployment the portion covered by a valid ROA was closer to 30%.   The net effect of these factors meant that, if there were to be some failure in the RPKI components as we were deploying them, we would end up in no worse a situation than we had been prior to the rollout.

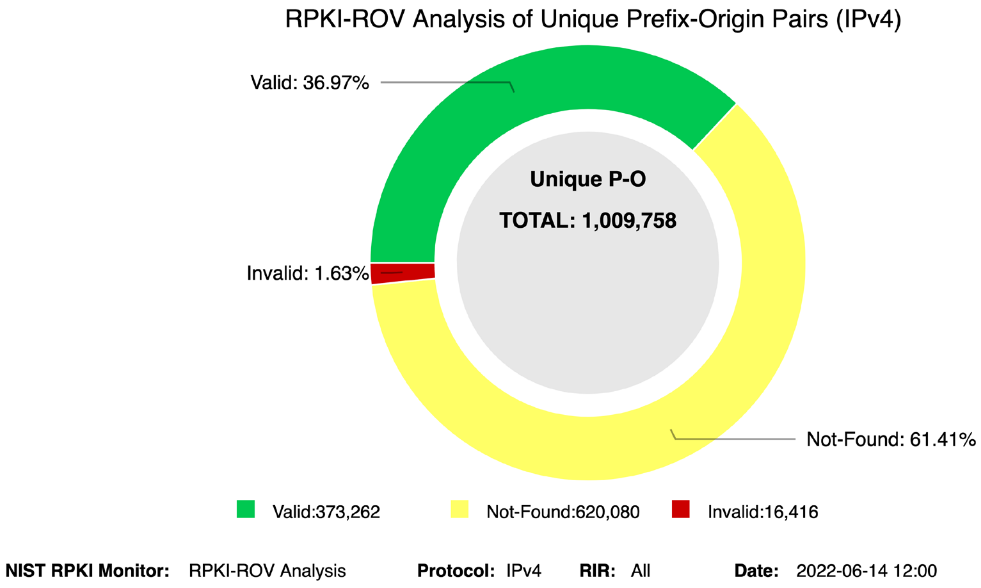

Figure 6: RPKI ROV Analysis of IPv4 prefix-origin pairs

(Courtesy USA National Institute of Standards and Technology)

5.1.3. Deployment considerations

Given the fairly large number of variables that we had to confront, some basic guiding principles made it possible to narrow the world of possibility to a smaller set of options. We sought to reduce the potential risks involved with a new technology and these concepts helped:

- Diversity
- Redundancy
- Incremental rollout (and rollback)

5.1.3.1. Diverse RP software packages

One key decision was to utilize two different RP software packages. RPKI is still a maturing technology with many different deployment insights and considerations coming to light over time. By using two different implementations, if one had a problem either transient or perhaps somewhat longer lasting, that package could be taken out of production temporarily.

Initially, we chose:

- Routinator (written in the Rust programming language by NLnet Labs)
- RIPE Validator (written in Java by RIPE NCC)

When RIPE Validator was declared end-of-life in 2021, we replaced that with:

- rpki-client (written in C as part of the OpenBSD project)

5.1.3.2. Diverse data centers

Since a given data center could conceivably to offline or become isolated from some part of the network, we further refined our redundancy plan utilize one two VC servers in each of two diverse data centers (e.g., "East" and "West"). Each data center then would feature one of each of the above RP packages running.
Each edge router then would be configured with RTR sessions to each of the four VC servers. The final piece is to enable validation within BGP on a per-neighbor basis.

5.1.3.3. Incremental rollout which can be incrementally rolled back

Once a router is configured with RTR sessions with VCs, it is then ready to have validation enabled on eBGP sessions with other networks. We chose to enable ROV first on BGP sessions with a given partner network and went through a field trial working first with one, then another, and then began to expand the list. This deployment approach allowed for incremental activation and, if needed, incremental roll-back of the changes should any problems be encountered.

We started with a few partner networks and one router platform to be able to monitor for any possible ill effects and gradually expanded this circle to include others.

5.2. Publishing

In order to get protection from the RPKI-ROV system, an address-holder must publish ROAs to cover their IP address space indicating which AS(es) are authorized to originate BGP announcements for said address space. Comcast's address space has been issued almost entirely within the ARIN (American Registry for Internet Numbers) region so that was the focus of our considerations.

5.2.1. Hosted vs. Delegated (vs. Hybrid) model

At the time we were planning our deployment, there were two models for publishing RPKI data offered by ARIN and we will discuss each of those options here.
In the "hosted" model an organization generates a public-private key pair and sends the public key to ARIN and receives a "Resource Certificate" which covers the resources that ARIN has issued the organization. The organization is then able to sign ROAs with its corresponding private key and load them each into the ARIN system.
In the "delegated" model, the "parent" registry (ARIN in this case) issues a resource certificate to the "child" which hosts its own CA (Certificate Authority) and can issue ROAs for said resources or even further sub-delegate if desired.

There were (and are at the time of this writing) two open-source software packages which can be used as a CA and to issue ROAs:

- Krill (written in Rust by NLnet Labs)
- rpkid (written in Python2 and C by Dragon Labs)

In considering which model would make the best choice for us, the relative immaturity of these software packages and lack of broad operational experience suggested the best approach would be to use the "hosted" model for the initial rollout at least. Similarly, there had been a few instances operational problems of RIR repositories or CAs and should those happen, it would be preferable to share fate with a large number of RPKI users such that the most operational attention and expertise could be brought to bear on a solution.

One complication to the delegated model is that, as initially conceived, the publication of the ROAs and other RPKI material (manifests, CRLs – Certificate Revocation Lists) is done by the operator of the delegated CA. However, this represents an operational risk and has different availability requirements than the CA itself. Hence the "hybrid" or "publish in parent" approach has been developed and is offered by a number of RIR or LIR issuers. This option was not available from ARIN at the time of our deployment so could not be considered.

In either model, software mechanisms can be brought to bear to automate the issuing and analysis of ROAs against other information such as IP address management systems and routing tables. In both scenarios well-featured APIs (Application Programming Interfaces) are available for these purposes.

5.2.2. Risks

Before we started issuing ROAs for our address space, all our routes would be in the "not-found" status. As we start issuing ROAs, they must ensure they properly match the BGP routes that we advertise to other networks.

For many network operators, particularly many or most Enterprises and perhaps CDN (Content

Distribution Network) operators keeping these elements (BGP announcements and ROAs) aligned might be quite straightforward. However, for Comcast's network, the situation is rather complex. We have something over one hundred IP address blocks issued from ARIN however these are split up across more than twenty ASes, networks, and services. There is also a certain amount of dynamism in the addressing.

Our addressing design, generally, features the large IP address blocks being advertised from our Backbone network (AS7922) and different parts issued among our regional, data center, enterprise and other networks. Hence many different more-specific routes within our blocks would show up to other networks with different origin AS numbers.

5.2.3. Bottom-up ROA creation

Given this routing and addressing design, if we were to issue ROAs for the large blocks with an origin of AS7922, immediately all other routes which were properly visible for reasons of particular traffic flow would become "invalid" to other networks that were doing ROV which was a significant list by the time of our rollout.

Instead, the proper course was to issue ROAs for the various smaller-sized BGP announcements that were expected to be visible externally. (It is normal for a network to carry more-specific announcements internally and Comcast has thousands of these.) We gradually filled in ROAs for these pieces over the course of several weeks, starting very slowly until we had a good level of confidence we could properly detect if there was any problem caused. That detection included monitoring the level of traffic destined for a prefix using NetFlow data collected at our network edges.

Once all the more-specific routes in a given block were complete, we could "top off" by issuing a ROA for that large block and the over-arching advertisement from our Backbone AS. Issuing ROAs to deal with incremental changes after this first large deployment would be simpler and bear little risk.

All this work was considerably aided by software automation for the analysis and ROA issuance. Some was internally developed as well the ARIN ROA request script developed by Rich Compton of Charter Communications which, as the name suggests, uses ARIN APIs to effect bulk issuing (or deletion) of ROAs in ARIN's hosted system.

5.3. Learning

In the course of our deployment, a number of realizations came to light that we share here in hopes that it might be useful in the considerations of others who contemplate a similar course.

5.3.1. Bugs

Almost inevitably, bugs and unforeseen error conditions are discovered in the course of implementing new technology, particularly one as complex as RPKI. We had the good fortune to not be an early adopter and that strategy proved beneficial as many early growing pains had already been found and fixed by others and we appreciate their role in the development.

5.3.1.1. RP software

In some cases, one or another of the RP software packages would yield significantly different results from the other. Generally, these cases resulted from some differences in processing or implementation details sometimes including how they interacted with other infrastructure (e.g., publication points).
There have also been a few vulnerability reports (though none were catastrophic) and patches issued at different rates.

5.3.1.2. Router software

We worked with our router vendors for advice on best current code in the software trains that we use prior to enabling ROV. If there were outstanding bug reports, we examined the circumstances and impacts to help assess the relative risks.

5.3.2. Monitoring and instrumentation

From time to time, on one vendor's router platform we would see occasional situations where the number of prefixes received from the VC servers via RTR sessions would drop to zero and only slowly get repopulated as new VRPs came in. Monitoring the RTR sessions on each router which is configured with them is crucial for detecting and remedying the situation.

Each of our RP software packages comes with the ability to publish metrics for consumption within Telegraph, Prometheus or some similar monitoring solution and then to visualize the data over time using Grafana. This can allow for tracking any anomalous behavior to a given start time.

5.4. Future work

As of this writing, ARIN has begun supporting a “hybrid” or “publish in parent” ROA deployment model where the CA server is housed and managed by the network operator but the publication can be done on ARIN's high-availability Publication Point servers. We will likely consider making use of this which could also simplify administration of ROAs for address space issued by other RIRs.

The IETF SIDROPS (Secure Inter-Domain Routing OPerationS) Working Group is where engineers from industry and other interested parties (e.g., academia, government, etc.) continue to convene to monitor the expanding deployment of RPKI and related technologies and discuss incremental changes or advice that may be appropriate. Also, under development are expanded uses of the RPKI architecture to further improve the security and resilience of the global Internet routing system.

6. Conclusion

The global Internet routing system is critical to the services, commerce, education, and entertainment of the world's population. Over the course of decades it has, through careful engineering and collaboration, managed to grow and scale to meet the changing objectives of its users. The fundamental security model is one piece that needed to evolve but also required backward compatibility for incremental rollout while still realizing benefit with partial implementation. RPKI is the enabling foundation and ROV is the first realization of such improvements.

The operational model of RPKI is reasonably flexible to accommodate many different operational and deployment scenarios, however this malleability also means that deciding exactly how to deploy and in what order can seem daunting. By gradually narrowing options by adoption of guiding principles, it becomes easier to plot the way forward.

Abbreviations

AfriNIC	African Network Information Center
API	Application Programming Interface
APNIC	Asia-Pacific Network Information Center
ARIN	American Registry for Internet Numbers
AS	Autonomous System
ASN	Autonomous System Number
BGP	Border Gateway Protocol
CA	Certificate Authority
CRL	Certificate Revocation List
eBGP	External Border Gateway Protocol
IANA	Internet Assigned Numbers Authority
iBGP	Internal Border Gateway Protocol
IETF	Internet Engineering Task Force
IP	Internet Protocol
LACNIC	Latin America and Caribbean Network Information Centre
LIR	Local Internet Registry
PP	Publication Point
RIPE (NCC)	Réseaux IP Européens (Network Coordination Centre)
RIR	Regional Internet Registry
ROA	Route Origin Authorization
RP	Relying Party
RPKI	Resource Public Key Infrastructure
RRDP	RPKI Repository Delta Protocol
RTR	RPKI To Router protocol
SIDR	Secure Inter-Domain Routing
SIDROPS	Secure Inter-Domain Routing OPerationS
SLURM	Simplified Local Internet Number Resource Management
VC	Validating Cache

Bibliography & References

Rekhter, Y., Ed., Li, T., Ed., and S. Hares, Ed., "A Border Gateway Protocol 4 (BGP-4)", RFC 4271, DOI 10.17487/RFC4271, January 2006

J. Hawkinson and T. Bates. "Guidelines for creation, selection, and registration of an Autonomous System (AS)", RFC 1930, doi:10.17487/RFC1930, March 1996

B. R. Smith and J. J. Garcia-Luna-Aceves, "Securing the border gateway routing protocol," *Proceedings of GLOBECOM'96. 1996 IEEE Global Telecommunications Conference*, 1996, pp. 81-85, doi:10.1109/GLOCOM.1996.586129.

S. Kent, C. Lynn and K. Seo, "Secure Border Gateway Protocol (S-BGP)," in *IEEE Journal on Selected Areas in Communications*, vol. 18, no. 4, pp. 582-592, April 2000, DOI: 10.1109/49.839934

Murphy, S., "BGP Security Vulnerabilities Analysis", RFC 4272, DOI 10.17487/RFC4272, January 2006

Mohapatra, P., Scudder, J., Ward, D., Bush, R., and R. Austein, "BGP Prefix Origin Validation", RFC 6811, January 2013

K. Zetter, "Revealed: The Internet's Biggest Security Hole.," Wired, August 2008.

N. Anderson, "How China swallowed 15% of 'Net traffic for 18 minutes," ArsTechnica, November 2010.

D. Godin, *Traffic Misroutes through China Telecom* https://arstechnica.com/information-technology/2018/11/strange-snafu-misroutes-domestic-us-internet-traffic-through-china-telecom/, ArsTechnica, November 2018

Wikipedia, *BGP Hijacking – Public Incidents*
https://en.wikipedia.org/wiki/BGP_hijacking#Public_incidents

NIST, *Robust Interdomain Routing*, https://www.nist.gov/programs-projects/robust-inter-domain-routing, April 2022

Haag W., Montgomery D., Barker William C., Tan A., *Protecting the Integrity of Internet Routing: Border Gateway Protocol (BGP) Route Origin Validation*, NIST Special Publication, NIST-SP-1800-14, July 2018

Sriram K., Montgomery D., *Resilient Interdomain Traffic Exchange: BGP Security and DDoS Mitigation*, NIST SP 800-189, December 2019

Hannachi L., Sriram K., Borchert O., Montgomery D., *NIST RPKI Monitor*, NIST Software Release, April 2021