Master Thesis

Measuring RPKI Route Origin Validation in the Wild

Andreas Reuter

Matr. 4569130

Supervisor: Prof. Dr. Matthias Wählisch

Institute of Computer Science, Freie Universität Berlin, Germany

January 24, 2018

I hereby declare to have written this thesis on my own. I have used no other literature and resources than the ones referenced. All text passages that are literal or logical copies from other publications have been marked accordingly. All figures and pictures have been created by me or their sources are referenced accordingly. This thesis has not been submitted in the same or a similar version to any other examination board.

Berlin, January 24, 2018

(Andreas Reuter)

Abstract

Abstract

A proposal to improve inter-domain routing security—Route Origin Authorization (ROA)—has been standardized. Prefix owners can use ROAs to authorize specific autonomous systems to legitimately originate their IP prefixes. More and more networks are using ROAs to secure their prefixes, but little is known about whether BGP routers actually validate received routes against these ROAs, a process known as Route Origin Validation (ROV). It is unclear which networks blindly accept illegitimate routes, which reject them outright, and which de-preference them if legitimate alternatives exists.

In this thesis, we first clarify the problem space and revisit the state of the art approach, which attempts to use uncontrolled experiments to characterize ROV adoption by comparing legitimate and illegitimate routes [39]. We examine the limitations of this approach and show that it can lead to a high rate of false positives. Our measurements suggest that routing observations attributed to ROV are likely to be caused by (non-security related) use of traffic engineering techniques. Furthermore, we argue that gaining a solid understanding of ROV adoption is impossible using currently available data sets and techniques. We introduce a new, verifiable methodology which improves upon the state of the art by leveraging data from controlled experiments. With our approach, we conduct various experiments aimed at testing different ROV-related routing policies and present three AS that do implement ROV, confirmed by operators.

Acknowledgments

I would first like to thank my thesis advisor Matthias Wählisch for the continuous support, patience, and guidance. He has provided many useful suggestions throughout this work, and has consistently steered me in the right direction. I am grateful for the opportunities he has given me and look forward to working with him in the future.

Furthermore I would like to thank Thomas Schmidt, Randy Bush, Ethan Katz-Bassett, and Italo Cunha for many useful discussions and suggestions.

I am grateful to Ethan Katz-Bassett and Italo Cunha for letting us use the PEERING testbed, without which the experiments described in this thesis would not have been possible.

I would also like to thank Randy Bush for providing us with IP prefixes for the experiments, helping us set up a child RPKI CA, as well as providing valuable insights from the operator-side of networking.

Contents

CHAPTER 1

Introduction

1.1 Motivation

The Internet today connects billions of devices and is integral to many vital functions in our society. With its enormous growth, the Internet has gained great importance for almost all branches of industrial and social interaction and has been classified as critical infrastructure in a number of countries [31, 72]. As we come to rely more and more on digital connectivity, the potential damage of an attack on the Internet's backbone infrastructure increases. Already we have seen governments and private organizations exploit vulnerabilities in the fundamental routing protocol of the Internet to cause large monetary damage and to enforce censorship on their population.

The Internet is a network of networks. Each network is a so called *autonomous system* (AS), identified by its *autonomous system number* (ASN). An AS on the Internet operates a set of IP address that can be assigned to devices within the AS. These IP addresses are aggregated in *IP prefixes*. AS announce the IP prefixes they own to other AS using the Border Gateway Protocol (BGP). The receiving AS can then choose to further propagate the announcements to other neighboring AS. The Border Gateway Protocol is built on trust, since at the time of its design the Internet consisted of only a handful of AS which were all cooperative. This means that when an AS announces that it owns a certain IP prefix, other AS have no way of knowing whether this is true or not. This makes BGP, and with it the Internet, an easy target for malicious AS to disseminate false routing information to disrupt or divert traffic flow.

Attacks on the Internets backbone infrastructure via BGP have become more frequent in recent years [21, 1, 17]. While the vast majority of these attacks affected IP prefixes with relatively little incoming traffic, there were some incidents involving prefixes of popular web services such as YouTube that affected a large number of users. In order to secure BGP against these attacks, the Secure Inter-Domain Working Group (sidr) was formed by the IETF [37]. This working group designed a solution called BGPsec [50], which uses cryptographic operations on BGP routers to prove the authenticity and integrity of BGP update messages. The Resource Public Key Infrastructure (RPKI) is the framework necessary to support BGPsec. Since BGPsec requires cryptographic operations on the BGP routers

themselves, it is costly to implement and its deployment is still far off. In the meantime, the RPKI can be used to partially secure BGP by offering a trusted mapping between Internet resources, *i.e.,* IP prefixes and ASN, and the organization that own them. It allows resource owners to authorize autonomous systems to originate their IP prefixes. Other AS receiving the announcement can use the RPKI objects to check whether the announcement is legitimate, a process called *Route Origin Validation*. AS can then choose at their own discretion whether to drop illegitimate announcements.

1.2 Objective

There exists a plethora of research on the current adoption of the RPKI by Internet resource owners [69, 70, 47, 9, 62, 42]. This research mostly focuses on the deployment of RPKI objects such as Route Origin Authorizations (ROA) and Resource Certificates. However, the creation and deployment of these objects alone does not lead to a more robust Internet infrastructure. The semantics of these object is i) an attestation of ownership and ii) an authorization of usage of Internet resources. In order to protect against AS that illegitimately announce IP prefixes, the actual BGP routers on the Internet need the information in the RPKI object and consider it in their routing decision. This is called Route Origin Validation (ROV) and without its deployment the RPKI is ineffectual in securing BGP. There has been almost no work published on the deployment and usage of ROV on BGP routers. The goal of this thesis is to measure the adoption of ROV. Specifically whether any AS on the Internet have deployed ROV on any of their BGP routers and whether they are using validation results in their routing policy.

1.3 Contributions

This thesis makes a number of contributions to achieve the thesis objective:

- Clarify the problem space of measuring ROV.
- Analysis of the methodology and results of existing work.
- Presentation and examination of the limits of the existing methodology.
- Presentation of a new methodology that addresses the limits of existing work.
- Conducting experiments based on the new methodology to achieve the thesis objective.
- Set up a longitudinal study to monitor the deployment of ROV on the Internet.

1.4 Thesis Structure

Chapter 2 of this thesis explains the technical background necessary to understand Internet backbone routing and route origin validation. Chapter 3 introduces the challenges in measuring ROV while Chapter 4 analyses the current state of the art methodology in detail. Chapter 5 then presents our new methodology using active experiments. Chapter 6 gives an overview of the various tools developed for this work. Chapter 7 discusses work related

to measuring BGP and the RPKI, while Chapter 8 concludes this work and discusses future work.

CHAPTER 2

Technical Background

This chapter describes the relevant technical foundations of the Internet that are needed to understand the rest of this thesis. It provides definitions for various key concepts.

2.1 Inter-Domain Connectivity

2.1.1 Internet Protocol

One of the most fundamental protocols for the operation of the Internet is the *Internet Protocol* (IP). The Internet Protocol was designed to provide the functionality necessary to deliver data from a source to a destination across an interconnected system of networks such as the Internet [60]. There are two versions of the Internet Protocol in use today, IPv4 and IPv6. The source and destination of the data being delivered are specified with an *IP address*, a sequence of 32 bits for IPv4 and 128 bits for IPv6. For the sake of simplicity, this thesis will use IPv4 in any examples given.

IP addresses can be aggregated into so called *IP prefixes*. An IP prefix consists of an IP address and a prefix length. An IP prefix with the IP address A and prefix length l contains all IP addresses whose first l bits are identical to the first l bits of A. Various examples can be found in Table 2.1.

IP prefix	**IP addresses**	**No. of addresses**
192.168.1.1/32	192.168.1.1	1
192.168.1.2/31	192.168.1.2 192.168.1.3	2
192.168.1.0/24	192.168.1.0 - 192.168.1.255	256
192.168.0.0/16	192.168.0.0 - 192.168.255.255	65536

Table 2.1: Examples of IP prefixes and the IP addresses they contain.

2.1.2 Autonomous Systems

The Internet is a collection of inter-connected *Autonomous Systems* (ASes), with the purpose of facilitating communication between devices across systems by achieving reachability of IP prefixes. An Autonomous system (AS) is an independent network on its own. Some AS are globe-spanning networks consisting of thousands of devices, and are connected to hundreds of other AS. Other AS may only consist of few devices, and only connect to one other AS. Each AS is identified by its unique *Autonomous System Number* (ASN). In RFC-1930, an autonomous system (AS) is succinctly described as "a connected group of one or more IP prefixes run by one or more network operators which has a single and clearly defined routing policy" [43]. This definition hinges on the term *routing policy*. A routing policy pertains to how an AS exchanges routing information with other AS and can be defined by the network operator. In practice this definition does not always hold and abstracting an AS to a atomic unit while ignoring its internal structure is an oversimplification [25, 63, 73]. An AS might use various routing policies, dependent on different parameters such as economic relationships with other AS.

Management of Internet Resources

Before an organization can form an AS and connect it to other AS, they must first acquire IP prefixes and an ASN, which are referred to as *Internet Resources*. An organization running an AS must obtain the necessary Internet resources either from a *Regional Internet Registry* (RIR) or third party organizations that have obtained their resources from a RIR. There are 5 RIRs, each of them responsible for a geographical region of the planet and each of them administrating a portion of IPv4 and IPv6 IP space, as well as a subset of ASNs. Table 2.2 shows the names and geographic regions of the RIRs.

RIR	**Regions**
AFRINIC	Africa
ARIN	North America
APNIC	Asia, Australia, New Zealand
LACNIC	South America
RIPE	Europe, Russia, Middle East, Central Asia

Table 2.2: The five RIRs and their administrative regions.

In the definition of a routing policy we have stated that AS exchange routing information with each other. The exchange of routing information between AS is crucial to achieve connectivity across the Internet. To exchange this information, network operators use the Border Gateway Protocol.

2.2 Border Gateway Protocol

The purpose of the Border Gateway Protocol (BGP) is to facilitate the exchange of network reachability information [61]. Two BGP speaking routers connect using TCP and exchange network routing information using BGP Update messages. We say that two connected BGP routers are **peers**. Two AS which have at least one BGP connection between can also be said to be **peers**. The distinction between two routers peering and two AS peering should be made obvious from context.

2.2.1 BGP Updates

A BGP Update message may serve two kinds of purposes: Advertisement of a route or withdrawal of a route. A route in the context of BGP is defined as "a unit of information that pairs a set of destinations with the attributes of a path to those destinations" [61]. Since an understanding of BGP Update messages is integral context for this thesis, we provide a detailed description of their purpose and content.

Advertisement

An advertisement specifies one or more network destinations (*i.e.,* IP prefixes). Advertisements contain a set of path attributes that all destinations have in common. The destinations are contained within the *Network Layer Reachability Information* field. Route announcements are used by an AS to inform its neighbors about the IP prefixes associated with the AS. These announcements may then be propagated its neighboring AS to spread the reachability information throughout the Internet. Note that throughout this thesis the terms *advertisement* and *announcement* are used interchangeably. Here is a list of path attributes that are relevant in the context of this thesis:

Origin
: The origin path attribute specifies whether the reachability information contained in the advertisement pertains to the interior of the sender AS or to an external AS. It is also possible that the reachability information was injected from another protocol. This is a well-known, mandatory attribute.

AS Path
: The AS path attribute represents a sequence of autonomous system numbers (ASN). An BGP router propagating an advertisement will prepend its own ASN to this attribute before sending it to its peers. An exception to this is when the advertisement is propagated to an internal peer. The AS path is the reverse sequence of AS that a advertisement has traversed thus far. Thus the right-most AS on the AS path is the **Origin AS** for the network destinations. It is also possible for multiple ASN to occupy the same position on the AS path by using *AS sets*. This is a well-known, mandatory attribute.

Next Hop
: This attribute defines the IP address of the router to which traffic should be forwarded to for the destinations contained in the network layer reachability information field. This is a well-known, mandatory attribute.

Multi-Exit Discriminator
This is an optional attribute that can be used on inter-AS connections to discriminate between multiple exit or entry points towards the same neighboring AS.

Local Preference
A BGP speaker will calculate its preference for each route it receives, based on its local routing policy. If the BGP router propagates the route, it will pass on its local preference for the route only in the case that the peer it is propagating the route to belongs to the same AS.

Communities
This attribute allows a BGP router to convey additional information to its peers about the set of destinations contained in an advertisement. This can be used to simplify otherwise complex routing policies by 'tagging' sets of destinations with meta information [32].

There is a number of additional path attributes that a BGP router might use such as Aggregator and Atomic Aggregate. These attributes however are not relevant in the context of this thesis and have been omitted.

Withdrawal

A withdrawal of a route specifies **only a network destination** and does not carry any path attributes. Withdrawal messages can be used by routers to inform their peers that a previously advertised route has become unavailable. If the connection between two BGP routers is lost, all routes exchanged between them are implicitly withdrawn. It is possible for a BGP Update message to carry one or more route advertisements as well as one or more route withdrawals. It is important to note that **BGP routes do not expire**. A route will be considered available as long as the peer from which the route was received has not withdrawn it.

2.2.2 BGP Route Selection

The most important design decision that allows BGP disseminate routing information on a scale as big as the Internet is that it severely limits the information that is being passed between BGP speakers. For any given prefix, a BGP router will select only one route, the best route, out of all available route. We refer to the process of choosing this route as *best route selection* or *best path selection*. This route may, but not must, then be passed on to peers of the router. All other available routes are not passed on. All routing information a BGP router obtains is stored in the *Routing Information Base* (RIB), which consists of three separate parts:

Adj-RIB-In
The Ajd-RIB-In stores all routing information received by the BGP router. More specifically, it stores all routes learned from inbound BGP update messages. It represent the complete database of available routing information that was received by the router. This means that it can contain multiple routes for the same network destination. The contents of the Adj-RIB-In can then be used to select the best routes for

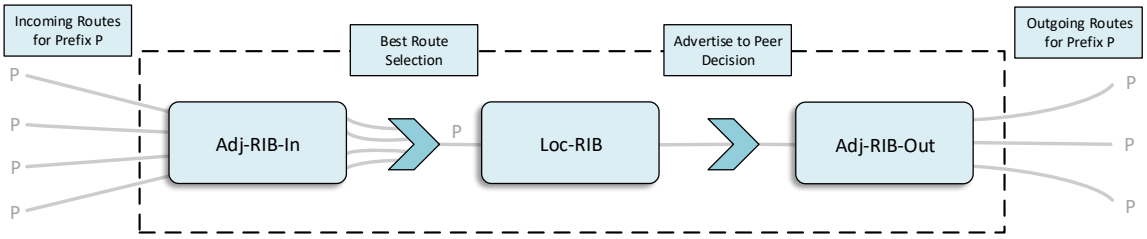


Figure 2.1: Relationship between Adj-RIB-In, Loc-RIB, and Adj-RIB-Out.

each network destination.

Loc-RIB

The Loc-RIB contains all routes that were selected by the BGP router. It is the result of applying the local inbound routing policy to the contents of the Adj-RIB-In to select the best available paths for network destinations. This means that for each network destination it contains at most one route, namely the one that local policy determined to be the best available route. The Loc-RIB is thus a subset of the Adj-RIB-In. The contents of the Loc-RIB are part of the actual forwarding table of the router.

Adj-RIB-Out

The Adj-RIB-Out stores all routing information that the BGP router has selected for advertisement to its peers. This does not mean that all routes contained in the Adj-RIB-Out are advertised to all peers. The routing information in the Ajd-RIB-Out is the result of applying the local out-bound routing policy to the contents of the Loc-RIB. This means that the Adj-RIB-Out is a subset of the Loc-RIB.

Note that the RIB described here is a conceptual model and BGP implementations need not necessarily maintain these Adj-RIB-In, Loc-RIB, and Adj-RIB-Out as separate data structures.

To select the best route for a prefix, a BGP router will consider all routes for this prefix that are inside the Loc-RIB. First, the BGP router will calculate the degree of preference for each route. If the route was learned from an internal peer, either the local preference sent by that peer will be taken as the degree of preference, or the router will calculate the degree of preference based on its own local routing policy. In the case that the route was learned from an external peer, *i.e.,* a router belonging to a different AS, the degree of preference will always be calculated using the local routing policy. After the degree of preference has been calculated for each route, the router selects the route with the highest preference. In case of a tie in degree of preference, RFC4271 specifies the following tie-breaking rules:

1. Shortest AS Path
2. Origin Type: Prefer internal over external.
3. Preferred MED (Only applicable if routes are learned from same AS).
4. Prefer routes learned externally over routes learned internally.
5. Prefer route from peer with lowest BGP Identifier value.
6. Prefer route learned from peer with lowest IP address.

(a) AS100 originates 203.0.113.0/24. AS400 chooses the route to AS100 via AS200.

(b) AS666 also originates 203.0.113.0/24. AS400 chooses the shorter route to AS666 instead of to AS100.

Figure 2.2: AS666 hijacks 203.0.113.0/24 by originating the prefix. AS400 accepts the bogus route because it is shorter than the legitimate one.

2.2.3 BGP Security Issues

BGP is based on trust, as any AS can advertise any IP prefix to its neighbors. There is no mechanism built into BGP that would allow a network operator to check the legitimacy of received announcements. The lack of such a feature is at the core of the security problems of BGP. The two main attack vectors are:

Lack of Authorization
A BGP router can originate advertisements for IP prefixes that do not actually belong to its AS. If the actual owner AS also advertises these prefixes, other BGP routers will then receive the two competing advertisements and choose whichever route is more attractive. This will lead to some routers choosing the illegitimate route, which will lead to network traffic being sent to the wrong AS. This is typically referred to as a *prefix hijack* attack.

Lack of Authentication
A BGP router cannot determine whether a received BGP update has been altered or not. This means that an attacker can manipulate path attributes in order to influence the path selection process of other routers. For instance, an attacker might manipulate the AS path attribute, artificially inserting themselves into a route in order to divert traffic via their AS.

The impact of prefix hijack attacks can vary greatly. A hijack of a prefix that is being used for a popular service can cause large scale outages of that service, while a hijack of a prefix with no significant traffic will largely go unnoticed in the global Internet. There have been a number of incidents involving prefix hijacks that have caused a significant number of users to lose connectivity to popular services such as the YouTube incident [21] or the China Telecom incident [1]. More recently, prefixes belonging to a number of large financial institutions were hijacked by Rostelecom [17].

2.2.4 AS Relationships

To establish connectivity to the global network, an AS must ensure that (*i*) all other AS have a route to the IP prefixes of the AS, and (*ii*) the AS has a route to all IP prefixes of other AS. A straightforward way to achieve this is to establish a peering session with all other AS and exchange routes directly. This does not scale. Instead, there exist several very large, well connected AS that offer transit connectivity for smaller AS. The larger AS acts as a provider to the smaller AS. This involves propagating announcements for IP prefixes of the smaller AS to its many neighbors. These neighbors are often also large provider AS and will disseminate the routing information to their neighbors as well. This way the smaller AS can ensure that its prefixes are reachable without having to establish many peering sessions. The provider AS will also offer routes for all other prefixes to the customer AS to ensure that devices in the customer AS can reach the Internet. This *customer-provider* relationship is the most prevalent on the Internet, with the majority of AS being quite small in size and purchasing connectivity from a small number of very large AS. These large AS will often establish peering sessions between them to enhance their connectivity. We call these relationships *peer-to-peer* relationships, since both AS will gain connectivity to other ones customer AS. While in a customer-provider relationship the flow of money is quite obvious (the customer pays the provider). In a peer-to-peer relationship there is not always a monetary flow between the two involved AS. An exception to that is *paid peering*, whereas AS agree to only provide partial transit to each other. The Gao-Rexford model [38] characterizes the roles of customer, provider, and peers states rules that pertain to exporting routes:

Exporting routes to a provider AS
: An AS may advertise routes for its own prefixes and the prefixes of its customers to its provider AS. It may *not* export routes learned from other provider or from peers. In other words, a provider will provide **transit** for its customer AS and the customers of its customers, but not for other providers or peers of its customers. Transit means an AS offers access to its peers, customers, and providers to another AS [41].

Exporting routes to a customer AS
: An AS will export routes for its own prefixes, and routes for prefixes from its providers and peers, to its customer AS. In other words, an AS will provide transit to its customer AS.

Exporting to a peer
: An AS may export routes for its own prefixes and the prefixes of its customers to its peers. It may *not* export routes for prefixes from its providers or from other peers to its peers. In other words, an AS will not provide transit service for its peers.

While the rules can be helpful to reason about AS relationship observed on the Internet, it is important to note that this model is a simplification and does hold for all AS relationships.

2.2.5 Internet Exchange Points

An Internet Exchange Point (IXP) is a facility where multiple AS can publicly or privately interconnect. An IXP provides a switching infrastructure that enables layer 2 connectivity between all member AS.

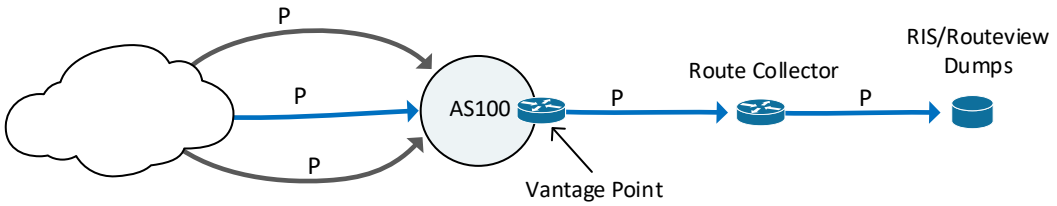


Figure 2.3: The vantage point receives three different announcements for prefix P. It selects one route and exports it to the route collector. The route collector dumps the received routes periodically.

Many IXPs offer so called *route servers*. Route servers are BGP speakers that can be used to disseminate routing information between a large number of AS. Instead of having each AS peer with every other AS, all AS peer with the route server and announce their routes to it. The route server will propagate received routes to all of its peers. It is important to note that a classic route server is simply a BGP router. This means that the route server performs best path selection. In case there are multiple AS at the IXP announcing a route for the same prefix to the route server, only one of them will be selected to be propagated to the other peers.

2.2.6 Measuring BGP

Measuring BGP in the context of this thesis involves analyzing the BGP updates a router receives and exports to infer its routing policy or the routing policy of other routers that propagate the updates. In order to analyze these it is first necessary to obtain this information. For the vast majority of BGP routers on the Internet, this information is not made publicly available. However, the Routeviews [20] and RIPE RIS [16] projects operate a number of BGP routers whose RIBs are periodically dumped and available to download. Since this thesis uses data from these two projects, it is important to clearly define what type of data we are referring to and how it was obtained.

The Routeviews and RIPE RIS projects operate so called *route collectors*. A route collector is a BGP router which periodically dumps all BGP updates received from its peers. This information is dumped in two formats, (*i*) the raw BGP update messages received by the route collector and (*ii*) the contents of the Ajd-RIB-In, which contains all routes learned via received BGP update messages. A peer of a route collector is called a *vantage point*, or a *monitor*. The Routeviews and RIPE RIS projects peer with a combined number of 960 vantage points. Through a route collector we get insight on which paths the vantage points have chosen to export. It is important to understand that a vantage point only sends the selected best route for a prefix to a route collector, not all available routes, as illustrated in Figure 2.3. It is also not guaranteed that a vantage point sends *all* selected best routes to a route collector. It is possible that some vantage points chose to export a subset of selected routes. We say a vantage point *provides*, *exports*, or *chooses* a route, if the route was selected as best route sent to a route collector. In an effort to make monitoring and measuring of BGP easier, the BGP Monitoring Protocol (BMP) [66] has been standardized. BMP can be used by BGP routers to export not just their selected route, but all available

routes for a prefix, for monitoring purposes. A BMP infrastructure where vantage points would send all available routes to route collectors using BMP would greatly diminish the problem of limited visibility and make BGP measuring substantially easier. Unfortunately, as of time of writing such an infrastructure does not yet exist.

2.3 Resource Public Key Infrastructure

The Resource Public Key Infrastructure (RPKI) is part of an effort to secure BGP against the attacks described in section 2.2.3. The RPKI uses X.509 public key certificates [34] to attest ownership of Internet resources. In this context Internet resources are IP addresses and AS numbers. The certificates used in the RPKI are called resource certificates [45] and are extended to contain IP prefixes and AS numbers that are allocated to the owner of the resource certificate. Resource certificates can be validated using the same certificate path validation algorithm as regular X.509 certificate, with one additional step: A resource certificate needs to contain a subset of the resources contained within the parent resource certificate. In the case of IP addresses, a resource certificate may not contain any IP addresses that are not contained within the parent resource certificate. In the case of AS numbers, a resource certificate may not contain any AS number that is not contained in the parent resource certificate. This validation rule means that the certificate tree of resource certificates mirrors the hierarchical nature of IP space ownership. Each RIR operates a self-signed resource certificate which contains all resources owned by that RIR. These self-signed resource certificates are also referred to as *trust anchors*.

Resource owners can use their resource certificates to authorize specific AS to legitimately originate their IP prefixes. This is done by issuing a Route Origin Authorization (ROA).

2.3.1 Route Origin Authorization (ROA)

A ROA is a cryptographic object, designed specifically for the RPKI. It contains a *End-Entity* resource certificate. The key distinction of an EE resource certificate is that it can not be used to issue further resource certificates. Thus, EE resource certificates are always a leaf in the certificate tree. The EE resource certificate is used to validate a ROA. Aside from the EE resource certificate, a ROA also contains these fields:

IP prefixes This field contains a list of tuples of IP prefixes and a maximum prefix length. An IP prefix together with a maximum length define a range of prefixes that this ROA is authorizing an AS to originate.

ASN This field contains exactly one ASN. The ROA is authorizing this AS to legitimately originate advertisements for the IP prefixes listed in the *IP prefixes* field.

For example, the ROA in Table 2.3 authorizes AS100 to originate announcements for prefixes 198.51.100.0/24, 198.51.100.0/25, and 198.51.100.128/25. The ROA in Table 2.4 authorizes AS500 to originate announcements only for prefix 192.0.2.0/24.

The RPKI defines various other objects, such as manifests and certificate revocation list. Since these objects are not relevant for the work presented in this thesis, we chose not to cover them. An interested reader can find further information in RFC6486 and RFC6487.

ROA	
Prefix:	198.51.100.0/24
Max. Length:	25
ASN:	100

Table 2.3: ROA authorizing AS100 to originate 198.51.100.0/24-25

ROA	
Prefix:	192.0.2.0/24
Max. Length:	24
ASN:	500

Table 2.4: ROA authorizing AS500 to originate 192.0.2.0/24-24

2.3.2 Route Origin Validation

The information contained in ROAs can be used by BGP routers to validate incoming BGP announcements. Specifically, they can check whether the AS that originated the advertisement is authorized to do so by a ROA. This process is called *Route Origin Validation* (ROV). ROV can yield three different results:

Valid

The RPKI validity state of a route is **valid** if there exists a ROA which contains an IP prefix that matches the announced prefix and the following holds:

1. The length of the announced prefix does not exceed the maximum length of the matching prefix in the ROA. In this case the route is invalid because of a **length mismatch**.
2. The AS originating the announcement matches the ASN in the ROA. In this case the route is invalid because of an **AS mismatch**.

Invalid

The RPKI validity state of a route is **invalid** if for *every* ROA which contains an IP prefix that matches the announced prefix *one* the following holds:

1. The length of the announced prefix exceeds the maximum length of the matching prefix in the ROA.
2. The AS originating the announcement does not match the ASN in the ROA.

Not Found

The RPKI validity state of a route is **not found** if there exists no ROA which contains an IP prefix that matches the announced prefix.

We say a prefix P_1 **matches** a prefix P_2 if either P_1 and P_2 are identical or P_2 is a sub-prefix of P_1. In the course of this thesis, we may refer to the **not found** RPKI validity state also as **unknown**. Table 2.5 shows several pairs of ROAs and announcements together with the ROV result.

ROA	**BGP Announcement**		
	Prefix: 192.0.2.0/24 Origin AS: 500	Prefix: 192.0.2.0/20 Origin AS: 500	Prefix: 192.0.2.0/16 Origin AS: 300
ROA_1 Prefix: 192.0.2.0/24 Max. Length: 24 ASN: 500	Valid	Not Found	Not Found
ROA_2 Prefix: 192.0.2.0/16 Max. Length: 16 ASN: 500	Invalid Length	Invalid Length	Invalid AS
ROA_3 Prefix: 192.0.2.0/16 Max. Length: 24 ASN: 500	Valid	Valid	Invalid AS
$ROA_1 \cup ROA_2 \cup ROA_3$	Valid	Valid	Invalid AS

Table 2.5: RPKI validity states for routes and ROAs

Route origin validation is also sometimes referred to as BGP Prefix Origin Validation.

ROV in Routing Policy

RFC 6811 specifies that a proper ROV implementation should evaluate each route received from other BGP speakers via BGP update messages. It should also evaluate any route the router has received through other means such as redistribution through other routing protocols or locally defined static routes [54]. The evaluation itself will simply annotate a route with its RPKI validity state, but not exclude the route from the Adj-RIB in [29]. A network operator can use the local routing policy of a router to determine if routes should be treated differently according to their RPKI validity state. One such policy might be to simply exclude all invalid routes from the best route selection process. We refer to the exclusion of a route from the BGP route selection process as **dropping**, **filtering**, or **discarding** of the route. If all IP prefixes were secured within the RPKI, filtering all invalid routes is the intended application of ROV. In that case it would prevent any prefix-hijacking attacks, as only valid routes would be propagated. This would require every resource owner to issue ROAs for their prefixes, and to deploy ROV on all of their BGP routers on the Internet. However, as of now only 8% of prefix/origin pairs have been secured with a ROA [9] and operator gossip suggests that not many, if any, AS have deployed ROV on their routers and are actually dropping invalid routes. Dropping invalid routes altogether is not the only application of the validation results. Previous work suggest that [69, 47] many invalid routes are the result of misconfiguration rather than prefix hijacks. This means that a policy that drops invalid routes altogether might actually cause an AS to lose connectivity to certain prefixes. A more cautious policy might be to simply lower the preference of invalid routes, or to always prefer non-invalid routes over invalid routes for any prefix.

2.3.3 Global RPKI

The entirety of all RPKI objects are hosted in publicly available repositories. The vast majority of objects reside in repositories hosted by the five RIRs, however, it is also possible for owners of Internet Resources to host their own repository as long as it is publicly available. Cache servers periodically download all RPKI objects and cryptographically validate them. All ROAs that pass validation are then made available to BGP routers using the RTR protocol [28]. Any BGP routers with a proper ROV implementation will periodically download the latest validated ROA payloads from the cache server, which they can then use to validate routes.

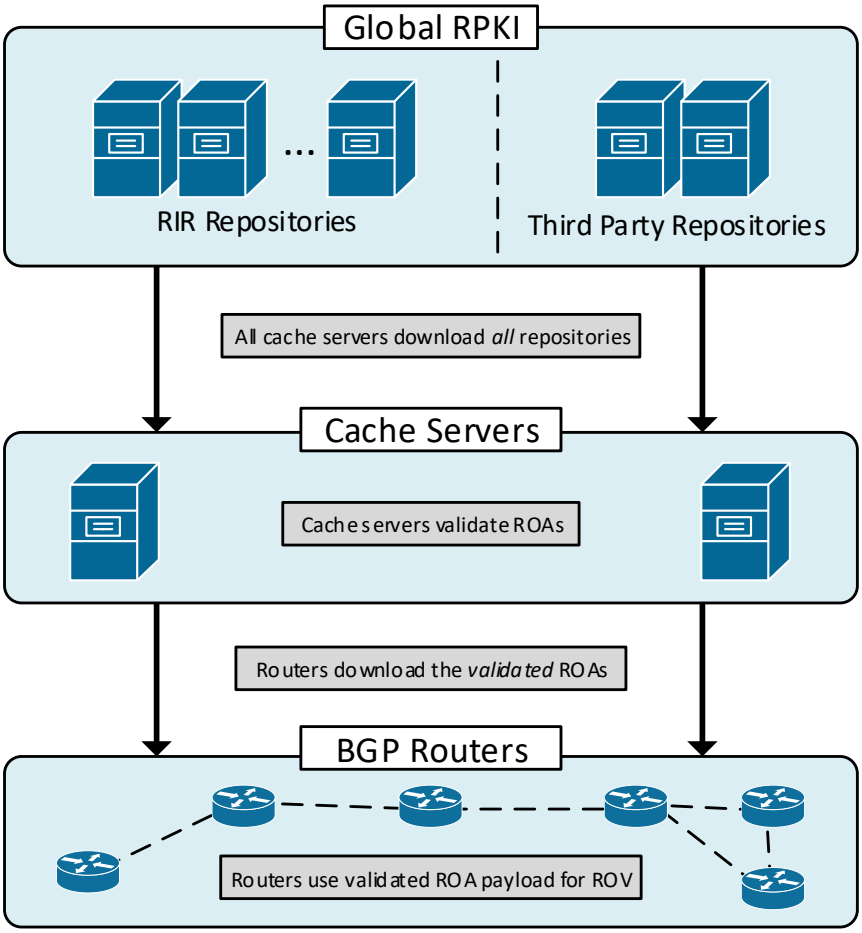


Figure 2.4: An overview of the RPKI infrastructure.

CHAPTER 3

Challenges

There are two major challenges that we face as researcher when it comes to measuring the adoption of ROV on the Internet: Limited Visibility and Limited Control. They are are not specific to measuring ROV, but apply to any BGP measurement that aims to infer routing policy. Since in order to successfully and reliably measure the adoption of ROV and its usage in local policy an approach is needed that deals with both of these challenges, in this chapter we will first analyze the problems they pose further.

3.1 Limited Visibility

The Internet is a complex network consisting of close to 60,000 AS, each of them having deployed at least one BGP speaking router. The flexible nature of BGP allows for intricate relationships between AS. The customer-to-provider and peer-to-peer relationships described by the Gao-Rexford model (see Chapter 2) can not always adequately explain routing decisions made by individual AS. As researchers, there are two key mechanism we can use to get insight on routing decisions:

Looking Glasses
: Public BGP router interfaces made available by individual AS, which can be used to check received and selected routes.

Route Collectors
: Vantage points exporting their selected best routes to public route collectors.

As of now, there is no publicly available unified interfaces for looking glasses, although there are efforts working towards this [40]. The lack of such an interface makes it tedious to obtain routing information from looking glasses in an automated fashion, and thus they are of limited use for large scale analysis. However, looking glasses can still be used to analyze specific events of interest. This leaves us with the data that vantage points export to public routing collectors. Because of the information-hiding nature of BGP, the data obtained from these vantage point is also limited only to the selected best routes the vantage points have chosen for a given prefix. In addition to that, some of these vantage points might only send a partial feed of BGP update messages to the collector, making the information obtained

incomplete. It is however not reliably known which vantage points send a full and which send only a partial feed.

3.1.1 Vantage Point Visibility

Every vantage point has a different view of the AS-level Internet, none of them have a complete view of all AS paths. Figure 3.1 shows for each vantage point the number of different prefixes it has a route for (top) as well as the number of distinct origin AS it has a route to (bottom). The vantage points are on the x-axis, ranked by the number of prefixes they have a route for. We can categorize the vantage points in roughly three groups. The vantage point in x=[0,275] see a large number ($>$ 600,000) of prefixes. These vantage points have an almost 'global' routing table, by which we mean that they have a route for almost all prefixes and do not rely on default routes. Another group are the vantage points in x=[300,575]. When compared to the first group, these vantage points have very limited visibility. They have routes for 30,000 to 40,000 prefixes, less than 10% of the 'global' routing table. The third group is the vantage points in x=[576,960]. These vantage points have generally very low visibility, but are more diverse than the other groups. This group of vantage points has routes to 1 to 10,000 prefixes. We can see the same grouping in the bottom plot of Figure 3.1, albeit with more noise. Generally, the number of distinct origins a vantage point has routes to correlates with the number of prefixes the vantage point has a route for.

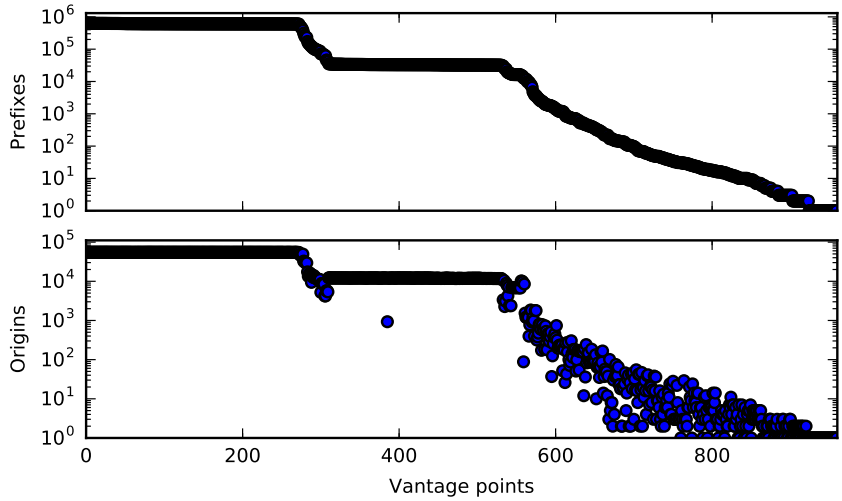


Figure 3.1: Vantage Point Visibility: Number of prefixes (top) and origin AS (bottom) seen by RIPE RIS and Routeviews vantage points.

This large diversity in prefix and origin AS visibility amongst vantage point can have a major impact when analyzing routes exported by these vantage points. This is relevant not just in the context of this thesis, but any work aiming to measure a BGP-related phenomenon using control-plane data. Note that while some approaches in measuring BGP make use of data plane platforms such as RIPE Atlas [55], the challenge of low visibility persists in that domain as well and might even be exasperated by usage of tunnelled connections that

obscure borders between AS [46].

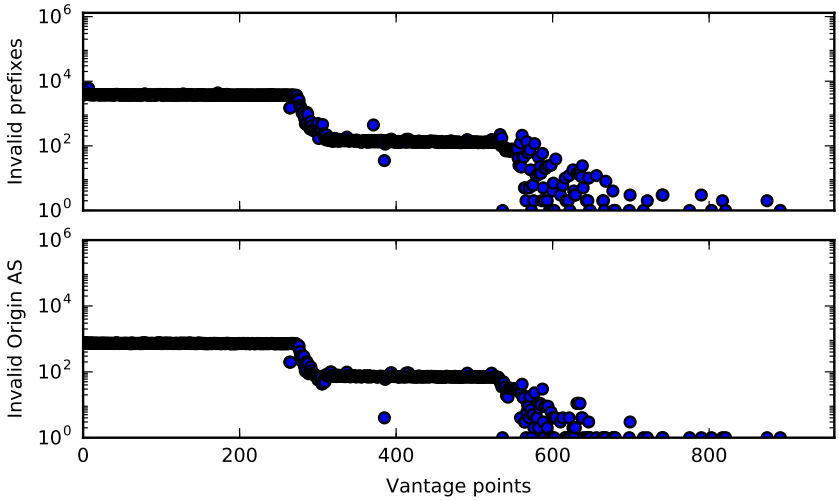


Figure 3.2: Vantage Point Visibility of Invalid Routes: Number of prefixes (top) and origin AS (bottom) seen by vantage points.

When measuring the adoption of ROV-based filtering policies we are of course interested in routes of invalid RPKI status. Analogue to Figure 3.1, Figure 3.2 shows the number of prefixes vantage points exported an *invalid* route for (top) and the number of distinct origin AS these routes lead to (bottom). For both prefixes and origins we can see that the general pattern from Figure 3.1 persists, albeit smaller in magnitude and with more noise. The same three groups that we have seen previously are also apparent here. Note that some vantage points in the third group (x=[576,960]) do not even export one invalid route, while many others export less than 10. This is also reflected in their low number of origin AS with invalid routes to. This essentially cuts down the number of available vantage points to study existing invalid routes from down from 960 to 646.

Lastly, not all of these vantage points have invalid routes for the same prefixes, as in the general case. As discussed in Chapter 2, a route can be invalid because of multiple reasons: Mismatch of origin AS, mismatch of prefix length, or both. Figure 3.3 breaks down all invalid routes that vantage points have exported by their reason for invalidity. As reference, the top plot shows again the number of prefixes the vantage points exported invalid routes for. The bottom plot shows for each vantage point the breakdown of invalid routes by their reason for invalidity. Note how the three groups of vantage points are again present in this data. We see that the group of vantage points with high visibility show a relatively consistent breakdown, with almost 60% of invalid routes being invalid because of a length mismatch, 34% being invalid because of an origin AS mismatch and the remaining routes being invalid for both those reasons. But even amongst the vantage points with a near 'global' routing table, we can see variations between vantage points. For instance, vantage point [198.32.176.10, AS14361] and [198.32.195.24, AS15301] have almost the same number of invalid routes, 4055 and 4066 respectively. However, their numbers of invalid routes with both a length mismatch and an origin AS mismatch are 662 and 385 respectively. When

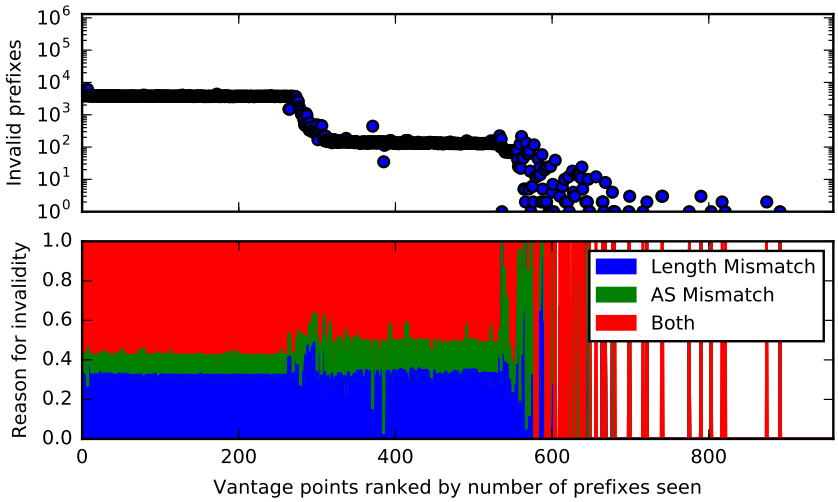


Figure 3.3: Vantage Point Visibility of Invalid Routes: Number of prefixes (top) and the reasons for invalidity (bottom) seen by vantage points.

looking at the second group of vantage points (x=[300,575]), we can see the same general pattern as with the first group, however with a lot more noise. The fraction of invalid routes with both length and origin AS mismatch is fluctuating more amongst these vantage points. Vantage points x=386 and x=372 show only a small fraction of invalid routes having an origin AS mismatch, significantly differing from the fraction of vantage points with similar numbers of (not just invalid) routes. Since the third group of vantage points have so few invalid routes, the breakdown into reason for invalidity fluctuates erratically between vantage points. From those vantage points who have invalid routes in the third group, we can see that a mismatch in length seems to be the predominant reason for invalidity.

3.1.2 Origin-Prefix Completeness

Figure 3.3, 3.2 and 3.1 underline how diverse the different 'views' of the AS-level Internet can be. We see that vantage points differ significantly in the number of prefixes they have routes to, the type of routes of those routes (reason for invalidity), and the number of AS originating prefixes. We are also interested in the nature of prefix and origin AS visibility: Does a vantage point generally have routes for *all* prefixes that an AS originates, or just a subset? In other words: What fraction of all prefixes originated by an AS does a vantage point see? We define this as *per-origin prefix visibility*. For instance, an AS might originate prefixes P_1, P_2, and P_3. If vantage point V_1 has routes for P_1 and P_2 and vantage point V_2 has routes for P_2 and P_3, both V_1 and V_2 have a per-origin prefix visibility of 2/3 for this origin AS. A per-origin prefix visibility of 1 thus means that a vantage point has routes to all prefixes originated by a certain AS. Figure 3.4 shows the average *(arithmetic)* per-origin prefix visibility of all 960 vantage points. We determine the total set of prefixes an AS originates by looking through data from all vantage points. If at least one vantage point has a route for a prefix P originated by AS O, we add P to the total set of prefixes originated by AS O.

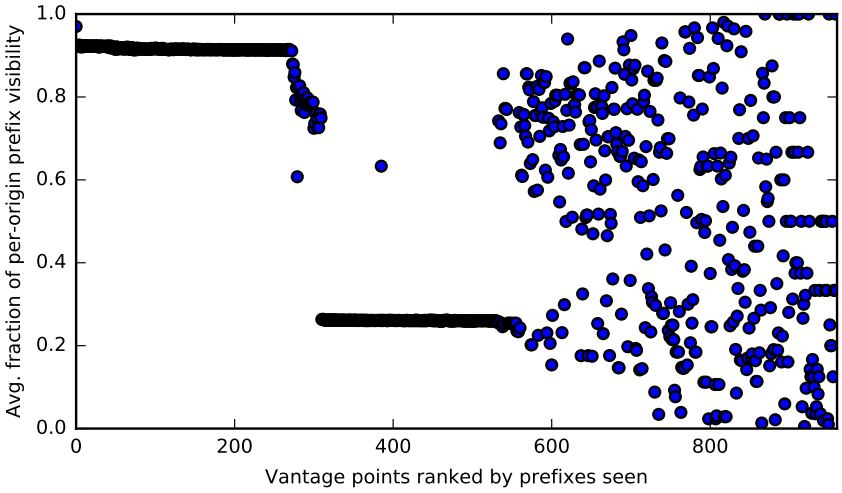


Figure 3.4: Average per-origin prefix visibility of vantage points.

We can see that there are almost no vantage points that have an average per-origin prefix visibility of 1. The only such vantage points are found in last 100 vantage points, ranked by the number of prefixes. We can again make out the 3 groups of vantage points that we have seen in previous figures. The first group of vantage points, those with a near 'global' view, tend to have a per-origin prefix visibility of 0.93 with one vantage point going as far up as 0.97. This means that even for vantage points that have high general visibility we do not get a complete set of routing information for observed origin AS. The second group of vantage points shows a significant drop in per-origin prefix visibility compared to the first group. While there are some vantage points between the two groups that have values of 0.6 or more, the majority of vantage points in the second group has a value of less than 0.3. This value shows just how pruned the view of the Internet we get through vantage points is: Not only do vantage points in the second group already see a very reduced number of origin AS, they also have routes to only a very reduced number of prefixes from these origins. This has implications if the methodology for measuring ROV-adoption relies on comparing routes to the same origin, as will be explored further in Chapter 4. The third group of vantage points show a wild distribution of per-origin prefix visibility, ranging from close to 0 to 1. This diversity is similar to what we have seen in previous figures. The third group of vantage points has such low visibility that analysis results are often not representative of the group. One reason for low per-origin prefix visibility could be the same reason why some vantage points a low number of routes to a collector: They are not exporting their entire routing table, but merely a subset. An alternative explanation is that a vantage point with low visibility might use default routes in its routing table. This could also explain low per-origin prefix visibility.

3.2 Limited Control

Another challenge is that of limited control. For a given prefix, data from route collectors only shows us the best selected route for each vantage point that peers with the collector.

To properly reason about the routing decision of a vantage point, there are two points of crucial information that we are missing:

Incoming Routes
The vantage point will select one route of all available routes for a given prefix. The available routes were received from other BGP routers, often located in other AS, as a result of their best route selection process. These routers will, in turn, have incoming routes for the prefix received from other AS.

Decision Process
The BGP best route selection process is standardized in RFC4271. This selection process involves attributes such as *Local Preference*, which can be set at will by network operators. Additionally, BGP implementations might use other attributes that are not specified in RFC4271, such as route age, in their implementation of the decision process. Both of these points mean that it can become very difficult to understand routing decisions by certain vantage points. A router might discard a route because of its RPKI validity state, or simply because the length of the prefix is an odd number. Understanding the decision process of a given vantage point becomes even harder when we do not know all incoming routes for a prefix, as is typically the case.

Since we don't know which routes are available to a vantage point for a given prefix, we can not know whether a route was not selected because there was a better route or because it was never propagated to the vantage point in the first place. **The routing policy of one BGP router might influence which routes are available to another router.** These routers need not be immediate neighbors in order to influence routers in other AS. For instance, if a route suddenly disappears we can not know which of the AS on the path is responsible. It could be that the origin AS has withdrawn its announcement of the prefix, it could also be that any other AS on the path has done so and there exists no alternate route.

3.3 Operational Concerns

Limited visibility and limited control are both challenges inherent to measuring BGP, and are a direct result of the information-hiding nature of BGP. As researchers, we might face additional challenges that are not related to fundamental design of BGP, but rather the implementation of it. For instance, major vendors of BGP routers use custom BGP best route selection algorithm. Cisco introduces an additional *weight* attribute that supersedes the local preference of a route, as well as *route age* as an additional tie breaking attribute [5]. Some observations might not have a feasibly explanation when only considering the contents of RFC4271, which standardizes BGP, but can be explained by a vendor-specific feature. Similarly, it is possible that differences between ROV as described in RFC6811 and ROV implementations by major vendors exist. In fact, operator gossip within the IETF suggest that popular BGP implementations that feature ROV do not implement it correctly. Specifically, that some devices do not validate routes distributed by other routing protocols than BGP. Additionally, our test have shown that some devices do not re-validate routes properly, specifically devices running custom editions of Cisco's Experimental IOS 15.3. This is problematic since the RPKI validity state of a route might change. This potentially exposes

the device to attacks. As researchers, we must be aware that ROV implementations in the wild might not align perfectly with the RFC, and measure accordingly.

CHAPTER 4

Uncontrolled Experiments

In this chapter we discuss and examine the current state of the art for measuring ROV adoption on the Internet. We start off by describing the methodology and replicating it with various data sets. We then question the methodology on i) how well it deals with the challenges we have presented in the previous chapter, and ii) how reliable the resulting classifications of AS are. Following this we then argue for a new methodology using a more controlled approach.

4.1 State of the Art

There exists currently little work on the measurement of ROV adoption on the Internet, as of now we know of only one paper [39] that deals with this subject. This paper presents a methodology that attempts to classify AS as either (*i*) not using ROV or (*ii*) using ROV to filter invalid routes.

4.1.1 Existing methodology

The existing work leverages RIB dumps from route collectors by analyzing the *AS Path* attribute of routes exported by the vantage points. Specifically, it uses the RPKI validity state of a route to classify AS on the AS path as either (*i*) *not ROV enforcing*, (*ii*) *ROV candidate*, or (*iii*) *ROV enforcing*. In this context, (*i*) means that an AS does not use ROV to filter invalid routes, (*ii*) means that there is some indicator that the AS is using ROV to filter invalid routes, and (*iii*) means that the AS is using ROV to filter invalid routes. Any AS that was classified as *ROV enforcing* must have been previously classified as a *ROV candidate*.

Not ROV Enforcing

The authors of [39] base their classification of AS as *not ROV enforcing* on a implicit assumption, which we write out explicitly here for easier reference:

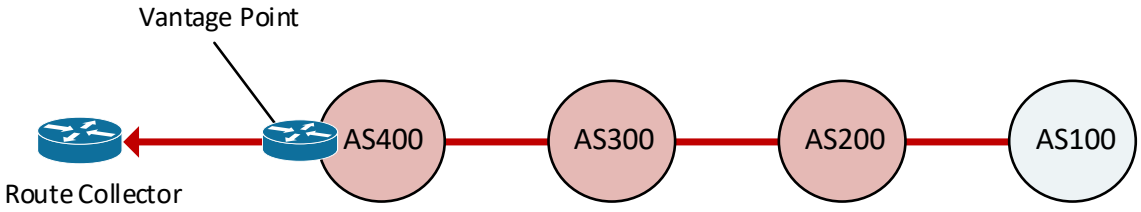


Figure 4.1: All AS except the origin AS are classified as *not ROV enforcing* (shaded red).

Absolute Filtering

An AS that has been observed to propagate an invalid BGP announcement is not using ROV to filter **any** invalid routes.

This assumption is the basis to classify AS that are observed on the AS path attribute of an invalid route as *non ROV enforcing*. An exception is made for the origin AS of the route. This is because an AS that is originating invalid routes might still conceivably filter incoming invalid routes since outgoing and incoming routes are subject to different routing policies. The authors make a second exception in case the route originates from an AS that is a customer of the AS that contains the vantage point which dumped the route. The authors reason that an AS might not filter invalid routes received from a customer AS, but might still do so for routes learned from peers or providers. This exception holds even if the vantage point receives the route not directly from the origin AS but is still a provider to it for other routes. The terms provider and customer AS here refer to the Gao-Rexford model for AS relationships, briefly explained in Chapter 2.

It is important to note that classification of AS as *not ROV enforcing* is the first step of the methodology and precludes those AS from being considered for later classification as *ROV candidate* and hence also *ROV enforcing*. The number of AS that are classified as *not ROV enforcing* establish a lower bound for the number of AS that do not enforce ROV, since it is likely that there are more AS that are propagating invalid announcements than the ones that can be observed on AS paths dumped by route collectors.

ROV Candidate

To classify an AS as a *ROV candidate* the authors of [39] leverage the existence of AS that originate one prefix with a non-invalid (*i.e.,* valid or unknown) announcement and another prefix with an invalid announcement. They look for a vantage point that exports route for both of these prefixes and compare the AS paths of the two routes. In case the two routes chosen by the vantage points have divergent AS paths, they check whether there exists **exactly one** AS on the path of the valid route which has not been classified as *not ROV enforcing* previously. In other words, the constraint for flagging an AS as a *ROV candidate* is that (*i*) it was not previously marked as *not ROV enforcing* (*ii*) it occurs on a non-invalid route originating from an AS which also originates at least one invalid route, and (*iii*) on the AS path of the route it was observed on, it is the only AS not marked as *not ROV enforcing* (with the exception of the origin AS). While these 3 conditions do not explicitly mention that there must be a divergence between the AS path of an invalid and a non-invalid route

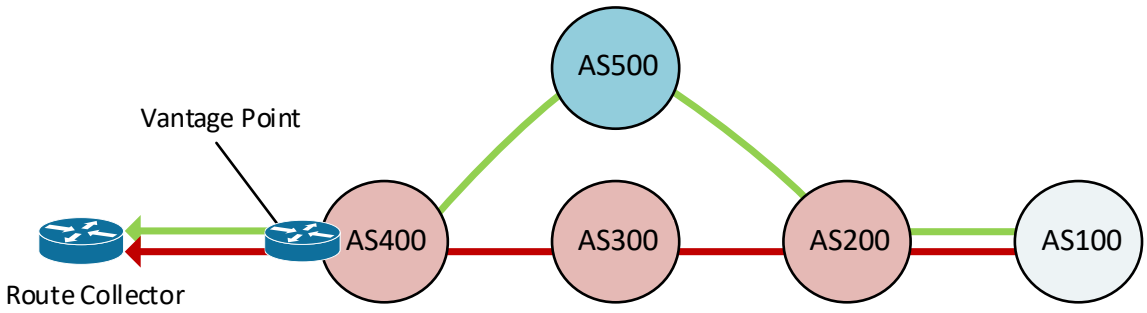


Figure 4.2: AS500 is marked as a ROV candidate (shaded blue). AS on the invalid route are marked as non ROV enforcing (shaded red).

with the same origin, this is implied by conditions (*i*) and (*ii*). The authors reasoning for condition (*iii*) is that it reduces the chance of a false positive by eliminating ambiguity.

An AS that is classified as a *ROV candidate* is also done so by associating it with the origin AS of the non-invalid route the ROV candidate was observed on.

ROV Enforcers

The authors of [39] use a simple condition to classify an AS as *ROV enforcing*: If an AS has been marked as a *ROV candidate* for three distinct origins, the authors classify it as *ROV enforcing*. It is important to note that the methodology used in [39] is not described in great detail. Specifically the reasoning behind classifying AS as ROV candidates and subsequently as ROV enforcers is not provided. A more precise explanation of the methodology could not be obtained even after multiple attempts.

Results

The authors use data from an unspecified time frame in July 2016. Using their methodology they have found that most of the top 100 AS, ranked by size of customer cone, do not enforce ROV. They found 9 AS among the top 100 that do enforce ROV, and 19 AS that could not be classified using this methodology.

4.1.2 Limits of the Existing Methodology

We believe that the methodology presented in [39] suffers from multiple major drawbacks that make it unsuitable to reliably classify AS as either *not ROV enforcing* or *ROV enforcing*.

In the case of classifying AS as *not ROV enforcing*, recall that the methodology classifies any AS found on the AS path of an invalid route as *not ROV enforcing* and uses the number of AS classified that way as a lower bound for the total number of AS that do not enforce ROV. This method rests of the implicit **Absolute Filtering** assumption. We think that this assumption is too strict and the fact that an AS was observed propagating one invalid announcement cannot be used to infer that it is not using some kind of ROV related filtering

policy. The authors of [39] seem to agree with this notion, as they make an exception for the origin AS since an AS might use ROV but not apply it to its own routes. They also make an exception for invalid routes that originate from a customer AS of the AS which contains the vantage point, which means they must think there are certain situation where an AS is using ROV only selectively. With these two exceptions, the authors of [39] acknowledge that enforcing ROV must not mean that an AS drops *all* invalid routes. We think that it is also feasible that more exceptions can exists such as an AS only filtering routes learned from certain peers, perhaps chosen arbitrarily. One example that could lead to this is when an AS has only partially deployed ROV. We argue that the lower bound that the methodology seeks to establish is not accurate and that is is possible that some AS that were classified as *not ROV enforcing* are false negatives.

In the case of classifying AS as *ROV candidates* and as *ROV enforcing*, the methodology relies on leveraging path divergences between invalid and non-invalid routes. It attributes a certain subset of these divergences to ROV based filtering of invalid routes and classifies an AS on the non-invalid path as a *ROV candidate*. We think that this method of classifying AS does not take into account the challenges that come with measuring BGP, which we have described in Chapter 3. Attributing a path divergences ignores the issue of **limited control**: The methodology offers no means to control any of the route attributes, which makes it very difficult to discern whether two routes have divergent paths because of ROV based filtering or because of other arbitrary reasons. A vantage point might choose two different routes for two prefixes belonging to the same origin AS because of business related reasons, misconfiguration, or traffic engineering. It is also possible that a vantage point is using a routing attribute to break a tie between for both routes, and by coincidence happens to chose different routes for two prefixes. Figure 4.3 illustrates a real world example, where a vantage point in AS25220 (80.81.194.140) has chosen two divergent routes, one being invalid and one being non-invalid, for two prefixes belonging to the same origin AS. The divergence in this case was due to the vantage point using *route age* as a tie breaker, and by chance the announcement for P_1 arrived earlier via AS3356 while the announcement for P_2 arrived earlier via AS1299. We have contacted the operators of AS3356 and confirmed that they are not using ROV on any of their routers. Note that in this case, the methodology used in [39] could easily have resulted in AS3356 being classified as a *ROV candidate*, even though there is no indicator that this AS is using such a policy. The authors of [39] seem to be aware of the problem of limited control, as they require an AS to be marked as a *ROV candidate* for three different origin AS before it is marked as *ROV enforcing*, likely in an attempt to reduce the likelihood of an AS choosing divergent routes for reasons other than ROV. We argue that requiring an AS to be marked as a *ROV candidate* for 3 different origins does not deal with the problem of limited control sufficiently. It is simply an arbitrary cut off point that offers no additional support for the claim that a *ROV candidate* is actually *ROV enforcing*. The bottom line even for an AS that meets this criteria is still: It might be using ROV to filter invalid routes, it might also not be.

The second challenge we have described in the previous chapter is that of **limited visibility**. We argue that the methodology presented in [39] does not take this problem into account sufficiently. While the authors note that the AS they classify as *not ROV enforcing* are merely a lower bound of the total number of AS that do not use ROV and thereby acknowledging the incompleteness of BGP data and the possibly of additional AS propagat-

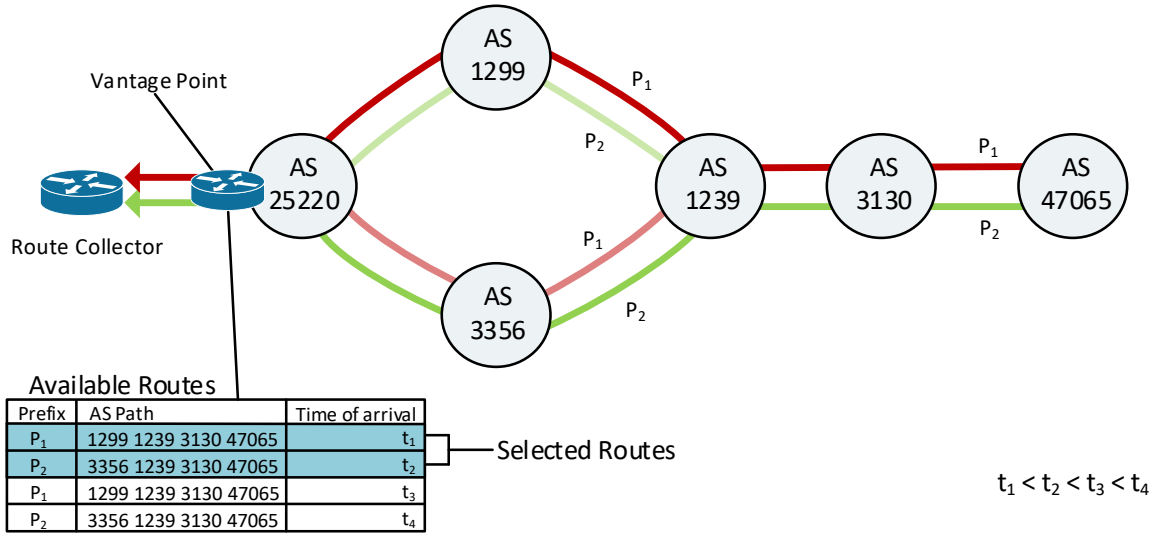
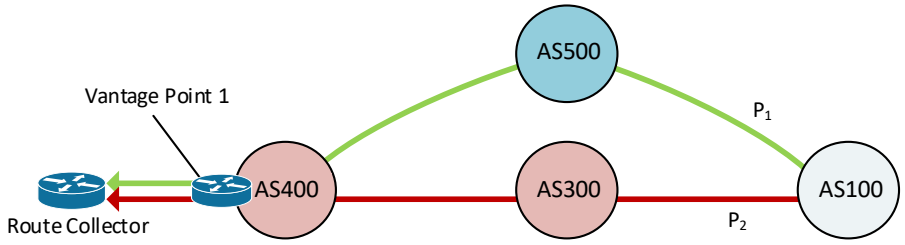


Figure 4.3: The vantage point receives two routes for both prefixes. For prefix P_1 the route via AS1299 arrives earlier than the one via AS3356, while the opposite occurs for prefix P_2. The vantage point uses route age to break the tie between the two available routes.

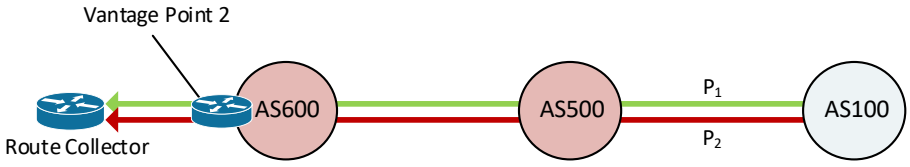
ing invalid routes that are not visible in the data set they have used, they ignore that the problem of incomplete data impacts their classification of AS as *ROV candidates*. An AS marked as a *ROV candidate* must not have previously been marked as *not ROV enforcing*. It is thus possible that when using a different data set, the same AS that has been previously marked as a *ROV candidate* would now be marked as *not ROV enforcing*, since the new data might include an invalid route that contains this particular AS on the AS path. This means that **the methodology has the potential to arrive at contradictory classifications** for the same AS when using different data sets. As there is not complete data set of all BGP routes that could be used for a 'definitive classification', it is not possible to solve this problem. In other words, the methodology might classify an AS as a *ROV enforcers* one week, and as *not ROV enforcing* the next week.

4.1.3 Replicating Existing Methodology

In order to gain a better understanding of the methodology and the impact of the problems we have described, we wanted to run this methodology on various data sets. Unfortunately, the authors of [39] have not provided their implementation of the methodology and even after multiple attempts we were not able to obtain the code. Thus we are unable to reproduce the results they have published and are forced to replicate the methodology with our own implementation. We will give an overview of the steps our implementation takes, an interested reader can find the full code at `https://github.com/RPKI/rov-measurement-code`. As input data, we take any BGP RIB dumps obtained using bgpreader, annotated by us with RPKI validity state information obtained at the same time as the route dumps occurred. To check for AS relationships, we use CAIDA's AS relationship data set [4] from the same month as the the route dump occurred.



(a) Routes chosen by VP1 for P_1 and P_2.



(b) Routes chosen by VP2 for P_1 and P_2.

Figure 4.4: Considering only data from VP1, AS500 might be a viable ROV candidate. If we add data from VP2 we see that is in fact *non ROV enforcing*.

We then perform the following steps:

1. **Mark non-ROV Enforcers**
   We iterate over all routes. For every invalid route we check whether the origin AS is a customer AS of the AS whose vantage point has dumped the route. If this is **not** the case, we mark every AS on the AS path except the origin AS as *not ROV enforcing*.
2. **Identify relevant routes**
   We filter the data to only include routes originating from AS that originate at least one invalid and one non-invalid route *with different AS paths*. We include **all** routes from origin AS that fulfill this condition. We do this on a per-vantage-point basis, *i.e.,* we have a separate data set now for each vantage point.
3. **Mark ROV Candidates**
   For each vantage point, for each origin AS, we iterate over all pairs of invalid/non-invalid routes. If the AS paths of the two routes diverge at any point, we check whether there exists exactly on AS on the non-invalid path that has not been marked as *not ROV enforcing*. If this is the case, we mark this AS as a *ROV candidate* and associate it with the origin AS of the routes.
4. **Mark ROV Enforcers**
   For each vantage point, we iterate over all AS marked as *ROV candidates*. Any ROV candidate that has 3 or more different origin AS associated with it, we mark as *ROV Enforcer*.
5. **Output Results**
   We collect all ROV enforcers found by each vantage point in a set. We then output the set of ROV enforcers.

The authors of [39] use data from 44 Routeviews vantage point, however they do not specify which vantage point exactly. The time when these routes were dumped is also not specified.

This information could not be obtained from the authors even after multiple attempts. This prevents us from using our implementation of the methodology to validate the original results, forcing us to use a different data set. Our complete data set to use with our implementation is from the 00:00 UTC route collector dump on October 25 2016. It contains routes from all Routeviews and RIPE RIS collectors, exported from 960 different vantage point. It is roughly 27GB in size. Immediately after downloading this data we annotate it with RPKI validity state information from the same time as the collector dumps occurred. Using the complete data set for analysis, our replication of the methodology classifies 922 AS as *not ROV enforcing* and 237 AS as a *ROV candidate*. Out of those 237, only the following 4 AS are associated as a *ROV candidate* for at least 3 origin AS and are thus classified as *ROV enforcing*:

$$AS8100\ AS25761\ AS17819\ AS262150$$

The authors of [39] have classified 9 AS out of the top 100 AS as *ROV enforcing*, but do not specify which AS. Our results differ significantly, with none of the 4 AS we have found being in the top 100 AS. This is to be expected, since we are using a different, more extensive, data set than the original work. To explore the impact of the data set on the resulting classifications, we re-run the analysis with subsets of our data set. For instance, if we restrict the data set to routes dumped by the `routeviews-equix` collector, which receives announcements from 34 vantage points, running the analysis results in zero ASes marked as *ROV enforcing*. In contrast, the `routeviews-wide` collector receives announcements from only 6 vantage points, out of which 2 have very low visibility (routes for less than 1000 prefixes), but the analysis results in the following AS classified as *ROV enforcing*:

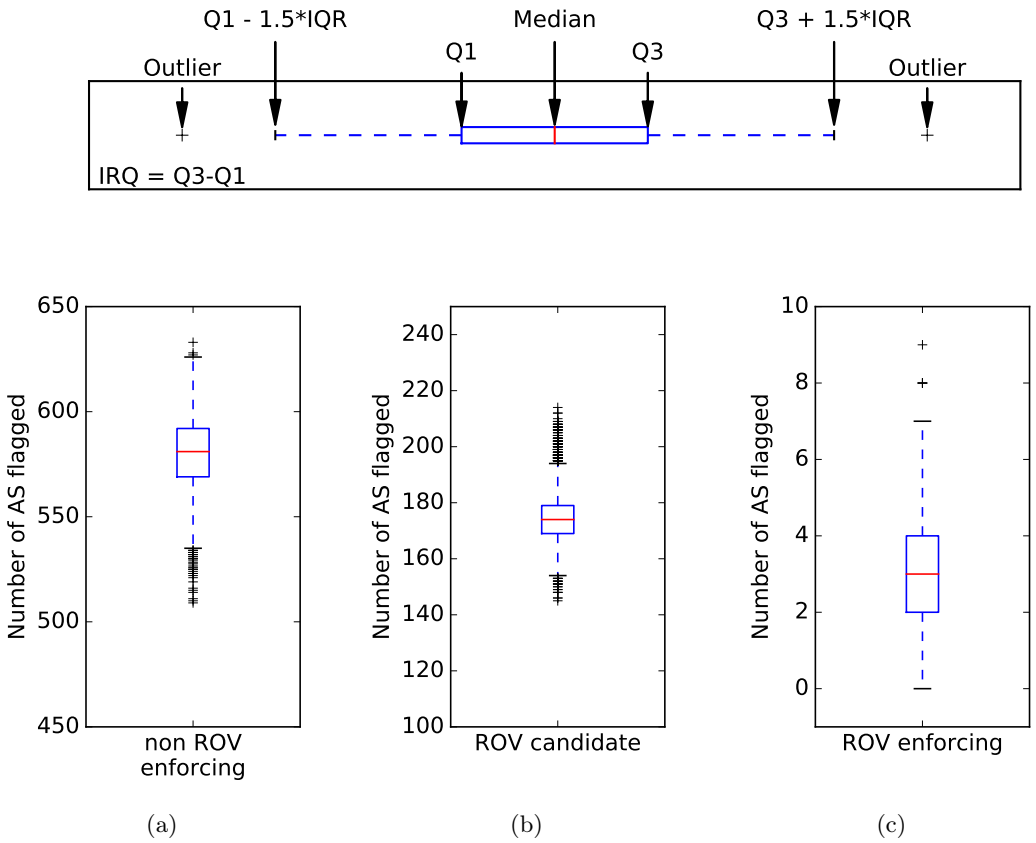
$$AS48237\ AS262150\ AS3786$$

These wildly different results underline the key flaw of the methodology described in subsection 4.1.2. The methodology does not take into account the limited visibility of BGP events and thus the incompleteness of the input data set, which can lead to false classification

Vantage Point Set Selection

We have established that the results of analysis are dependent on the input data. As we have shown above, analyzing data from different vantage points can lead to different AS being classified as *ROV enforcing*. We have shown in Chapter 3 that different vantage point can have dramatically different views of the Internet. We now explore the impact of the input data on the analysis results, specifically the selection of the set of vantage points whose exported routes are analyzed. Similar to [39], we select 44 vantage points and run our replication of the methodology on routes exported by them. We perform this analysis 5,000 times, each time selecting a unique set of 44 vantage points to draw data from.

Figure 4.5 summarizes the statistical properties (quartiles, extreme non-outliers, and outliers) of 5,000 samples of 44 vantage points, for the number of AS classified as *not ROV enforcing*, *ROV candidate*, and *ROV enforcing*. We can see that all three numbers can differ wildly depending on the vantage points whose exported routes were analyzed. Note that only for 1 set of vantage point did the method classify 9 AS as *ROV enforcing*.



(a) (b) (c)

Figure 4.5: Statistical impact of vantage points on the number of classified ASes (5,000 samples of 44 randomly selected vantage points).

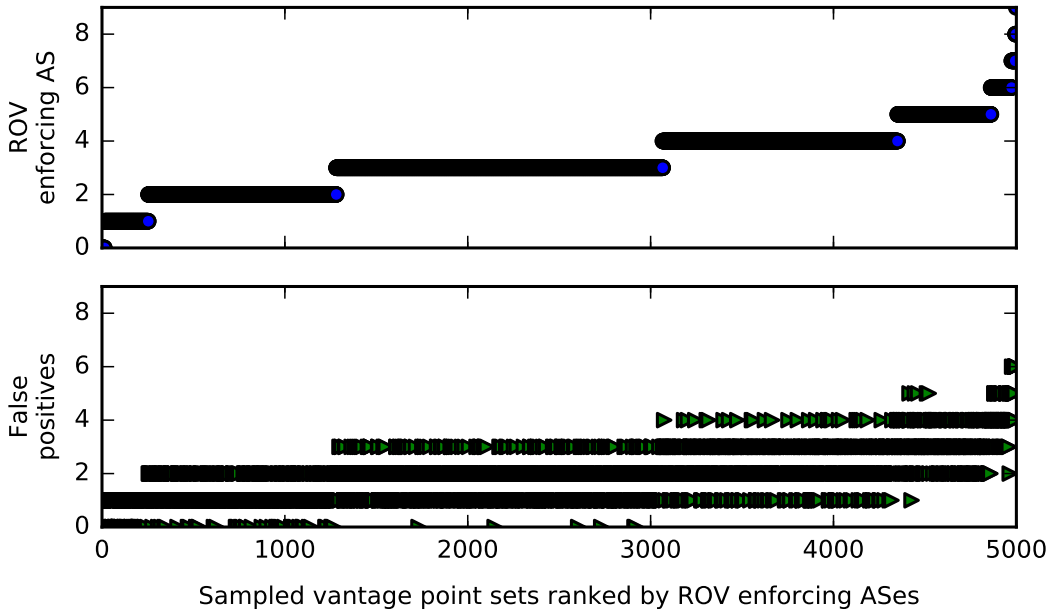


Figure 4.6: Number of AS classified as ROV enforcing and number of false positives.

In our critique of the methodology we have claimed that it has the potential to arrive at contradictory classifications for the same AS when using different data sets. An AS can be classified as *ROV enforcing* using one data set, and classified as *not ROV enforcing* when using a different data set which shows it to propagate invalid routes. We explore the impact of this using the result from the analysis of the 5,000 vantage points sets. First, we combine the data from all 960 vantage point and run our analysis with this. We store any AS classified as *not ROV enforcing* in a set. This 'global' set of AS classified as *not ROV enforcing* will be our ground truth. We now perform the analysis of the 5,000 vantage point sets again, but this time use the 'global' set of *not ROV enforcing* AS to check whether an AS can be classified as a *ROV candidate*. We call an AS that was previously classified as *ROV enforcing*, but now using the 'global' set is actually classified as *not ROV enforcing*, a **false positive**.

Figure 4.6 shows for each vantage point set the number of AS classified as *ROV enforcing* (top) and how many out of those are false positives (bottom). We can see that the majority of vantage point sets have at least one false positive. We can also see that for the vantage point set for which the highest number of AS (9) were classified as *ROV enforcing*, 6 of them are actually false positives.

Figure 4.7 shows the relative frequency of false positives each vantage points set. We can see that for only 121 vantage point sets there were zero false positives, while more than 4000 vantage point sets had a false positive rate of over 50%. For 654 vantage point sets, the false positive rate was 100%.

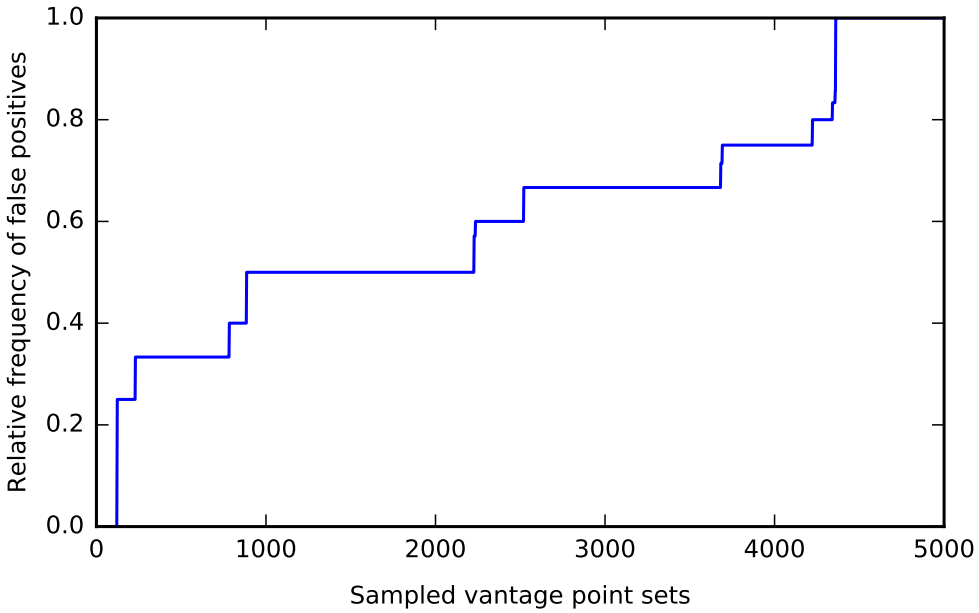


Figure 4.7: Relative frequency of false positives.

Conclusion

In Chapter 3 we presented *limited visibility* and *limited control* as the major challenges in measuring the usage of ROV. Our analysis of the existing methodology shows that it does not adequately deal with these challenges. Since the methodology only leverages already existing BGP announcements, it can not be used to reliably infer whether an AS is using ROV enforcement or not. This is an issue of limited control. After replicating the methodology we have shown that inferences based on routes provided by the selected vantage points can lead to incorrect identification of ROV non-enforcement and ROV enforcement policies. This is an issue of limited visibility. By leveraging a bigger data set, we have shown that the methodology results in misclassification of AS. This problem is inherent with the uncontrolled, passive approach of the methodology and is a natural consequence of limited visibility. The methodology does not reliably measure whether an AS is not ROV enforcing or ROV enforcing and thus we have no confidence in the results produced by it.

4.2 Uncontrolled Experiments: Analysis of Invalid Announcements

In Chapter 3 we have laid out reasons why determining whether a specific AS uses ROV is challenging. In fact, using only uncontrolled experiments it is impossible to determine while maintaining high accuracy. In the previous section we have then demonstrated the inadequate handling of the limited control and limited visibility that measuring BGP entails, and shown that this resulted in a high rate of misclassification. However, we think that it is still possible to analyze BGP data from uncontrolled experiments and glean interesting information related to ROV from it. This does *not* involve attempting to classify individual

AS as ROV enforcing or non ROV enforcing, but rather to analyze the characteristics of invalid routes to gain a better understanding of how they are routed throughout the Internet. Similarly to the existing work, we analyze the AS path attribute of invalid routes and non-invalid routes that originate from the same AS. This analysis is always done from the point of view of a fixed vantage point, since when comparing data gathered from different vantage points any inferred policy of AS on the AS path might just be the result of the differences between the vantage points.

4.2.1 Path Diversity

To gain a better understanding of how invalid announcements are routed, we compare them to non-invalid announcements of the same origin. Specifically, we would like to know whether these invalid announcements tend to propagate differently than the non-invalid ones. We limit ourselves to the origin AS for which our fixed vantage point has at least one non-invalid route and one invalid route. We exclude origin AS with only non-invalid announcements or only invalid announcements because in these cases we have no basis for comparison of AS path attributes of invalid and non-invalid announcements. In other words, to determine whether an invalid advertisement is being filtered by some AS, we need to at least compare it to a similar non-invalid advertisement, *i.e.,* that comes from the same origin AS and is observed by the same vantage point as the invalid one. For these origins, we measure the *path diversity* as seen from a vantage point. Path diversity of an origin as seen from a vantage point is defined as the number of *distinct* AS paths that lead from the vantage point to the origin. For instance, the following advertisements for prefixes P_1, P_2, P_3:

Prefix	Path
P_1	$V \to A \to B \to O$
P_2	$V \to A \to B \to O$
P_3	$V \to A \to C \to O$

show a path diversity of 2 for origin O with regards to vantage point V. For each origin O that fulfills the condition we measure the path diversity as seen from a fixed vantage point V for (*i*) all routes V has to O, regardless of the RPKI validity state of the announcements, and (*ii*) all non-invalid routes V has to O. If the vantage point V has the same routes for both invalid announcements and non-invalid announcements of origin AS O, the path diversity for (*i*) and (*ii*) will be the same, since invalid routes would add no new distinct AS paths to the set of AS paths from non-invalid routes. If the invalid announcements of origin AS O tend to take different AS paths to reach vantage point V than the non-invalid announcements, the path diversity of (*i*) will be higher than that of (*ii*). We calculate the path diversities (*i*) and (*ii*) for all origins that a vantage point V has at least one non-invalid and one invalid route to. This results in two path diversity distributions, showing for a fixed vantage point V how many origins have a specific path diversity. As a control group of routes, we also calculate (*iii*) the path diversities of origins when removing a random subset of routes, equal to the number of the invalid routes that were omitted in (*ii*).

Figure 4.8 shows the path diversities as seen by a vantage point located at the Frankfurt Internet Exchange DE-CIX. The x-axis shows the distinct number of AS paths leading from

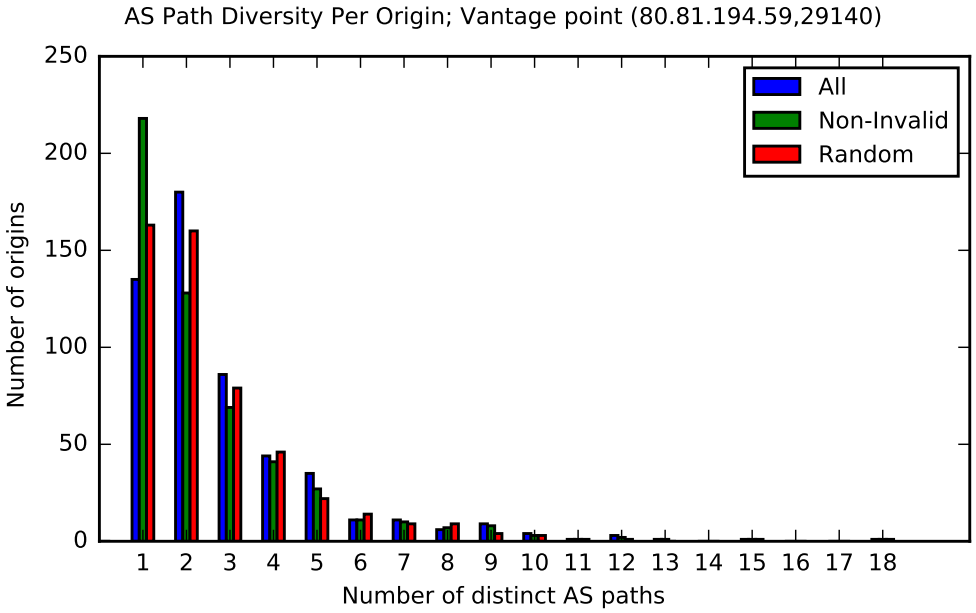


Figure 4.8: Path diversity as seen from vantage point at DE-CIX.

the vantage point to an origin, *i.e.,* the path diversity of the origin as seen from the vantage point. The y-axis marks the number of origin AS that have a specific path diversity as seen from the vantage point. We observe that for AS paths of (*ii*) non-invalid routes, for approximately 40% of origin AS there exists exactly one path to the vantage point. The remaining origin ASes have two or more paths leading to the vantage point, which can be explained by multi-homed AS. We can see that the path diversity distribution for (*i*) all routes is shifted to the right of that of (*ii*) only non-invalid routes. This means that the inclusion of invalid routes has increased path diversity, *i.e.,* invalid routes tend to be routed differently than non-invalid routes. Furthermore, we can see that the path diversity distribution for the (*iii*) control group routes, is very similar to the one for (*i*) all routes. This confirms that invalid routes tend be route differently, since if they weren't we would expect (*iii*) to be more similar to (*ii*).

Figure 4.8 shows the path diversity distributions for only one vantage point. Figure 4.9 shows the path diversity distributions of all 960 vantage points in our data set. The vantage points are on the y-axis sorted by the number of different origin AS they have routes to. The top plot shows us the path diversity distribution for all routes, the bottom one for non-invalid routes only. Notice that the first and second vantage point groups that we observed in the previous section are present again, although less pronounced. The third group of vantage point is not part of the plot, since none of these vantage point observed origin ASes with at least one non-invalid and one invalid announcement. The first group of vantage points (y=[0,275]) shows the same general patterns that we saw in Figure 4.8 regarding all routes and only non-invalid routes. For non-invalid routes (bottom plot), we can see the same peak at x=1 followed by a steady decline in number of origins. For all routes (top plot), we can see that it is also similar to Figure 4.8. Clearly, the other vantage points of the first group follow the same trend as at the Frankfurt Internet Exchange vantage point. The second group also

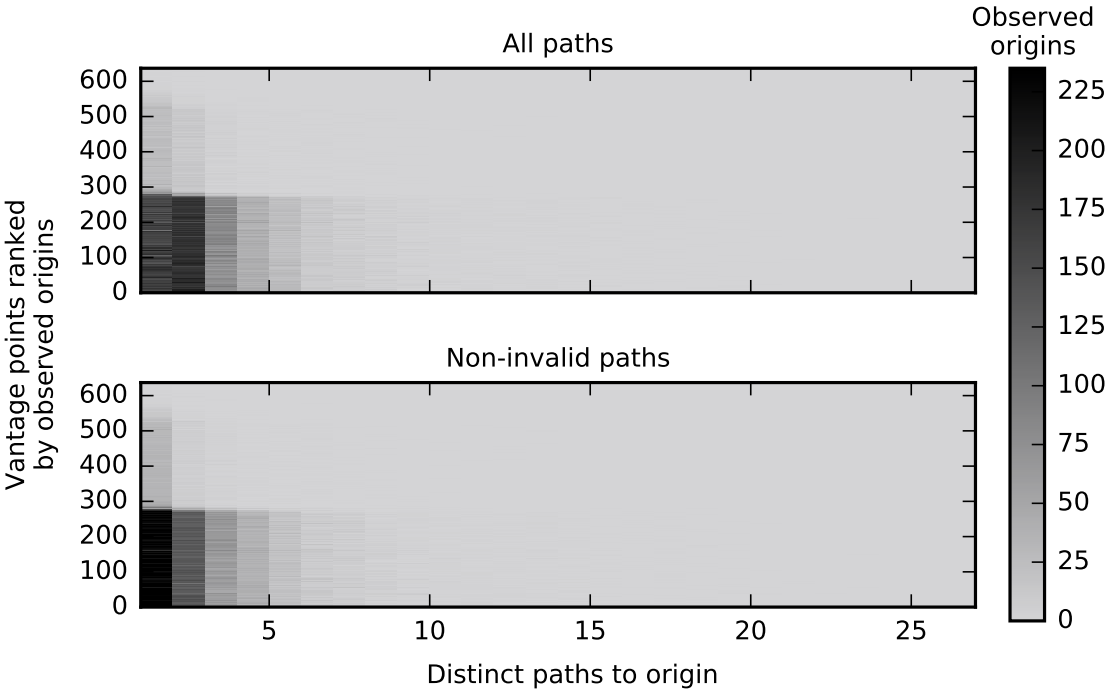
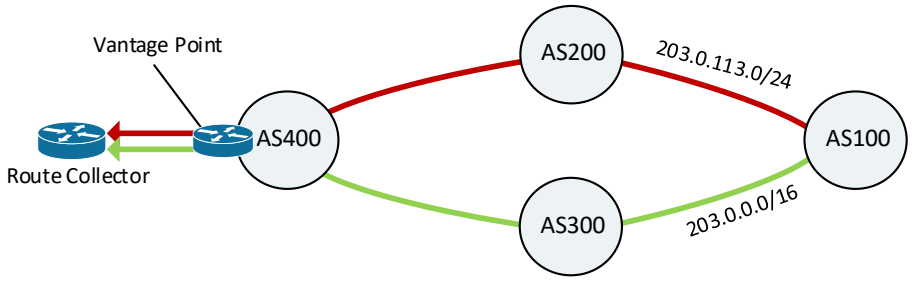


Figure 4.9: Frequency of distinct paths to origins with at least 1 non-invalid and 1 invalid prefixes as seen from vantage points.

follows the same trends, albeit much less pronounced since these vantage points overall see fewer origins that fulfill the condition. This confirms that what we have seen for the Frankfurt Exchange point is also true in general: **Invalid prefix announcements tend to be routed differently than non-invalid prefix announcements.** If there is a vantage point that exports two divergent routes, one invalid and one non-invalid, that are both originated by the same AS, we say that the prefix of the invalid route has is an **invalid prefix with routing differences**. These routing differences must not necessarily be due to AS using ROV, it is possible that invalid announcements are being treated differently for other reasons than their invalidity. One possible explanation is that of traffic engineering. A multi-homed AS with a main provider and a back-up provider might want to ensure that incoming traffic is received via the main provider, while the back-up provider is simply for redundancy. A common traffic engineering technique achieves this by announcing prefixes to the main provider while announcing a less specific prefix that covers the same space to the back-up provider. Because of longest prefix matching, traffic will be sent via the main provider. Figure 4.10(a) shows an example of this. It is now possible that the network operator wants to secure the prefixes with RPKI and creates a ROA for the less specific prefix, in the case of the example for 192.168.0.0/16. A common misconfiguration for ROAs is to set the maximum length field too short. If this happens in the case, and the maximum length is set to something less than 24, the announcements of the /24 prefixes to the main provider will be invalid while the announcement of the /16 prefix will be valid. As researcher, we might then observe a vantage point exporting divergent routes and be tempted to attribute this to ROV, when in reality the divergence has nothing to do with ROV.



(a) In order to steer incoming traffic via AS200, AS100 announces more a specific prefix. AS300 serves as a back-up provider.

ROA	
Prefix:	203.0.0.0/16
Max. Length:	16
ASN:	100

Figure 4.10: The maximum length field of the ROA is misconfigured to 16, making the announcement of 203.0.113.0/24 invalid.

In situations as we have just described, three conditions are met: (*i*) The invalid prefix exceeds the maximum length specified in the ROA, (*ii*) the AS path of the route for invalid prefix diverges at the first hop after the origin from the AS path of the non-invalid prefix, and (*iii*) the non-invalid prefix covers the invalid prefix. Figure 4.11 shows the reasons for invalidity for all invalid prefixes that were announced by an origin AS which also announces non-invalid prefixes. The x-axis list the vantage points of our data set, the top plot shows the number of invalid prefixes each vantage point has exported routes for, and the bottom plot breaks those prefixes down by reason of invalidity. We can see that 30% of these invalid prefixes are invalid only because of an origin AS mismatch, 60% of them are invalid because they exceed the maximum length specified in the ROA, and the remaining 10% are invalid for both of these reasons. This shows that a significant part of the invalid prefixes that show routing differences are invalid because of their length, making them possible candidates for the traffic engineering scenario we have described above. However, there is still a large chunk of prefixes that are invalid because of an AS mismatch, which is not explained by the traffic engineering theory.

The check for our second condition, we find the divergence point between the AS path of an invalid route and the AS paths of non-invalid routes from the same origin AS. We define the divergence point between two paths to be the position on the path where the paths first contain different AS, starting at origin O (index 0). For example, the two paths:

$$O \to A \to B \to V$$
$$O \to A \to C \to V$$

diverge on point 2, with AS B and C being different. Let O be an origin that advertises at least one invalid prefix with routing differences seen from vantage point V. We determine the divergence point distribution between (*i*) each pair of distinct non-invalid paths leading from O to V, and (*ii*) each pair of distinct paths leading from O to V. This gives us two

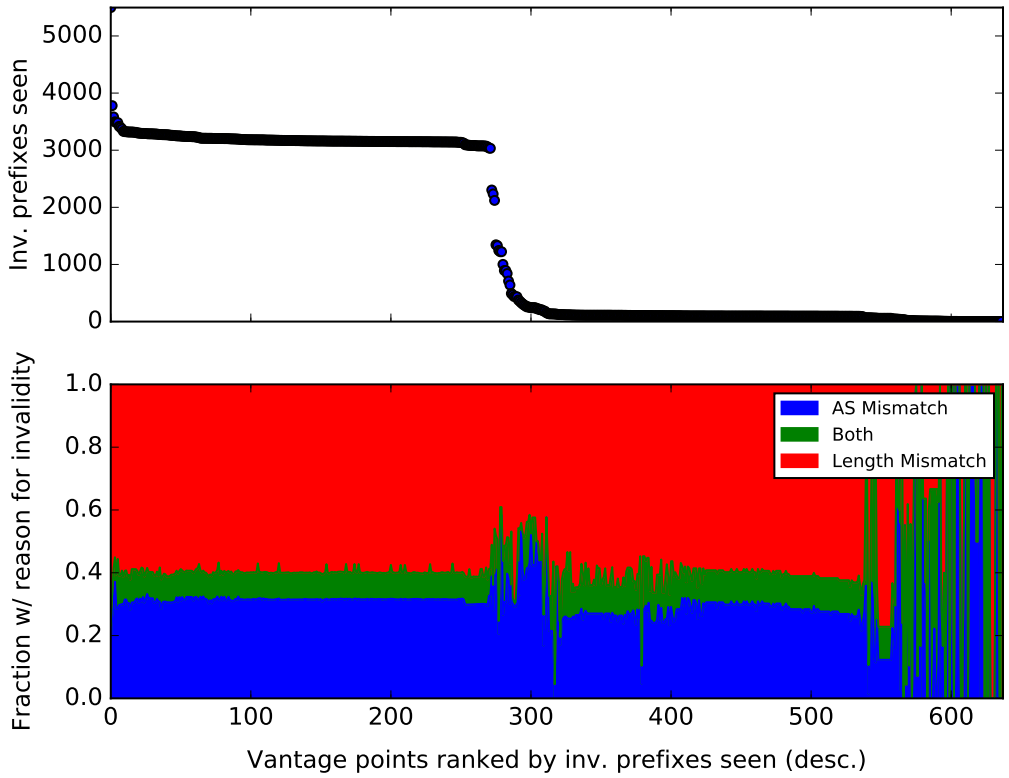


Figure 4.11: Reasons for invalidity for prefixes whose origin AS is also announcing a non-invalid prefix.

divergence point distributions, one for non-invalid paths and one for all paths. An example with origin O, vantage point V, non-invalid prefix announcements for p_1, p_2, p_3, p_4, invalid prefix announcements for p_5, p_6 and observed announcements:

Prefix	Path
p_1	$O \to D \to E \to F \to V$
p_2	$O \to A \to G \to C \to V$
p_3	$O \to A \to G \to C \to V$
p_4	$O \to A \to B \to C \to V$
p_5	$O \to A \to B \to C \to V$
p_6	$O \to A \to H \to I \to V$

gives us 3 pairs of distinct non-invalid paths with the divergence point distribution:

Div. Point	Frequency
1	2
2	1

and 6 pairs of distinct paths with the divergence point distribution:

Div. Point	Frequency
1	3
2	3

In this example p_6 exhibits routing differences, because its path is distinct from any of the non-invalid prefix announcements. We calculate the two path divergence distributions for each origin on a per-vantage-point basis, and then normalize over the amount of total path pairs compared. Figure 4.12 shows these distributions for a vantage point at DE-CIX. We can see that for both non-invalid path pairs and all path pairs, over 80% of them diverge at the first hop of the AS path. This strengthens the argument that our traffic engineering theory can explain a large number of observed routing differences, rather than ROV. The differences between the two distributions are insignificant. An interesting feature is the tail of the distributions, showing that some paths diverge as late as the fifth hop. The peak of the divergence points at the first hop can be seen across all vantage points.

The third condition for the traffic engineering scenario we have described is that there exists a non-invalid prefix announced by the same origin which covers the invalid one. Figure 4.13 shows the fraction of such invalid prefixes with routing differences on a per-vantage-point basis. We can see that especially for the vantage point with high visibility (x=[0,275]) over 70% of invalids are covered. This is a another indicator that a large number of divergences could be due to traffic engineering rather than ROV based policies.

Figure 4.14 shows the divergence point distribution between invalid prefixes with routing differences and their covering non-invalid prefix. The y-axis shows the vantage points, excluding the ones that do not see any covered invalid prefixes with routing differences, while the x-axis shows the divergence point. This also includes *Same Path*, meaning that there is no divergence in the AS paths of the invalid prefix and the covering non-invalid.

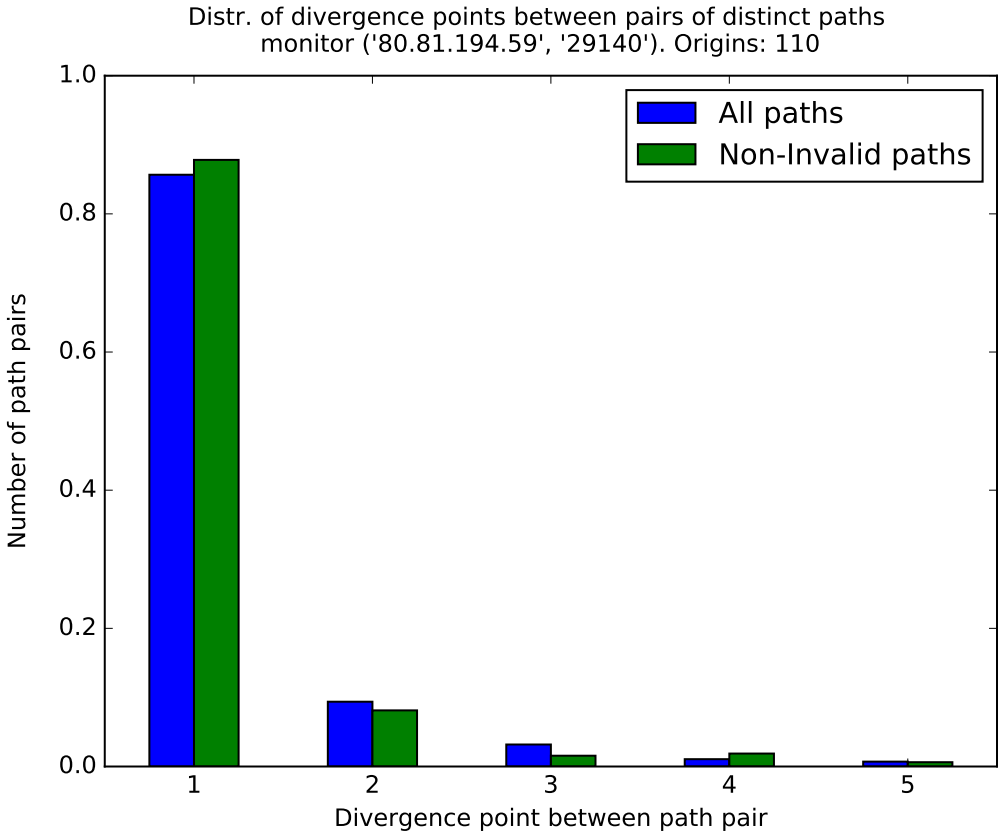


Figure 4.12: Path divergence distributions for all origins observed by a vantage point, for all paths and only for non-invalid paths.

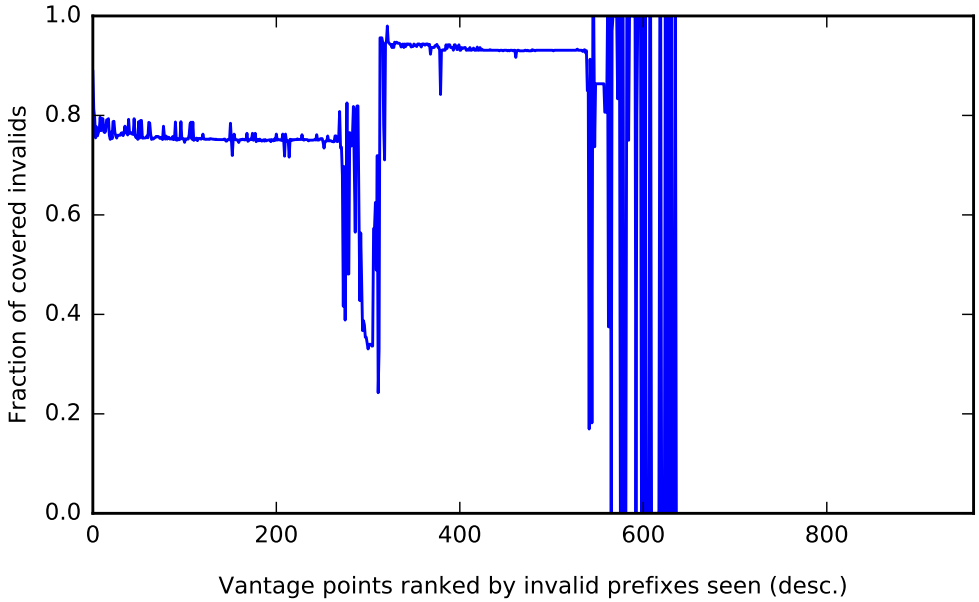


Figure 4.13: The fraction of invalid prefixes with routing differences that are covered by a non-invalid prefix of the same origin.

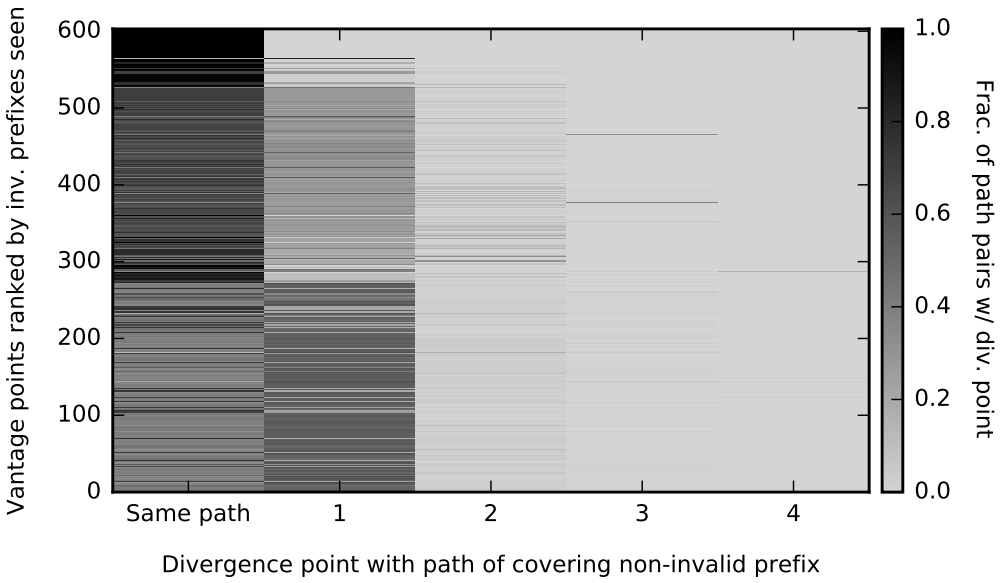


Figure 4.14: Divergence point distribution between invalid prefixes and their covering non-invalid prefix.

The color intensity shows the fraction of path pairs observed by a vantage point that have a certain divergence point. We see that 'Same Path' dominates for almost all vantage point. Excluding that, the peak divergences are at the first hop for nearly all vantage points. This further underlines the traffic engineering theory.

4.3 Conclusion

In this chapter we have analyzed the state of the art of measuring ROV adoption on the Internet. After analyzing the flaws in the methodology we have replicated it and used our own implementation to demonstrate that the methodology produces inaccurate, unreliable results. We conclude that uncontrolled, passive experiments are fundamentally not suited to determine whether a specific AS is using ROV or not. Analysis of invalid prefix announcements has shown that there are indeed routing differences that are worth exploring, although a large number of them are likely explained by traffic engineering. To accurately measure ROV adoption, a new methodology is needed which does not rely on data from uncontrolled experiments. It must adequately deal with the challenges we have laid out in Chapter 3. In the next Chapter we present a methodology based on controlled, active experiments and show that it can be used to accurately determine whether an AS is using ROV.

CHAPTER 5

Controlled Experiments

This chapter presents a methodology to measure ROV adoption using controlled experiments, improving upon the current state of the art. First, the experimental facilities that we use to conduct experiments are outlined. We then describe the purpose, setup, and results of multiple experiments. Finally, it gives an outlook on future work such as more fine-grained experiments with which more complex policies can be found.

5.1 Experimental Facilities

For our experiments, we use the facilities of the PEERING [65] testbed to send out BGP updates. The testbed provides access for researchers and educators to the global BGP routing system, enabling them to conduct more informative control-plane research. The PEERING testbed is well suited for the task of finding AS that use ROV-based routing policies because it provides connectivity to a large IXPs such as the Amsterdam Internet Exchange (AMS-IX) as well as direct peering connections to various well-connected AS such as AS3130.

5.1.1 Connectivity as Documented

This section describes the ways the PEERING testbed is connected to other AS. It uses information obtained from the PEERING web-sites and the various locations that PEERING has deployed routers. It is possible that the information on these web-sites is outdated and does not accurately reflect the actual state of connectivity. We call an AS that *should* be directly connected to PEERING according to information on the PEERING web-site or on the website of an Internet Exchange point at which PEERING peers *reachable according to documentation*. Furthermore, simply because an AS is directly reachable according to documentation it is not guaranteed that it will have a routing policy in place that is favorable to PEERING, *i.e.,* actually uses the routes received from PEERING in its best path selection process. In practice, even if such a policy is in place it might not benefit researchers if the AS contains no vantage points. Keeping this in mind, we argue it is still beneficial to give a *best case* overview for the connectivity of the PEERING testbed. The best case being

that every AS that is reachable according to information from the PEERING and various Exchange Point websites is actually reachable in practice *and* has a favorable routing policy in place.

The PEERING testbeds operates BGP routers at various sites in North and South America as well as Europe. These BGP routers are referred to as *BGP muxes*. Each mux connects with one or more peers, each peer is identified by a unique *session ID*. Users of the PEERING testbed can send BGP updates to either all peers of a mux, or specify which peers should get an update message using session IDs. The PEERING testbed allows users to send BGP updates for both IPv4 and IPv6 prefixes. Table 5.1 gives an overview for each mux, showing the number of peers and the number of different AS these peers belong to.

Mux		**IPv4 Peers**		**IPv6 Peers**	
Name	**Network Location**	**Router**	**AS**	**Router**	**AS**
amsterdam01	Amsterdam IX	67	54	54	44
clemson01	Clemson University	1	1	0	0
cornell01	Cornell University	1	1	0	0
gatech01	Georgia Tech	1	1	0	0
grnet01	Greek Research Network, GR	1	1	0	0
isi01	Los Nettos Regional Network	1	1	0	0
neu01	Northeastern University	1	1	0	0
phoenix01	Phoenix IX	1	1	1	1
sbu01	Stony Brook Univ.	1	1	0	0
seattle01	Seattle IX	12	10	10	8
ufmg01	Federal Univ. of Minas Gerais	1	1	0	0
uw01	Univ. of Washington	1	1	0	0
wisc01	Univ. of Wisconsin-Madison	1	1	0	0

Table 5.1: Direct peering connectivity of PEERING BGP muxes.

In addition to these peers, the amsterdam01, phoenix01, seattle01, and ufmg01 mux are also peering at route servers. The amsterdam01 mux peers with the legacy route servers at Amsterdam IX (AMS-IX [3]), the seattle01 mux with the route servers at Seattle IX (SIX [18]), the phoenix01 mux with the route servers at Phoenix IX [14], and the ufmg01 mux with the MLPA route server at IX.br at the location in Belo Horizonte [10]. Each of these exchange points documents a list of AS peering at their route server on their website. These route servers enhance the connectivity of the PEERING testbed massively, allowing users to send BGP updates to a larger number of AS. Table 5.2 shows for each mux that is peering with a route server the number of AS peering with that route server and thus can be reached with update messages.

In addition to the route servers already mentioned, the Amsterdam IX operates an additional pair of route servers with enhanced functionality. These are called *Falcon Class* route servers and offer BGP community-based route origin validation support [2]. By tagging received BGP announcements with standard BGP communities, the falcon route servers can convey information about RPKI validity status of the announcement to the route server peers. Additionally, the falcon route server offers various filtering policies for the route server

Mux	AS reachable through route server
amsterdam01	664
seattle01	170
ufmg01	37
phoenix01	33

Table 5.2: Route server connectivity of PEERING BGP muxes.

peers. This allows a route server peer to, for instance, only receive BGP announcements whose RPKI validity status is valid or unknown (*i.e.,* filter invalids). Additionally, the falcon route servers can be used by route server peers to shape their policy for outgoing announcements. As an example, a route server peer can instruct the falcon route server to exclude certain peers when propagating one of their announcements. While an AS using the falcon route server to filter routes based on RPKI validity status does not technically perform route origin validation, we feel this is a very relevant feature when it comes to deployment of route origin validation since it achieves the same outcome. There are 93 AS peering with the falcon route servers at AMS-IX, excluding the PEERING AS. This brings the number of AS reachable via a route server via the amsterdam01 mux up to 675 AS. Note that the information regarding the peers of the falcon route servers was not obtained from a website but lifted directly from a looking glass of a falcon peer. Table 5.3 shows the total number of AS reachable directly, via a route server, and both options combined.

Clearly the number of AS reachable through route servers is much larger than the number of AS reachable through direct peering. In order to reduce overhead and simplify the implementation of experiments, we chose to ignore the BGP muxes with very low connectivity. These are clemson01, gatech01, grnet01, isi01, neu01, sbu01, uw01, and wisc01. This leaves us with amsterdam01, seattle01, phoenix01, and ufmg01. Between those four muxes, amsterdam01 dominates in connectivity with a total of 680 reachable AS, with the next best mux being seattle01 being able to reach 172 AS. In comparison, phoenix01 and ufmg01 only connect to 35 and 37 AS respectively.

Mux	AS reachable (direct)	AS reachable (route server)	AS reachable (both)
amsterdam01	54	675	680
seattle01	10	170	172
ufmg01	1	37	37
phoenix01	1	33	34
Sum			871

Table 5.3: Cumulative sum of AS reachable via the four top PEERING muxes.

The grand total of all AS reachable directly via the amsterdam01, seattle01, phoenix01, and ufmg01 muxes is 871.

5.1.2 Connectivity and Visibility

The previous subsection listed the number of AS that can be reached with BGP updates directly using the PEERING testbed infrastructure according to documentation. As already mentioned, documented reachability must not always translate to practical reachability as information obtained from websites might be incorrect or unfavorable routing policies might be in place. However, even in the absence of these two factors a BGP update sent to an AS that is not providing a vantage point, and thus offers no information on their selected paths, is of limited use to us. Note that such an AS might still provide transit for received prefixes, propagating it to its peers which might provide a vantage point. This can still be of use when trying to determine the routing policy of AS the update was originally sent to. In this subsection we quantify how many of the AS reachable according to documentation fulfill the following conditions:

Peer Condition
: The AS must be an active BGP peer of the PEERING testbed. This means this AS must either have a direct peering session established with the PEERING testbed, or must be peering at a route server where a BGP mux of the PEERING testbed is also peering.

Acceptance condition
: The AS must consider routes received from the PEERING testbed for BGP best path selection.

Visibility Condition
: The AS in question must provide a vantage point which exports routes received from the PEERING testbed to a route collector.

An AS that fulfills all three conditions can be targeted easily with experiments. In later sections, we will discuss how much these conditions can be relaxed, especially the Peer condition, to broaden the coverage of experiments while still retaining accuracy in results.

Peer Condition

Due to technical limitations of the client software PEERING provides, checking which AS is actually peering with a mux is not directly possible. However, it is possible to obtain a list of routes that a mux has received from its peers. We classify an AS as an active peer if it has sent at least one route to a PEERING mux. This excludes AS that are peering with a mux, but do not send any BGP announcements to it. Table 5.4 shows the number of active peers classified this way, as well as the total number of active peers of all muxes. The table also shows the active peers as a percentage of the AS that are reachable according to documentation via a mux. Note that we are now only considering the amsterdam01, seattle01, phoenix01, and ufmg01 muxes since the remaining muxes are quite limited in connectivity. The amsterdam01, ufmg01, and phoenix01 muxes all show a significant decline in number of peer AS compared to the number of AS reachable according to documentation. Surprisingly, the seattle01 mux actually has *more* active peers than peers reachable according to documentation.

Mux	Active peers [AS]	[%] of reachable AS (doc.)
amsterdam01	589	86.6%
seattle01	179	104%
ufmg01	25	67.6%
phoenix01	24	68.6%
Sum	769	88%

Table 5.4: Number of active peers (AS) of 4 top PEERING muxes and the percentage of AS reachable according to documentation that are active peers.

Acceptance Condition

Determining whether an AS has considered the routes we have sent via PEERING for best path selection is not possible in all cases. For example, an AS that accepts the route but chooses not to propagate the announcement to any peers. An AS might propagate the announcement to its peers but is not peering with a route collector. In this case there is still the possibility that the announcement eventually propagates to an AS that *does* peer with a route collector and we will be able to determine that the original AS is indeed accepting our routes. To determine which AS fulfill the acceptance condition, we analyze the AS paths vantage points have exported to route collector for PEERING prefixes. An example path might be:

$$AS400 \leftarrow AS300 \leftarrow AS200 \leftarrow AS100 \leftarrow \text{PEERING}$$

PEERING is the origin AS. AS400 contains a vantage point that exported this path to a route collector. The fact that this route was propagated to AS400 tells us that AS300, AS200, and AS100 must have considered the announcement sent by PEERING in their best path selection. Hence, they all fulfill the acceptance condition. To determine a lower bound of the number of AS that accept our routes, we announce four /24 prefixes, one to each of the top 4 muxes. We then count the number of distinct AS found on AS paths dumped by any RIPE RIS or Routeviews route collector for each prefix, over a period of 16 hours. Note that a higher number of AS found in this way *does not mean that the mux is especially well connected*. Instead means that there is a higher number of distinct AS between the mux and the various vantage points. We do this do get an idea of how well our announcements propagate, not to evaluate mux connectivity.

Table 5.5 shows the number of AS found to be propagating PEERING announcements originating from the 4 muxes. It also shows the total number of AS that have been found to propagate any PEERING announcement, regardless of origin mux. We can see that there is a big overlap between the accepting AS per mux. This is unsurprising since in all cases we are using the same route collectors as sources, which have a limited number of vantage points exporting routes to them. The vantage points in turn have a limited number of upstream AS they receive routes from. We intersect the set of AS that fulfill the Acceptance condition with those which fulfill the Peer condition to find out how many active peers of PEERING actually propagate the announcement to other AS. Table 5.6 shows how many AS fulfill both

Mux	Accepting AS
amsterdam01	274
seattle01	271
ufmg01	278
phoenix01	45
Sum	311

Table 5.5: Number of AS accepting PEERING prefixes for best path selection and propagating them to other AS.

the Peer and the Acceptance condition. In total, we could only determine 89 AS that fulfill

Mux	Accepting peers [AS]	[%] of reachable AS	[%] of AS reachable (doc.)
amsterdam01	69	11.7%	10.1%
seattle01	18	10.5%	10%
ufmg01	4	16%	10.8%
phoenix01	3	12.5%	8.6%
All	89	11.5%	10.2%

Table 5.6: Number of AS that fulfill the Peer *and* the Acceptance conditions, the percentage of active peers AS, and the percentage of AS reachable according to documentation.

both conditions. However, it is likely that there are more peers of PEERING that propagate routes, but which do not show up on any AS path exported by a vantage point. This means that 89 is a lower bound for the number of AS fulfilling both conditions.

Visibility Condition

Determining whether an AS fulfills the Visibility condition is straightforward. We can simply check which AS is exporting routes to route collector by analyzing the dumps of the collector. Any such an AS must by definition also fulfill the Acceptance condition, since the act of exporting a route to a collector cannot happen when the AS hasn't considered the route for best path selection in the first place. Table 5.7 shows the number of AS that fulfill the Visibility condition for each mux, *i.e.,* that export routes to prefixes announced via that mux.

We can now combine the numbers for the Peer, Acceptance, and the Visibility conditions and get the total number of AS that we can measure reliably. Table 5.8 shows this number per mux, as well as the total.

Mux	Visibility AS
amsterdam01	225
seattle01	222
ufmg01	232
phoenix01	33
All 4	235

Table 5.7: Number of AS that fulfill the Visibility condition.

.

Mux	All 3 conditions (AS)
amsterdam01	60
seattle01	17
ufmg01	4
phoenix01	2
All 4	74

Table 5.8: Number of AS that fulfill the Peer, Acceptance, and Visibility conditions.

.

5.1.3 Internet Resources

AS Numbers

The PEERING testbed uses AS number 47065 to establish all peering sessions In addition to that, it also provides its user with 7 additional AS numbers: 33207, 61574, 61575, 61576, 263842, 263843, and 263844 [12].

IP Prefixes

The PEERING testbed owns 3 IPv4 prefixes (184.164.224.0/19, 138.185.228.0/22, 204.9.168.0/22) and 1 IPv6 prefix (2804:269c::/32). These resources are shared between users of the PEERING testbed. Additionally PEERING allows its users to announce prefixes from other organizations as long as permission has been granted. For our initial experiments we used two /23 IPv4 prefixes (151.216.32.0/23, 151.216.34.0/23) that RIPE NCC graciously made available to us for a limited time. Further experiments, and more importantly for the implementation of a longitudinal study, we have been granted access to a /19 IPv4 prefix (147.28.224.0/19) by RG.net

RPKI Certificate Authority

In order to issue and revoke ROAs one needs to own a RPKI Resource Certificate, which requires the operation of a RPKI Certificate Authority. RIPE NCC offers RPKI Resource Certification functionality which allows users to issue and revoke ROAs for their resources, provided those resources have been allocated using the ripe-ncc-ta.cer root certificate owned and operated by RIPE NCC. This functionality provides an abstraction layer to the operator of a Certificate Authority. This was sufficient for initial experiment that only involved static

ROAs. For experiments requiring dynamic ROA changes, we are using resources provided by RGnet. This required us to set up a child Certificate Authority, configuring RGnet as the parent Certificate Authority. RGnet then delegated the ownership of 147.28.224.0/19 to us, which allowed us to issue and revoke ROAs for this prefix. The ROAs along with the corresponding Resource Certificate, Manifest, and Certificate Revocation List, are published on RGnets infrastructure and can be downloaded by any relying party.

5.2 Experiments

The previous section has given an overview over the experimental facilities of the PEERING testbed, which will be using for all experiments. We have also established, as best as possible given the limited information, the coverage and reach of BGP announcements sent via the PEERING testbed. In this section we now present the various experiments that we have conducted in the course of this work, by first stating the goal of the experiment, second the setup and reasoning of the experiment, and thirdly discussing the results and implications of the experiment.

5.2.1 The Basic Approach

Experiment Goal

BGP as a protocol is quite flexible, as the nature of AS-level peering is shaped by inter-organizational relationships rather than purely technical requirements. Hence BGP allows for complicated, intricate routing policies which can be challenging to determine as an outsider. In our first experiment, the basic approach, we focus on what we think is the simplest routing policy related to ROV: The decision whether to consider an invalid route for best path selection, or to discard it based on its invalidity.

The goal is the find out which AS, if any, have already deployed such a policy. As discussed in Chapter 3, an AS is not 'atomic' with one clearly defined routing policy. This makes it necessary to be more precise about the goal of this experiment. Hence we state it differently:

Experiment Goal
Testing whether there exist AS that (*i*) have deployed ROV on at least one BGP router, and on that router are (*ii*) using the validation results to exclude invalid routes from BGPs best path selection process.

There are two conditions that we require a BGP router to fulfill in order for us to determine if a filter policy for invalid routes exists:

Reachability Condition
The BGP router we are testing needs to be able to receive BGP Updates from our experimental BGP router *directly*. This is achieved either via a direct peering session between the two routers or with both routers peering with the same route server.

Visibility Condition
The BGP router we are testing needs to be a vantage point. This is a necessary condition in order to observe the router's path selections, which is required to infer policy.

If the reachability condition does not hold, that means the BGP router is receiving our updates not directly from us but rather over at least one intermediate router. This introduces ambiguity, since in this scenario there are now two routers that could be using ROV to drop a route. If the BGP router we are testing does not export our invalid routes to a route collector, this could because it is using ROV to filter the route, or because the intermediate router is using ROV to filter the route. It is also possible that the intermediate router simply does not propagate our updates to the router we wish to test. Unless the intermediate router is also a vantage point, we cannot resolve this ambiguity. The two conditions stated above are quite limiting, since there exist few AS in the first place that are directly reachable via PEERING and also contain a vantage point, adding the requirement that it needs to be the same BGP router limits the coverage of the experiments severely. An AS might have different routing policies on different routers. However, these routers will still disseminate routes learned from other AS to other BGP routers within the AS. We can use this fact to relax the reachability and visibility conditions to deal with AS instead of BGP routers and thus increase the coverage of the experiment:

Reachability Condition (AS)

The AS we are testing needs to be able to receive BGP Updates from our experimental AS *directly*, either via a direct peering session or through peering with a route server.

Visibility Condition (AS)

The AS we are testing needs to contain a vantage point.

While this allows us to cast a wider net, we must be aware what implication these changes have. An AS that fulfills both updates conditions might operate a BGP router with a ROV-based filtering policy might still propagate invalid routes to its vantage point, by chance circumventing the router that would filter these routes as shown in Figure 5.1.

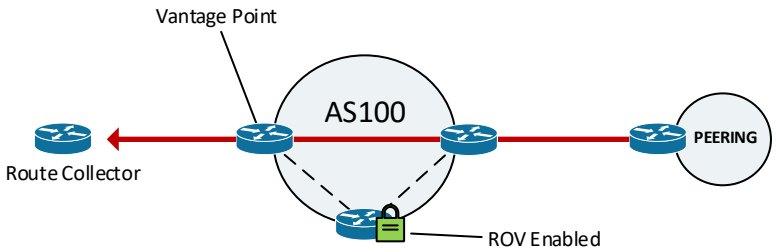


Figure 5.1: AS100 operates one router that filters invalid routes. However, the vantage point receives the invalid route from a different router and exports it.

This means that an AS with a vantage point that exports invalid routes could still have ROV partially deployed.

Experiment Setup

The basic reasoning to determine whether a router drops invalid routes is very simple: We announce a valid route and an invalid route. If the router propagates the valid route, but not the invalid route, it is likely to be filtering. Of course, this hinges on the fact that the router would treat both routes identically, *i.e.,* assigning them the same local preference, if

Prefix	$\mathbf{P_R}$	**Prefix**	$\mathbf{P_E}$
ROA_R		**ROA_E**	
Prefix:	P_R	Prefix:	P_E
Max. Length:	24	Max. Length:	24
ASN:	47065	ASN:	47065
Route for P_R		**Route for P_E**	
ASN:	47065	ASN:	47065
Validity State:	valid	Validity State:	valid

Table 5.9: Exact matching ROAs for P_E and P_R authorizing AS_E to originate.

they were both valid. These routes also must be for two different prefixes, or else the router can choose only one of them to propagate. We thus announce two prefixes:

Reference Prefix P_R
This prefix serves as a reference and throughout the experiment the RPKI validity state of its BGP announcement will not change.

Experiment Prefix P_E
The RPKI validity state of the BGP announcements for this prefix will change throughout the experiment.

To increase the likelihood of the router assigning different local preferences to the routes for arbitrary reasons, we select the prefixes to be very similar to each other. This means that:

- The prefixes are both of length 24.
- The prefixes are both part of the same /20 covering prefix.
- The prefixes both have the same route object in the RIPE Routing Registry. This is done since it has been shown that in some cases route objects are used to filter routes [22].

In addition to that, both prefixes are announced always at the same time, from the same PEERING mux, via the same peers. **We announce via multiple peers at once, since targeting each AS that fulfills the reachability and visibility conditions separately is not a scalable approach**. The origin ASN for both announcements is PEERING's AS47065.

The experiment involves creating ROAs for both P_E and P_R. We refer to a ROA that pertains to prefix P_E as ROA_E, and a ROA that pertains to P_R as ROA_R. For this experiment, the prefixes specified in ROA_E and ROA_R *exactly* match P_E and P_R. The maximum length field of ROA_E and ROA_R is set to the length of P_E and P_R respectively, which is 24 in both cases. Initially, we create ROAs for both prefixes with ASN set to PEERING's AS47065. This means routes for both prefixes have RPKI validity state valid. Table 5.9 shows the contents of both ROAs.

We require two initial conditions to be met before we can start conducting our experiment:

BGP Convergence
A vantage point that we are testing must have adopted stable, identical, direct routes for both P_E and P_R. The purpose of this is to ensure that the AS the vantage point

is in is treating both prefixes identically. To reach BGP convergence can take tens of minutes, as temporary routing table oscillations can occur during the best path selection [49].

RPKI Convergence

Both ROA_E and ROA_R have been published in the global RPKI. Enough time has to pass for this information to reach BGP routers. The propagation of this information involves i) the transfer of the global RPKI to cache servers and ii) the transfer of validated ROA information from cache servers to BGP routers using the rpki-rtr protocol (RFC6810). For i), widespread cache implementations download the global RPKI at a 10 minute [27] or 60 minute interval. In addition to that, older implementations of RPKI cache servers use *rsync* to transfer this data. Flat, non-hierarchical, repository structures have been shown cause excessive overhead during the data transfer, bloating the transfer duration to up to 30 minutes [62]. For ii), RFC6810 specifies a *Serial Notify* PDU that cache servers can use to alert routers to ROA changes [28]. However, this is an optional feature. The regular method for routers to obtain new ROA information is to send a *Serial Query* to cache servers, which will then respond with all ROA changes that have occurred since the last query from this router was received. RFC6810 does not specify a default refresh interval for these queries. The BGP origin validation implementation of Juniper, included in Junos12.2 or higher, does not specify a default value. There is however a tutorial by RIPE NCC [56] which sets the refresh rate to 120 seconds. Similarly Cisco's BGP origin validation implementation, included in XR4.2.1 or higher and XE3.5 or higher, does not define a default refresh interval but Ciscos documentation for this feature uses 600 seconds in examples. Finally, the open source implementation RTRlib [68] also provides no default values but uses a 30 second refresh interval in its tutorials [53].

In any case i) requires a much larger data transfer than ii), as well as CPU-intensive validation of Resource Certificates, it is possible that some operators have specified a longer download interval than the implementation defaults. To be safe, we allow for at least 4 hours of RPKI convergence time before we proceed with the experiment.

Figure 5.2: The vantage point of a target AS must choose the same direct route for both prefixes.

Once both conditions have been met, we can proceed by manipulating ROA_E. To be more precise, we remove ROA_E from the global RPKI by revoking the EE Resource Certificate contained within the ROA. Since now there exists no ROA that matches P_E, the BGP announcement of that prefix has RPKI status *unknown*. We then immediately issue a new ROA_E, this time authorizing AS51224 to originate P_E. The prefix in the new ROA_E still matches P_E exactly and has maximum length 24. The ASN 51224 is not owned by PEERING and is not announcing any IP space used in our experiments, however we are authorized to use this ASN. We refer to the process of revoking a ROA and reissuing it

Prefix	$\mathbf{P_R}$	**Prefix**	$\mathbf{P_E}$
ROA_R		**ROA_E**	
Prefix:	P_R	Prefix:	P_E
Max. Length:	24	Max. Length:	24
ASN:	47065	ASN:	51224
Route for P_R		**Route for P_E**	
ASN:	47065	ASN:	47065
Validity State:	valid	Validity State:	invalid

Table 5.10: After flipping ROA_E, routes for P_E are invalid. ROA_R and routes for P_R are unchanged.

with a different ASN (but identical prefix and max. length) as a *ROA flip* or *ROA change*. Table 5.10 shows the contents of both ROAs, route origins and validity state after the ROA flip.

Once the new ROA_E has been downloaded by cache servers, validated, and then distributed to BGP routers, the announcement for prefix P_E will be *invalid*, since the AS originating the prefix (AS47065) does not match the AS in ROA_E, which is the only ROA published for P_E.

5.2.2 Experiment Analysis

After ROA_E has been flipped, we observe whether any AS that fulfills both the visibility and the reachability conditions changes it's route for prefix P_E, *i.e.,* whether the vantage point contained in the AS now exports a different route for P_E. If we observe a vantage point changing it's route for prefix P_E, but not for prefix P_R, it is likely to be enforcing a routing policy that uses ROV results. It is crucial that the route for prefix P_R remains the same, as this prefix serves as a reference: Had the vantage point changed it's route for prefix P_E for reasons unrelated to RPKI status, we'd expect that reason to also cause a route change for prefix P_R. A vantage point that initially has chosen identical, direct, routes for both prefixes will typically react to the ROA flip in one of three ways:

No change

The vantage point does not change its route for either prefix. In this case there is no indicator that the vantage point is using ROV to drop invalid routes, nor that another BGP router of the same AS that has propagated the route to the vantage point is using ROV to drop invalid routes. It is important to note that this does not mean *the AS has not deployed ROV on one of its routers*, the route simply may have no propagated to the vantage point via one of such routers, as illustrated in Figure 5.1. Also, a BGP router might choose to only use ROV to drop invalid routes when the route was learned from a particular peer. For example, an AS might choose to drop invalid routes learned from AS they have a peer-to-peer relationship with, but never from AS they have a customer-provider or provider-customer relationship with. We call such a policy **selective filtering** of invalid routes, and the peers that the BGP router accepts invalid routes from *allowed peers* and the peers it rejects invalid routes from *forbidden peers.* the vantage point not changing its route could be because

PEERING is amongst the allowed peers. Overall, all we can learn from a vantage point not changing it's route for P_E after the ROA flip is that it is not using ROV to drop invalid routes learned from PEERING.

No route for P_E

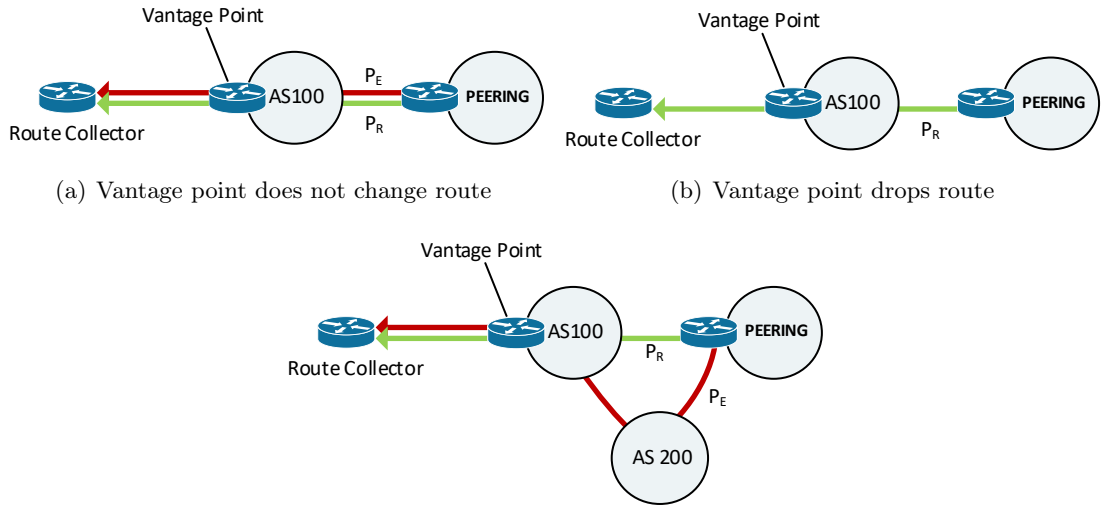
The vantage point drops its route for prefix P_E and does not adopt a different one. In this case there is a strong indicator that the AS that contains the vantage point has deployed ROV. It must not necessarily mean that the vantage point itself is filtering invalid routes, it is possible that the router that propagates our announcement to the vantage point is doing the filtering, or that both (or even all routers) have such a policy. We also cannot know for certain whether the AS is using selective filtering, since the reason the vantage point does not adopt a different route from an allowed peer might simply be for a lack of such a route.

Different route for P_E

The BGP router drops its route for prefix P_E and adopts a different route via a third AS. In this case there is also a strong indicator that AS that contains the vantage point has deployed ROV. This observation actually gives us more information than the **No route** observation: We know that the vantage point is not dropping all invalid routes, since it still exported a route for prefix P_E It is also possible that the vantage point has no ROV related policy at all. The router that propagated the original route might use (*i*) selective filtering, or (*ii*) drop all invalid routes and the vantage point has learned the new invalid route over yet another router. There is a third option, which is that (*iii*) no router is actually dropping any invalid routes at all. The ROA flip instead might lead to a router in the AS to assign a lower preference to the route learned directly via PEERING, but not to the route learned via the third AS. In this case, the vantage point will choose the route via the third AS without any invalid routes being dropped. This case shows that pinning down the actual policy of an AS, especially one whose interior topology is unknown, is very challenging.

The goal of the experiment is to i) *test whether there exist AS that have deployed ROV on at least one BGP router*, and ii) *are using the validation results to drop invalid routes.* When we observe a vantage point in an AS that fulfills both the visibility and reachability conditions changing its route for prefix P_E after the ROA flip, we can be sure that this AS falls under i) of our experiment goal. If the vantage point has **no route** for P_E, it also pertains to ii) of the experiment goal. In the case of **different route**, we can not be sure whether ii) of our experiment goal is fulfilled or if the a router in the AS is simply assigning different preferences to the now invalid routes. Of course, an AS with a vantage point that adopts a different route is still an interesting result when it comes to measuring ROV deployment!

It is still possible, albeit unlikely, that an observed route change for prefix P_E, but not for P_R, was caused by something unrelated to the ROA flip. Because of this, the experiment is not conducted just once but repeated over a longer time span to confirm that an observed route change for prefix P_E is consistent and due to the ROA flip. In order to accommodate for long ROA propagation times we perform the experiment in 24 hour cycles, *i.e.,* once per day. Since the BGP announcements are never withdrawn, we only have to send them out once and do not have to consider BGP convergence time, except for when we perform a ROA flip since AS with ROV deployed might change their routes for P_E. We cannot choose the



(a) Vantage point does not change route

(b) Vantage point drops route

(c) Vantage points chooses different route

Figure 5.3: Three possibilities for route changes for prefix P_E.

time for the ROA flip arbitrarily, since we only have have knowledge of the routes vantage points have chosen at the time the route collector dumps its RIB. The route collectors for the RIPE RIS project dump their RIB entries at 00:00 UTC, 08:00 UTC, and 16:00 UTC. The route collectors for the Routeviews project dump their RIB entries every 2 hours, starting at 00:00 UTC. Starting with the initial state described in Table 5.9, we flip ROA_E at 04:00 UTC to make the announcement for P_E invalid, giving 4 hours of time for the ROAs to propagate and for AS that have deployed ROV to change their routes. At 08:00 UTC we check the routes vantage points have exported to the route collectors and compare them to the routes they had exported at 00:00 UTC. At 12:00 UTC we flip ROA_E back, making the announcement for P_E valid again. After 4 hours of propagation time, we check the routes vantage points have chosen at 16:00 UTC and compare them to the routes we had observed at 08:00 UTC, when the announcement for P_E was invalid.

Figure 5.4 shows the experiment schedule together with relevant collector dumping times. The previous descriptions dealt with two prefixes only, P_R and P_E, and did not precisely

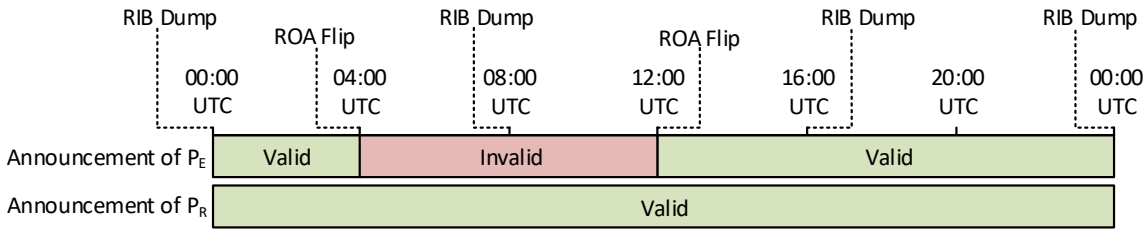


Figure 5.4: Timetable of the basic approach experiments. After the ROA flips we allow for 4 hours of propagation time until the RIB dumps occur.

specify *how* these prefixes have been announced. Considering our experiment goal, ideally

we would like to announce the two prefixes only to one AS at a time. However, the drawbacks of this approach are that (*i*) we lose the ability to detect more subtle policies such as selective filtering, and (*ii*) it introduces a lot of overhead. In subsection 5.1.2 we discuss that there are 74 AS that can be feasible measured using this approach. Announcing a set of prefixes to each of them would be very time consume, since one experiment last for a whole day and should be repeated to confirm any observed route changes.

Instead, we announce multiple pairs of reference/experiment prefixes. Each of these prefix pairs is announced to a different set of peers. Table 5.11 shows the prefix pairs and to which peers they are being announced. At first, we announce only to amsterdam01 and seattle01 since at this point we had only access to a limited number of prefixes, some of which we were planning for further experiment that would run simultaneously to this. After these announcements had propagated, we observed some AS choosing routes that we did not expect: Instead of using the route received directly via PEERING, they preferred a route via a third AS. Not using the direct route for our prefixes effectively removes an AS from the pool of measurable AS, since the third AS introduces ambiguity. In order to circumvent this, we announce more pairs of reference/prefixes exclusively to the route servers at AMS-IX. Some of these third AS did not peer at the route server, or simply chose not to propagate routes learned via the route server. This caused the AS that had chosen the indirect route for the initial prefix pairs to chose the direct route for the new prefix pairs. We announce both to the falcon route server and the default route server at AMS-IX. There were also several AS that chose a route via a third AS that learned the route via the route servers at AMS-IX. To counteract this, we announce another pair of prefixes to all peers of amsterdam01 *except* the route servers at AMS-IX. The final set of prefix pairs and the peers they are being announced to is shown in Table 5.11.

Reference Prefix	**Experiment Prefix**	**Mux**	**Peers**
147.28.240.0/24	147.28.241.0/24	amsterdam01	All
147.28.242.0/24	147.28.244.0/24	amsterdam01	AMS-IX Falcon route server
147.28.243.0/24	147.28.245.0/24	amsterdam01	AMS-IX Default route server
147.28.246.0/24	147.28.247.0/24	amsterdam01	All except route servers
147.28.248.0/24	147.28.249.0/24	seattle01	All

Table 5.11: Pairs of prefixes being announced as part of the experiment. See Section 5.1 for details on PEERING infrastructure.

Preliminary Results

The *basic approach* experiments have been started in February 2017 and running continuously since then. There have been multiple interruptions for technical reasons, such as operational issues at the PEERING testbed and software issues with the RPKI Certificate Authority we have set up. We have observed three AS that are using ROV to filter invalid routes: AS8283, AS3130, and AS50300. Table 5.12 shows the routes that a vantage point in AS8283 has chosen for reference prefix 147.28.246.0/24 (P_R) and experiment prefix 147.28.247.0/24 (P_E) over a span of 5 days.

Vantage point 80.249.211.161 (AS8283)			
	Time	AS Path for P_R	AS Path for P_E
2017-02-20	00:00 UTC	8283←47065	8283←47065
	08:00 UTC	8283←47065	8283←47065
	16:00 UTC	8283←47065	**No route**
2017-02-21	00:00 UTC	8283←47065	**No route**
	08:00 UTC	8283←47065	**No route**
	16:00 UTC	8283←47065	**No route**
2017-02-22	00:00 UTC	8283←47065	8283←47065
	08:00 UTC	8283←47065	8283←47065
	16:00 UTC	8283←47065	**No route**
2017-02-23	00:00 UTC	8283←47065	8283←47065
	08:00 UTC	8283←47065	8283←47065
	16:00 UTC	8283←47065	**No route**
2017-02-24	00:00 UTC	8283←47065	8283←47065
	08:00 UTC	8283←47065	8283←47065
	16:00 UTC	8283←47065	**No route**

Table 5.12: Routes for P_R, P_E exported by a vantage point in AS8283 to collector rrc03. RPKI status of prefix announcement at given time is marked green (valid) and red (invalid).

Both prefixes are announced directly to AS8283 via the *amsterdam01* PEERING mux. We can see that the vantage point in AS8283 always exports the same, direct route for the reference prefix P_R. This is expected behavior, since we never change the announcement for P_R. When we look at which routes the vantage point has chosen for the experiment prefix P_E, we notice a pattern: On all days except for February 21st, the vantage point does not have a route at 16:00 UTC. Since the period in which the announcement for prefix P_E is invalid is between 04:00 UTC and 12:00 UTC, this behavior is unexpected. The fact that the vantage point still has the direct route for P_E at 08:00 UTC, but then drops that route sometime between 08:00 UTC and 16:00 UTC indicates a long propagation delay. In other words AS8283 is filtering the invalid route, but receives the latest ROA data from a cache with a delay of at least 4 hours. We can pin point this delay further by looking at routes from a vantage point in AS23673 has chosen. This vantage point peers with a Routeviews collector, which dumps its RIB entries every 2 hours. Table 5.13 shows the routes this vantage point has chosen at 08:00 UTC, 10:00 UTC, and 12:00 UTC. At 08:00 UTC the exported route is still via AS8283, which is consistent with the routes the vantage point in AS8283 has exported. We see that only at 12:00 UTC does the vantage point in AS23673 choose a different route, presumably because the route via AS8283 has been withdrawn. This withdraw by AS8283 must have occurred between 10:00 UTC and 12:00 UTC. This puts the propagation delay for new ROAs to reach AS8283 at 6 to 8 hours, since we had

flipped the ROA at 04:00 UTC. Keeping this in mind, the route changes of the vantage

Vantage point 203.189.128.233 (AS23673)			
	Time	AS Path for P_R	AS Path for P_E
2017-02-20	08:00 UTC	23673←1299←8283←47065	23673←1299←8283←47065
	10:00 UTC	23673←1299←8283←47065	23673←1299←8283←47065
	12:00 UTC	23673←1299←8283←47065	23673←55329←4788←6939←47065

Table 5.13: Routes for P_R, P_E exported by a vantage point in AS23673 to collector route-views2. RPKI status of prefix announcement at given time is marked green (valid) and red (invalid).

point within AS8283 indicates a ROV-based filtering policy. The exception to the observed pattern on February 21st is that the vantage point exports no route for prefix P_E, on all three dumps. We conducted the experiment the same on February 21st as on the other days, and have observed many other vantage points having a route for prefix P_E during the entire day. We believe that this exception may be caused by AS8283 not recognizing the ROA flip that turns the announcement for P_E valid. The same route changes can also be observed when looking at the routes the same vantage point has chosen for other reference/experiment prefix pairs, which leads us to conclude that AS8283 is not using a selective filtering strategy.

Table 5.14 shows the routes that a vantage point in AS50300 has chosen for reference prefix 147.28.243.0/24 (P_R) and experiment prefix 147.28.245.0/24 (P_E) over a span of 5 days. The route AS50300 chooses is via the default route server at AMS-IX. We can see that the vantage point in AS50300 always exports the same, direct route for P_R, with the exception of February 22nd on which during the first two dumps the vantage point has no route for P_R. We can also see that the vantage point has the same, direct route for P_E at time periods in which the announcement of P_E is valid. For the dumps at 08:00 UTC, when the announcement of P_E is invalid, the vantage point has no route. The same exception as with P_R on February 22nd is observed with P_E. These route changes indicate that AS50300 is using some kind of ROV-based filtering policy. The drop of routes for both prefixes on February 22nd seems to be limited to routes learned via the route server at AMS-IX. This is evident when looking at the routes the vantage point chooses for the reference prefix 147.28.240.0/24 and the experiment prefix 147.28.241.0/24 which are both announced via all peers, including the route server at AMS-IX, of the amsterdam01 mux. Table 5.15 shows these routes.

Since prefixes 147.28.240.0/24 and 147.28.241.0/24 are announced via multiple peers, it is unsurprising that the vantage point in AS50300 has multiple available routes for them. We can see that the direct route is preferred for both prefixes, which is the same route that was chosen for prefixes 147.28.243.0/24 and 147.28.245.0/24 (see Table 5.14). On February 20th, 23rd and 24th we can see that instead of having no route for the experiment prefix, as was the case for 147.28.245.0/24, the vantage point chooses an invalid route learned from AS3356. This route was propagated by AS8283, which did not filter it yet because at this time it had not received the newest ROA information yet. These routing changes indicate

Vantage point 176.12.110.8 (AS50300)			
	Time	AS Path for P_R	AS Path for P_E
2017-02-20	00:00 UTC	50300←47065	50300←47065
	08:00 UTC	50300←47065	**No route**
	16:00 UTC	50300←47065	50300←47065
2017-02-21	00:00 UTC	50300←47065	50300←47065
	08:00 UTC	50300←47065	**No route**
	16:00 UTC	50300←47065	50300←47065
2017-02-22	00:00 UTC	**No route**	**No route**
	08:00 UTC	**No route**	**No route**
	16:00 UTC	50300←47065	50300←47065
2017-02-23	00:00 UTC	50300←47065	50300←47065
	08:00 UTC	50300←47065	**No route**
	16:00 UTC	50300←47065	50300←47065
2017-02-24	00:00 UTC	50300←47065	50300←47065
	08:00 UTC	50300←47065	**No route**
	16:00 UTC	50300←47065	50300←47065

Table 5.14: Routes for P_R (147.28.243.0/24) and P_E (147.28.245.0/24) exported by a vantage point in AS50300 to collector rrc03. RPKI status of prefix announcement at given time is marked green (valid) and red (invalid).

that AS50300 is filtering invalid routes learned via the route server, but not invalid routes learned from AS3356 and possibly other AS. This means that AS50300 employs a selective filtering policy. On February 21st, the vantage point has no route for the experiment prefix at 08:00 UTC. This is likely because the route learned from AS3356 has been withdrawn as a result of the anomaly at AS8283 that we have discussed earlier (see Table 5.12). Since the routes via the route server are being filtered, the vantage point has no route to choose for the experiment prefix. On February 22nd, we see the vantage point choosing the route via AS3356 and AS8283 for both prefixes. This is a result of the anomaly at AS50300 that we have seen in Table 5.14 and indicates that this anomaly must be related to the AMS-IX route server or rather, AS50300's policy regarding the route server.

During the experiments in February, we do not observe AS3130 to be filtering invalids. However, during the second week of experiments in May, we observe AS3130 enabling its ROV related filtering policy, specifically on May 15th. AS3130 operates two vantage points, and only one of them (147.28.7.1) lacks routes for our experiment prefix in the time its announcement is invalid. The routes exported by the other vantage point (147.28.7.2) remain the same for both the reference and the experiment prefix. This indicates partial deployment of ROV related filtering in AS3130. Table 5.17 shows the routes vantage point 147.28.7.1 has chosen on August 1st for reference prefix 147.28.248.0/24 and experiment

Vantage point 176.12.110.8 (AS50300)			
	Time	AS Path for P_R	AS Path for P_E
2017-02-20	00:00 UTC	50300←47065	50300←47065
	08:00 UTC	50300←47065	50300←3356←8283←47065
	16:00 UTC	50300←47065	50300←47065
2017-02-21	00:00 UTC	50300←47065	50300←47065
	08:00 UTC	50300←47065	**No route**
	16:00 UTC	50300←47065	50300←47065
2017-02-22	00:00 UTC	50300←3356←8283←47065	50300←3356←8283←47065
	08:00 UTC	50300←3356←8283←47065	50300←3356←8283←47065
	16:00 UTC	50300←47065	50300←47065
2017-02-23	00:00 UTC	50300←47065	50300←47065
	08:00 UTC	50300←47065	50300←3356←8283←47065
	16:00 UTC	50300←47065	50300←47065
2017-02-24	00:00 UTC	50300←47065	50300←47065
	08:00 UTC	50300←47065	50300←3356←8283←47065
	16:00 UTC	50300←47065	50300←47065

Table 5.15: Routes for P_R (147.28.240.0/24) and P_E (147.28.241.0/24) exported by a vantage point in AS50300 to collector rrc03. RPKI status of prefix announcement at given time is marked green (valid) and red (invalid).

prefix 147.28.249.0/24. These route changes occur consistently over the span of multiple weeks.

The ROA change turning the announcement of P_E invalid occurs at 04:00 UTC, but the vantage point has not dropped the route at the 06:00 UTC dump, yet has done so at the 08:00 UTC dump. From this observation we can infer a propagation delay between 2 and 4 hours. Similarly when ROA flips again at 12:00 UTC turning the announcement for P_E valid again, it takes again between 2 and for 4 hours for the vantage point to pick up the route again.

In summary, AS8283, AS3130, and AS50300 have all been found to be filtering invalid routes. AS50300 uses a selective policy, only filtering routes learned via the AMS IX route server. All 3 cases have been confirmed by contacting the operators.

Relaxing the Connected Condition

The RIPE RIS and Routeviews projects together provide routes from 1099 vantage points in total, distributed in 479 AS out of which 74 AS fulfill the Visibility and the Connected con-

Vantage point 147.28.7.1 (AS3130)			
	Time	AS Path for P_R	AS Path for P_E
2017-08-01	00:00 UTC	3130←47065	3130←46065
	02:00 UTC	3130←47065	3130←46065
	04:00 UTC	3130←47065	3130←46065
	06:00 UTC	3130←47065	3130←46065
	08:00 UTC	3130←47065	**No Route**
	10:00 UTC	3130←47065	**No Route**
	12:00 UTC	3130←47065	**No Route**
	14:00 UTC	3130←47065	**No Route**
	16:00 UTC	3130←47065	3130←46065
	18:00 UTC	3130←47065	3130←46065
	20:00 UTC	3130←47065	3130←46065
	22:00 UTC	3130←47065	3130←46065

Table 5.16: Routes for P_R (147.28.248.0/24) and P_E (147.28.249.0/24) exported by a vantage point in AS3130 to collector route-views2. RPKI status of prefix announcement at given time is marked green (valid) and red (invalid).

ditions. We can increase the number of measurable AS if we relax the *Connected* condition to simply state that an AS must be able to receive routes for PEERING prefixes, however not necessarily directly. This introduces ambiguity, since we need to distinguish between an AS that fulfills the conditions dropping invalid routes or one of the AS that propagates routes to it is dropping invalids, as illustrated in Figure 5.5. We can resolve this ambiguity somewhat by determining whether any third AS on the AS path to the AS with the vantage point are using a ROV related filtering policy. We have two ways to determine this, (*i*) if the AS on the path contain a vantage point themselves, we can simply use our basic approach to determine whether they filter, or (*ii*) if they do not contain a vantage point, we might check whether they appear on AS paths of other invalid routes. Neither of these methods is 100% reliable, in the case of (*i*) its possible the AS on the path is filtering, but the vantage point in that AS still exports invalid routes, similar to the situation shown in Figure 5.1. In the case of (*ii*) the same thing might occur in reverse, whereas the AS might still contain a router using ROV to drop invalids even if they are propagating invalid routes to certain other AS. It's also possible in this case that the AS might use a selective filtering policy. For these reasons we must be careful when inferring ROV based filtering policy of AS that only fulfill the relaxed Connected condition. There are two additional AS that have a vantage point that show route changes that indicate ROV based filtering, AS56730 and AS59715. Table 5.18 shows the routes the vantage point of AS56730 has exported for August 1st to August 4th for the prefix pair that was announced via the route server at AMS-IX. We can see that the vantage point consistently drops the route for the experiment prefix sometime between 00:00 UTC and 08:00 UTC, indicating ROV based filtering. However, a closer look at the AS path shows AS50300 to be on it, which we have already determined to filter invalid

Vantage point 147.28.7.2 (AS3130)			
	Time	AS Path for P_R	AS Path for P_E
2017-08-01	00:00 UTC	3130←47065	3130←46065
	02:00 UTC	3130←47065	3130←46065
	04:00 UTC	3130←47065	3130←46065
	06:00 UTC	3130←47065	3130←46065
	08:00 UTC	3130←47065	3130←46065
	10:00 UTC	3130←47065	3130←46065
	12:00 UTC	3130←47065	3130←46065
	14:00 UTC	3130←47065	3130←46065
	16:00 UTC	3130←47065	3130←46065
	18:00 UTC	3130←47065	3130←46065
	20:00 UTC	3130←47065	3130←46065
	22:00 UTC	3130←47065	3130←46065

Table 5.17: Routes for P_R (147.28.248.0/24) and P_E (147.28.249.0/24) exported by a vantage point in AS3130 to collector route-views2. RPKI status of prefix announcement at given time is marked green (valid) and red (invalid).

routes received via the AMS-IX route server. This means the reason that the vantage point in AS56730 does not have a route for the experiment prefix at 08:00 UTC is likely to be that AS50300 has dropped the route and thus did not propagate it to AS31463 which then in turn could not propagate it to AS56730. There is of course still a chance that AS56730 is filtering. To determine this we must look at other prefix pairs and the routes the vantage point in AS56730 has exported for them. Table 5.19 shows routes for the prefix pair announced via AS3130. We can see that the vantage point in AS56730 chooses the same route for both prefixes at all times, indicating the absence of a ROV related filtering policy.

The case of AS59715 is similar to that of AS56730. Table 5.20 shows the routes exported the vantage point in AS59715 for August 1st. These observations are consistent over the span of multiple weeks. We can see that the vantage point drops the invalid route for the experiment prefix between 04:00 UTC and picks it up again between 12:00 UTC and 14:00 UTC. Similarly to the case of AS56730, there is an AS on the AS path that we have determined to be filtering already: AS3130. There are however strong indicators that it is not AS3130 filtering the invalid routes in this case, but rather AS59715, or any of the other AS on the AS path. First, we know that AS3130 contains two vantage points and only one of them had been observed to be dropping invalid routes, see Table 5.16 and Table 5.17. Second, we observe AS3130 propagating an invalid route for AS56730, as seen in Table 5.19. Third, we note that the router(s) in AS3130 that *are* dropping invalids do it consistently between 06:00 UTC and 08:00 UTC, and reintroduce the route between 14:00 UTC and 16:00 UTC, see Table 5.16. This does not align with the timing of the vantage point in AS59715, which

(a) AS100 is filtering the invalid route.

(b) AS200is filtering the invalid route.

Figure 5.5: The vantage point in AS200 exports the valid route but not the invalid route. The invalid route could have been dropped by either AS.

drops the invalid route between 04:00 UTC and 06:00 UTC and reintroduces it between 12:00 UTC and 14:00 UTC. For these reasons it is reasonable to assume that AS59715 is using ROV to filtering invalid routes. This assumption was confirmed by the operators of AS59715.

5.2.3 Implementation Considerations

It is possible that some ROV implementations do not properly validate routes which could lead to invalid routes not being marked as invalid. If the router in question has a policy to drop invalid routes, a failure to mark invalid routes as such will cause them to be wrongly considered in the best path selection process. In order to test this we have been granted temporary access to a Cisco and a Juniper router. The Juniper device runs JUNOS 14.2R7.5 , the Cisco device runs a custom edition of Experimental IOS 15.3. Both devices perform the initial route origin validation correctly, *i.e.,* assigning the correct RPKI validity status to a route. We have tested this by issuing an exact matching ROA for a test prefix P_{test} authorizing AS47065, and then announcing P_{test} from AS47065 resulting in a valid route. We confirmed that both the Cisco and Juniper implementations mark the route as valid. We then withdraw the route and flip the ROA, changing the AS to AS51224 and then re-announce P_{test} from AS47065 and confirm that both implementations mark the route as invalid. The second functionality we have tested is the re-evaluation of *existing* routes when their RPKI status changes. If this is not properly implemented, our basic approach experiment will not show an AS to be filtering even if it has such a policy in place. To test this, we again issue an exact matching ROA for P_{test} authorizing AS47065 and announce this valid route to our test routers. We then change the ROA to authorize AS51224 instead but do not withdraw the announcement. We observe that both devices properly re-evaluate the route, mark it as invalid after the ROA change, and re-run best path selection. However, in the case of the Cisco device existing route maps were not re-applied the route. Route maps are a generic matching mechanism that can be used on Cisco devices to shape routing policy. For example, a route map might be used to lower the preference of a route with

Vantage point 195.66.226.20 (AS56730)			
	Time	AS Path for P_R	AS Path for P_E
2017-08-01	00:00 UTC	56730←31463←50300←47065	56730←31463←50300←47065
	08:00 UTC	56730←31463←50300←47065	**No Route**
	16:00 UTC	56730←31463←50300←47065	56730←31463←50300←47065
2017-08-02	00:00 UTC	56730←31463←50300←47065	56730←31463←50300←47065
	08:00 UTC	56730←31463←50300←47065	**No Route**
	16:00 UTC	56730←31463←50300←47065	56730←31463←50300←47065
2017-08-03	00:00 UTC	56730←31463←50300←47065	56730←31463←50300←47065
	08:00 UTC	56730←31463←50300←47065	**No Route**
	16:00 UTC	56730←31463←50300←47065	56730←31463←50300←47065
2017-08-04	00:00 UTC	56730←31463←50300←47065	56730←31463←50300←47065
	08:00 UTC	56730←31463←50300←47065	**No Route**
	16:00 UTC	56730←31463←50300←47065	56730←31463←50300←47065

Table 5.18: Routes for P_R (147.28.243.0/24) and P_E (147.28.245.0/24) exported by a vantage point in AS56730 to collector rrc01. RPKI status of prefix announcement at given time is marked green (valid) and red (invalid).

invalid RPKI status, or discard it all together. A BGP router not re-applying route maps can therefore lead to invalid routes wrongly being considered for best path selection. Note that this is not the only way of discarding invalid routes, which means that a BGP router with this particular version of IOS might still be filtering invalid routes. We found a second fault with Cisco's implementation, namely the failure to react to *Serial Notify* [28] messages sent by RPKI cache servers. Instead, the Cisco device will wait until the configured refresh interval has expired and then query the cache server for new ROAs. This can lead to slightly longer propagation time of ROA changes, depending on the value the refresh interval was set to.

5.2.4 Basic Approach Revisited

Considering the issue of route maps not being re-applied to route who's RPKI validity status has changed, it is possible that our basic approach has missed some AS that are in fact filtering invalid routes. This is why we must rerun our basic approach with some changes that will allow us to measure whether an AS is using a ROV related filtering policy even if it is using a faulty implementation. In order to force a BGP router using a faulty implementation to re-evaluate a route we can simply withdraw the route first, change the ROA in a way that changes the RPKI validity status of the route, and then after ample propagation time re-announce the route. We do this for both the reference prefix, without the ROA change of course, and the experiment prefix. The experiment will again be performed in a 24-hour cycle, starting at 00:00 UTC and aligning the ROA changes and BGP Updates

Vantage point 195.66.226.20 (AS56730)			
	Time	AS Path for P_R	AS Path for P_E
2017-08-01	00:00 UTC	56730←3356←1239←3130←47065	56730←3356←1239←3130←47065
	08:00 UTC	56730←3356←1239←3130←47065	56730←3356←1239←3130←47065
	16:00 UTC	56730←3356←1239←3130←47065	56730←3356←1239←3130←47065
2017-08-02	00:00 UTC	56730←3356←1239←3130←47065	56730←3356←1239←3130←47065
	08:00 UTC	56730←3356←1239←3130←47065	56730←3356←1239←3130←47065
	16:00 UTC	56730←3356←1239←3130←47065	56730←3356←1239←3130←47065
2017-08-03	00:00 UTC	56730←3356←1239←3130←47065	56730←3356←1239←3130←47065
	08:00 UTC	56730←3356←1239←3130←47065	56730←3356←1239←3130←47065
	16:00 UTC	56730←3356←1239←3130←47065	56730←3356←1239←3130←47065
2017-08-04	00:00 UTC	56730←3356←1239←3130←47065	56730←3356←1239←3130←47065
	08:00 UTC	56730←3356←1239←3130←47065	56730←3356←1239←3130←47065
	16:00 UTC	56730←3356←1239←3130←47065	56730←3356←1239←3130←47065

Table 5.19: Routes for P_R (147.28.243.0/24) and P_E (147.28.245.0/24) exported by a vantage point in AS56730 to collector rrc01. RPKI status of prefix announcement at given time is marked green (valid) and red (invalid).

with the route collector dump times in mind. All announcements are made from AS47065, same as in the basic approach. Table 5.21 shows the initial ROAs for P_R and P_E. Table 5.22 shows the schedule of ROA flips, announcements, and withdrawals.

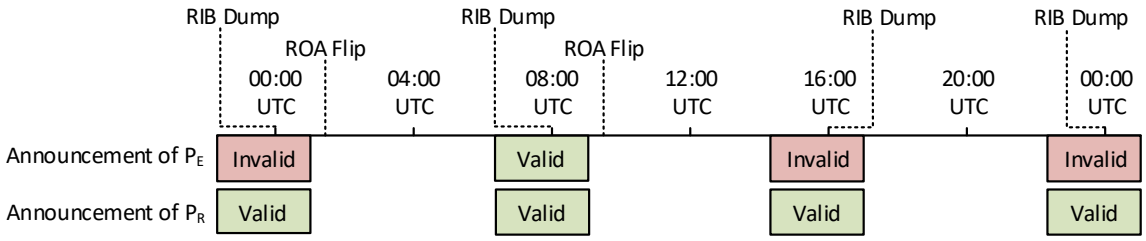


Figure 5.6: Timetable of the basic approach experiments. After the ROA flips we allow for 4 hours of propagation time until the RIB dumps occur.

We observe the routes vantage points have exported to the route collector at the 00:00 UTC dump, when both routes are valid, and the 08:00 UTC dump when the routes for P_E are invalid. Any route changes we observe can be interpreted in the same way as we have done in the original basic approach, since the reasoning of both experiments is the same. Another dump happens at 16:00 UTC, at which the route for P_E is still invalid. This can be used to detect AS that have ROV enabled routers with a unusually long ROA propagation time of

Vantage point 185.5.200.255 (AS59715)			
	Time	AS Path for P_R	AS Path for P_E
2017-08-01	00:00 UTC	59715←3269←6762←1239←3130←47065	59715←3269←6762←1239←3130←47065
	02:00 UTC	59715←3269←6762←1239←3130←47065	59715←3269←6762←1239←3130←47065
	04:00 UTC	59715←3269←6762←1239←3130←47065	59715←3269←6762←1239←3130←47065
	06:00 UTC	59715←3269←6762←1239←3130←47065	**No Route**
	08:00 UTC	59715←3269←6762←1239←3130←47065	**No Route**
	10:00 UTC	59715←3269←6762←1239←3130←47065	**No Route**
	12:00 UTC	59715←3269←6762←1239←3130←47065	**No Route**
	14:00 UTC	59715←3269←6762←1239←3130←47065	59715←3269←6762←1239←3130←47065
	16:00 UTC	59715←3269←6762←1239←3130←47065	59715←3269←6762←1239←3130←47065
	18:00 UTC	59715←3269←6762←1239←3130←47065	59715←3269←6762←1239←3130←47065
	20:00 UTC	59715←3269←6762←1239←3130←47065	59715←3269←6762←1239←3130←47065
	22:00 UTC	59715←3269←6762←1239←3130←47065	59715←3269←6762←1239←3130←47065

Table 5.20: Routes for P_R (147.28.248.0/24) and P_E (147.28.249.0/24) exported by a vantage point in AS59715 to collector route-views4. RPKI status of prefix announcement at given time is marked green (valid) and red (invalid).

ROA$_R$	
Prefix:	P_R
Max. Length:	24
ASN:	47065

ROA$_E$	
Prefix:	P_E
Max. Length:	24
ASN:	47065

Table 5.21: Initially, both ROAs authorize AS46075. Announcements of both prefixes will be valid.

longer than 8 hours. The same goes for the RIB dump at 00:00 UTC.

Preliminary Results

The basic approach revisited has yielded the same results as the basic approach. We have again found AS8283, AS50300, and AS59715 to be filtering. AS50300 once again was observed to employ a selective filtering policy. No additional AS have been found. As we have discussed in the analysis of the basic approach already, a negative result does not mean an AS is not using a ROV related filtering policy.

Time	Event	Route Status (P_R)	Route Status (P_E)
07:00 UTC	Announce P_R, P_E	Announced	Announced
08:00 UTC	Route Collector RIB dumps	Announced	Announced
08:15 UTC	Withdraw P_R, P_E	Not Announced	Not Announced
08:30 UTC	Flip ROA_E	Not Announced	Not Announced
15:00 UTC	Announce P_R, P_E	Announced	Announced
16:00 UTC	Route Collector RIB dumps	Announced	Announced
08:15 UTC	Withdraw P_R, P_E	Not Announced	Not Announced
23:00 UTC	Announce P_R, P_E	Announced	Announced
00:00 UTC	Route Collector RIB dumps	Announced	Announced
00:15 UTC	Withdraw P_R, P_E	Not Announced	Not Announced
00:30 UTC	Flip ROA_E	Not Announced	Not Announced

Table 5.22: Schedule of the basic approach revisited. We allow for 7:30 hours of ROA propagation time and 1 hour of BGP convergence time.

5.2.5 Advanced Approach

Experiment Goal

The experiments we have conducted so far have focused on measuring AS that have a policy to drop invalid routes altogether. This aligns with the purpose of the RPKI, which is to secure the AS level Internet against unauthorized announcements. Unfortunately, deployment of the RPKI and dropping of invalid routes especially, offers no real economic incentive at this time that would push big service providers to adoption. In fact, dropping invalid routes can in some cases actually have the opposite effect. Dropping invalid routes can lead to a loss of connectivity for the users of the AS, since there might not be a non-invalid alternative route for a prefix. Operator gossip suggests that this is one of the major reason that networks are hesitant to enable ROV based filtering. However, it is possible that there are AS out there using ROV related policies that are less aggressive than flat out dropping invalids. For instance, a router might have a policy to accept invalid routes but only in absence of any non-invalid (valid or unknown). Once a non-invalid alternative route appears, the router might switch to that route regardless if the invalid route is more attractive in the absence of RPKI validity information. Another possible policy is that a router will simply decrease the local preference if a route is invalid. While this makes it more likely that a valid route, if present, is preferred, the valid route might also have an ever lower preference due to other factors causing the router to still choose the invalid route. Our next experiment aims at finding AS that have deployed a policy that prefers valid to invalid routes, either categorically or by lowering the preference of invalid routes.

Experiment Goal

Testing whether there exist AS that (*i*) have deployed ROV on at least one BGP router, and on that router are (*ii*) using the validation results to prefer non-invalid over invalid routes for the same prefix.

Similarly as for the basic approach, we posit two conditions an AS must fulfill in order for us to measure whether it is using such a policy:

Reachability Condition (AS)
The AS we are testing needs to be able to receive BGP Updates from our experimental AS *directly*, either via a direct peering session or through peering with a route server.

Visibility Condition (AS)
The AS we are testing needs to contain a vantage point.

The same reasoning for these conditions apply: Direct reachability is necessary to ascribe observed route changes to the AS responsible. Visibility is necessary in order to view the routes the AS, or rather some router in the AS, has chosen.

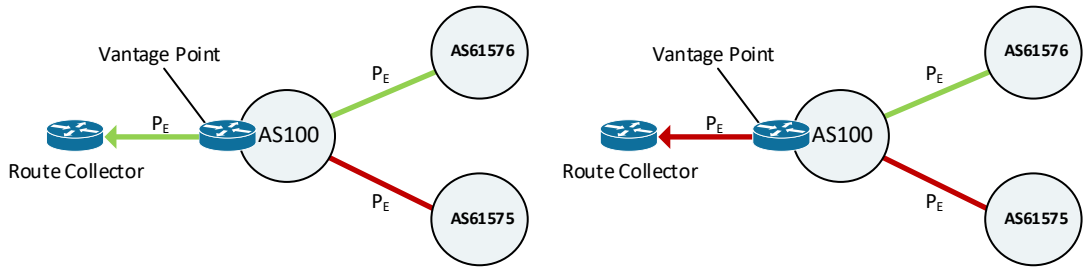
Experiment Setup

In order to test whether some router in an AS is preferring valid routes over invalid routes there must first be at least one valid and one invalid route available. This means that we must announce the same prefix in two different ways, one announcement being valid and one being invalid. Within the framework of the RPKI an announcement can be invalid because of (*i*) prefix length exceeds the maximum length specified in the ROA, or (*ii*) the origin AS of the announcement is not authorized by a ROA. Both cases of course require at least one ROA for the prefix, or a less specific covering prefix, to exist. For our experiment setup, we would like to have a valid and an invalid route for the *same* prefix, to maximize the likelihood that the announcement are treated the same in the absence of ROV related policies. For this reason, we cannot use (*i*) to create an invalid route since it would require a prefix with a different length than the one that is being announced validly. This means we have to create an invalid route via method (*ii*). This involves announcing the same experiment prefix from two different origin AS at the same time. One origin AS will be authorized to announce, resulting in a valid announcement. The other origin AS will not be authorized, resulting in an invalid announcement. A router receiving both an invalid and a valid announcement for the same prefix will have to chose one. If the router chooses the invalid route, we know that there is *not* a policy in place to categorically prefer valid over invalid routes. If the router chooses the valid route, we get no new information: It is possible that the router (*i*) is using a ROV related policy to prefer the valid route or(*ii*) is not using a ROV related policy and has chosen the valid route based on other factors unrelated to RPKI validity status. This is similar to the reasoning in the basic approach, where the absence of an invalid route need not mean that it was filtered. As with the basic approach, in order to distinguish between (*i*) and (*ii*), we must announce a *reference prefix*. We announce the reference prefix in the same way as the experiment prefix, *i.e.,* from two different origin AS. The AS numbers we are using for this are AS61575 and AS61576. A vantage point in an AS that fulfills both necessary conditions will now receive routes for the reference prefix from these two origin AS, both routes being valid, as well as routes for the experiment prefix from the two origin AS, with the routes from one origin being valid and the routes from the other being invalid. In the absence of ROV based policy, we'd expect an AS to chose the same routes for the reference prefix and for the experiment prefix, regardless whether the route for the experiment prefix is valid or invalid. This reasoning is based on the assumption that a router will chose the most attractive route out of all available routes for a certain prefix.

RFC4271 shows this assumption to be true, specifying the route selection phase of BGP as *responsible for choosing the best route out of all those available for each distinct destination /.../* [61]. However, we must remember that a route's attractiveness is specified by the *local preference* value it was assigned. BGP will choose the route with the highest preference, but must use other attributes as tie-breaker in case two routes for the same prefix have been assigned the same local preference value. These are the tie breaking attributes specified in RFC4271 are as follow:

1. AS Path Length
2. Origin
3. Multi Exit Discriminator
4. EBGP/IBGP
5. Interior Cost
6. Router ID Value
7. Peer Address

For a more detailed explanation of the individual attributes see Chapter 2. In additional to these tie breaking rules, implementations of BGP can have their own rules. For example, Cisco BGP routers have an additional cisco-specific parameter called *weight* which takes precedence over even local preference [5]. They also use route age, as in chose the oldest route in case of a tie, as a tie breaking attribute after *Interior Cost* and before *Router ID Value*. Similarly to Cisco, Juniper also uses a customer path selection algorithm in their implementation which prefers already active routes over new routes in case of a tie, as well as cluster list length in case the route was learned via a route reflector [19]. We want to ensure that a vantage point we are measuring *reliably* chooses one route over another, and does not use something arbitrary like peer address or route age to chose the route which can be misleading. For example, a vantage point might chose a route to origin AS61575 for the reference prefix and a route to origin AS61576 for the experiment prefix. Without loss of generality, lets say the announcement of the experiment prefix from origin AS61575 is invalid, while the announcement from origin AS61576 is valid. We also assume, for this example, that the vantage point is using route age as a tie breaker and does not have a ROV related policy. If the vantage point now receives two equally attractive routes for both prefixes, it might happen that for the reference prefix the announcement of origin AS61575 arrives earlier than the announcement of AS61576, and opposite for the experiment prefix the announcement of AS61576 arrives earlier than the one of AS61575. The vantage point will now chose the route to AS61575 for the reference prefix and the route to AS61576 for the experiment prefix. As researchers, we must be careful not to misattribute these route choices to a ROV related policy that prefers valid to invalid routes, and that in the absence of RPKI the vantage point would have chosen the route to AS61575 as it had done with the reference prefix. This is illustrated in Figure 5.7. In order to solve this problem, we must ensure that for both prefixes, not all available routes are equally attractive. This prevents the vantage points from having to rely on arbitrary attributes such as route age or peer address. Looking at the tie breaking attributes we have listed above, it is obvious that some of them such as Interior Cost are completely out of our control. There are also attributes such as peer address or MED which we cannot control as simple users of PEERING, but

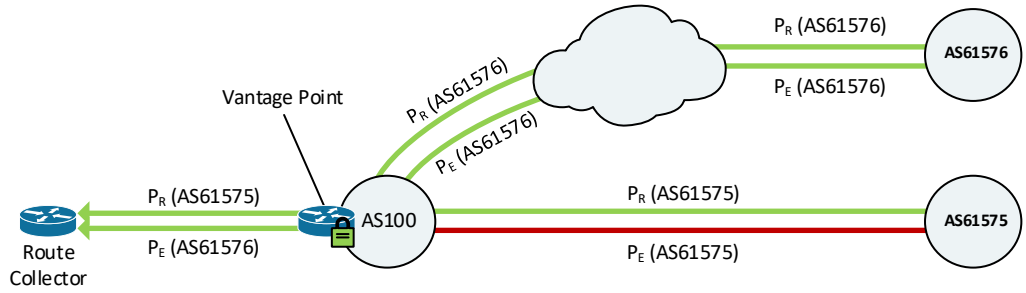


(a) The announcement from AS61576 arrives earlier.

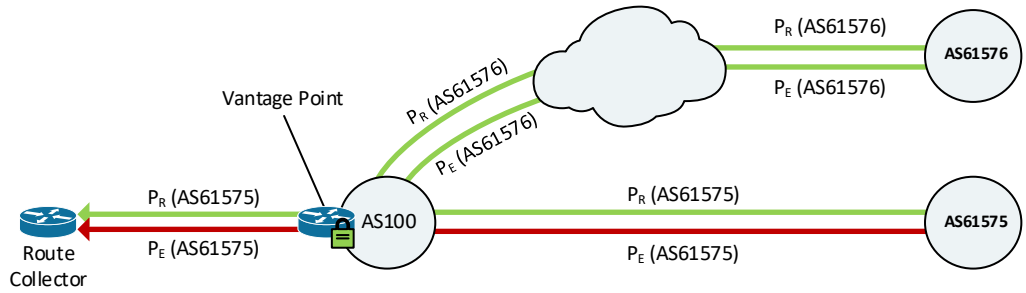
(b) The announcement from AS61575 arrives earlier.

Figure 5.7: The vantage point in AS100 uses route age to break the tie between two competing routes for P_E. The older route is preferred.

would need operative access to PEERING devices. There are some attributes however, that we can exert some control over. For instance, route age can be easily manipulated by us by announcing a prefix with origin AS61575, waiting a few minutes and then announcing the prefix with origin AS61576. A router using route age as a tie breaker will then chose the older route to AS61575. However, since route age is not part of the official RFC there is no guarantee that it will be used by all vantage points we want to test. The other attribute we can manipulate somewhat is the length of the AS Path. For example, we can make one route less attractive by prepending our own AS number to the path. This means we could announce the prefix with two different origin AS via the same mux, one announcement with a prepended AS path, and ensure that vantage points will chose the non-prepended announcement. Unfortunately, PEERING at this time does not allow announcing the same prefix via the same mux at the same time. This forces us to announce the prefix over two muxes. In this case, since the muxes are at different physical locations, the likelihood a vantage point receives two equally attractive routes is lowered. It also means that some vantage points will prefer routes to one mux and some vantage points routes to another mux. When announcing from two physical locations we need to remember the reachability condition we have posited for AS that we'd like to measure. Recall that an AS must be directly connected to PEERING. If we announce from two muxes, it possible that such an AS either (*i*) peers with only one mux directly, or (*ii*) peers with both muxes directly. In the case of (*i*), we know that routes received originated by the direct mux will have a shorter AS path than routes originated by the indirect one. This ensures that a vantage point in the AS won't use arbitrary attributes such as a peer address to chose which route to take. In the case of (*ii*), it is also very likely that the routes originated by the two muxes are not equally attractive. Since they originate from two different physical locations, it is likely the vantage point receives them via different routers inside the AS. This means that the interior cost of the two routes will differ. It is also possible that the vantage point is an edge router and learns one route directly from a mux, while learning the other route via other routers in the AS. In this case the vantage point will prefer the route learned via EBGP instead of the one learned via IBGP. In any case, announcing the same prefix from two different physical locations will likely lead to vantage points *reliably*, *i.e.,* without reliance on arbitrary attributes, choosing one route over the other. If we do happen to observe a vantage point arbitrarily choosing routes, we can still influence it by prepending the AS path



(a) The vantage point exports the prefers the longer, but valid, path for P_E and the direct path for P_R. This indicates that the invalid route from AS61575 was de-preferenced, while the valid one was not.



(b) The vantage points exports the direct path to AS61575 for both prefixes, regardless of RPKI validity state.

Figure 5.8: Competing announcements for the same prefixes can be used to discern whether an AS prefers valid over invalid routes.

of one route. Figure 5.8 illustrates the experiment setup with competing announcements for the reference prefix, both valid, and competing announcement for the experiment prefix, one valid and one invalid. Sub-figure (a) shows an AS preferring valid to invalid routes, by choosing the route to AS61576 for the experiment prefix while choosing the route to AS61575 for the reference prefix. (b) shows an AS choosing routes to AS61575 for both prefixes, regardless of RPKI validity state of the announcement of the experiment prefix. The setup shown in Figure 5.8 shows AS preferring the route to AS61575 over the route to AS61576. Naturally, there will be AS where the opposite is the case and the route to AS61576 is more attractive. The setup presented in Figure 5.8 will then be ineffective since the valid route for experiment prefix will be the more attractive one, even in the absence of ROV related policy. We can fix this problem easily, by alternating the RPKI validity state of the routes for the experiment prefix. To do this we change the ROAs the experiment prefix in such a way that the announcement with origin AS61576 becomes invalid and the one with origin AS61575 becomes valid. As with previous experiments, we conduct this experiment in a 24 hour cycle. Table 5.23 shows the initial ROAs for both prefixes as well as the routes being announced and their validity states. Initially, we configure ROAs for both prefixes in such a way that both announcements from both origin AS are valid.

Prefix	$\mathbf{P_R}$	**Prefix**	$\mathbf{P_E}$
ROAs for P_R		**ROAs for P_E**	
Prefix:	P_R	Prefix:	P_E
Max. Length:	24	Max. Length:	24
ASN:	61575	ASN:	61575
Prefix:	P_R	Prefix:	P_E
Max. Length:	24	Max. Length:	24
ASN:	61576	ASN:	61576
Routes for P_R		**Routes for P_E**	
ASN:	61575	ASN:	61575
Validity State:	valid	Validity State:	valid
ASN:	61576	ASN:	61576
Validity State:	valid	Validity State:	valid

Table 5.23: Initially, routes originated by both origin AS will be valid for both prefixes.

We give ample time for BGP convergence and ROA propagation to occur, and then remove the ROA for P_E which contains AS61575. This turns the routes for prefix P_E with origin AS61575 invalid. Table 5.24 shows the new configuration of ROAs and Routes for the experiment prefix.

This configuration allows us to test any AS that fulfills the (*i*) reachability condition, (*ii*) visibility conditions, and (*iii*) prefers the routes to origin AS61575. After ample propagation time, we then remove the remaining ROA for prefix P_E and issue a new ROA for the same prefix authorizing AS61575. Table 5.25 shows the new configuration, under which routes for P_E originated by AS61576 are valid and routes originated by AS61576 are invalid.

This new configuration allows us to test AS that we have missed with the previous one, namely those that prefer routes originated by AS61576. Note that the configuration for P_R is untouched throughout the experiment, and remains as described in Table 5.23. As with previous experiments, we must be careful to align the ROA changes with the dump times of the route collectors. We perform the first ROA change, the one making announcements of P_E with origin AS61575 invalid, at 00:30 UTC. This allows for 7:30h for the ROA changes to propagate to the routers until the 08:00 UTC dump. At 08:30 UTC we perform the second set of ROA changes, turning the announcements for P_E with origin AS61575 valid and the ones with origin AS61576 invalid. Again we allow for 7:30h of propagation time until the 16:00 UTC dump. At 16:30 UTC we reset the ROAs to the initial state shown in Table 5.23, turning all announcements for P_E, regardless of origin AS, valid. This again allows for 7:30h of propagation time until the 00:00 UTC dump.

For this experiment we again use two prefixes that are as similar as possible to one another, *i.e.,* they both have length 24, are part of the same /23 prefix, and are associated with the same route object. To create competing announcements for the same prefix, we must

Prefix	P_E
ROAs for P_E	
Prefix:	P_E
Max. Length:	24
ASN:	61576
Routes for P_E	
ASN:	61575
Validity State:	invalid
ASN:	61576
Validity State:	valid

Table 5.24
After removing one ROA for prefix P_E, the route with origin AS61575 becomes invalid.

Prefix	P_E
ROAs for P_E	
Prefix:	P_E
Max. Length:	24
ASN:	61575
Routes for P_E	
ASN:	61575
Validity State:	valid
ASN:	61576
Validity State:	invalid

Table 5.25
We replace the existing ROA for prefix P_E with a ROA authorizing AS61575.

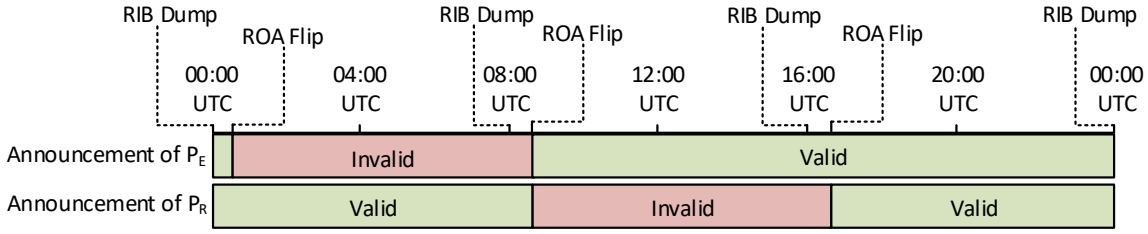


Figure 5.9: Timetable of the advanced approach experiments showing validity state of announcement for P_E for both origin AS. After each ROA change we allow for 7:30h of propagation time.

announce from two different muxes in PEERING. For this we chose the amsterdam01 and seattle01 muxes, since they offer the highest connectivity to other AS. The prefixes we use are 147.28.254.0/24 as our reference prefix P_R and 147.28.255.0/24 as our experiment prefix P_E. We are using AS61575 as the origin for the amsterdam01 announcements, and AS61576 as the origin for the seattle01 announcements. For technical reasons announcements are propagated to AS47065 internally in PEERING. This means that any AS path dumped by the route collectors will have AS47065 before the actual origin AS.

Experiment Analysis

For clarity, we refer to the initial configuration of ROAs described in Table 5.23 as $C1$, the configuration after the first ROA change described in Table 5.24 as $C2$ and the configuration after the second ROA change described in Table 5.25 as $C3$. While $C1$ is active, we observe any AS that fulfills both the visibility and the reachability condition adapts the same route for both the reference prefix and the experiment prefix. Any AS that chooses routes with different origin AS for the two prefixes can not be reliably measured. This is because such

route indicate that the AS is either (*i*) judging a route for a prefix less attractive than the same route for the other prefix, or (*ii*) using arbitrary attributes to break a tie in attractiveness between routes. Both of these reasons make it difficult to infer ROV related policy at later stages of the experiment. This route should also be the direct route to PEERING, with no intermediary AS in between. After the first set of ROA changes occurs and we enter configuration $C2$, there are two possible scenarios. Either (*i*) the route the vantage point in the AS had chosen for the experiment prefix is still valid, or (*ii*) the route the vantage point in the AS had chosen for experiment prefix is now invalid. In the case of (*i*), no changes were made to either of the routes the vantage point in the AS had chosen, so we expect the vantage point to stick with these routes in any case. In the case of (*ii*), if the AS we are measuring has deployed some kind of *prefer valid* policy this could have an effect on the route the vantage point chooses for the experiment prefix. If the vantage point itself uses such a policy, it might de-preference the invalid route and switch to the valid route. If another router in the AS uses such a policy, it might affect which routes are propagated internally to the vantage point and affects the vantage points route selection in this way. Either way, if we obverse the vantage point switching its route for the experiment prefix from the unauthorized to the authorized origin, it points towards a ROV related policy. If this AS wasn't found to be filtering in the previous experiments already, we can classify it at as preferring valid to invalid routes.

Preliminary Results

Unfortunately, the only AS that fulfill the reachability and the visibility condition and have a vantage point that displays the described path changes are AS3130 and AS50300, both of which we have confirmed to be filtering invalid routes in the previous experiments. Table 5.26 shows the routes chosen by the vantage point in AS50300 on August 2nd. We can see that the route for the anchor stays the same, while the route for the experiment prefix changes for the 08:00UTC dump, when the announcement of AS61575 becomes invalid. Note that the new chosen route is still to AS61575 and thus invalid. This is consistent with our previous observations of AS50300 where we have found that it selectively filters routes only learned via the route server but not the ones learned via its providers (AS3356). This observation is consistent over multiple days, not just August 2nd.

Vantage point 176.12.110.8 (AS50300)			
	Time	AS Path for P_R	AS Path for P_E
2017-08-02	00:00 UTC	50300←47065←61575	50300←47065←61575
	08:00 UTC	50300←47065←61575	50300←3356←8283←47065←61575
	16:00 UTC	50300←47065←61575	50300←47065←61575

Table 5.26: Routes for P_R (147.28.254.0/24) and P_E (147.28.255.0/24) exported by a vantage point in AS50300 to collector rrc03. RPKI status of prefix announcement at given time is marked green (valid) and red (invalid).

In the case of AS3130 we also make observations that are consistent with the ones made in previous experiments. Recall that AS3130 operates two vantage points, one of which

we have found to be filtering invalid routes (147.28.7.1), while the other one (147.28.7.2) did not. Table 5.27 shows the routes chosen by vantage point 147.28.71 on September 5th and 6th. The vantage point always prefers the route to seattle01 with origin AS61576 for the reference prefix. For the experiment prefix a switch is made when authorization to announce P_E for AS61576 is revoked at 08:30 UTC. On September 5th there is a period where no route is available at 10:00 UTC for the experiment prefix, this does not occur on September 6th. At 16:30 UTC, both AS are again authorized to announce the experiment prefix. The vantage point switches back to the preferred route with origin AS61576 at 20:00UTC on both days. This puts the propagation delay for the ROA changes between 1.5 and 3.5 hours. These observations underline what we have already discovered with previous experiments, namely that vantage point 147.28.71 drops invalid routes. There is no indicator that it accepts invalid routes but prefers valid routes over them. Vantage point 147.28.7.2, which was found to be not filtering invalids in previous experiments, chooses the same direct route to AS61576 for both prefixes consistently. This gives us no indicator that it is preferring valid over invalid routes. Note that it is of course possible that AS3130 *does* lower the preference for invalid routes, but not to a degree that it could overrule other attributes such as AS path length. If we relax the reachability condition similarly as we did for the basic approach, we have larger pool of measurable AS. Unfortunately, the only additional AS whose vantage points shows changes in selected routes that are consistent with the expected behavior is AS59715, which has been confirmed in previous experiments to drop invalid routes.

5.3 Conclusion

In this chapter we have evaluated the connectivity offered by the PEERING testbed. We have assessed the connectivity as it is documented on the PEERING websites and the websites of the various IXP where the testbed peers with other AS. We have then compared this documented connectivity to the actual connectivity offered by PEERING in practice to assess which AS we can reach with our experiments. We have then presented a methodology with which we can target specific AS to test them whether they are using origin validation results in their routing policy to drop or de-preference invalid routes. Using this methodology we have devised and conducted various experiments to test whether any AS use such a policy. We have also created an experiments specifically aimed at testing AS that are using faulty ROV implementations. With our experiments we have found AS8283, AS50300, AS59715, and AS3130 to be dropping invalid routes altogether. We have found no AS that prefer valid over invalid routes, and no AS that are using a faulty ROV implementation.

Vantage point 147.28.7.1 (AS3130)				
	Time	AS Path for P_R	AS Path for P_E	Authorized AS for P_E
2017-09-05	00:00 UTC	3130←47065←61576	3130←47065←61576	Both
	02:00 UTC	3130←47065←61576	3130←47065←61576	61576
	04:00 UTC	3130←47065←61576	3130←47065←61576	61576
	06:00 UTC	3130←47065←61576	3130←47065←61576	61576
	08:00 UTC	3130←47065←61576	3130←47065←61576	61576
	10:00 UTC	3130←47065←61576	**No Route**	61575
	12:00 UTC	3130←47065←61576	3130←2914←8283←47065←61575	61575
	14:00 UTC	3130←47065←61576	3130←2914←8283←47065←61575	61575
	16:00 UTC	3130←47065←61576	3130←2914←8283←47065←61575	61575
	18:00 UTC	3130←47065←61576	3130←2914←8283←47065←61575	Both
	20:00 UTC	3130←47065←61576	3130←47065←61576	Both
	22:00 UTC	3130←47065←61576	3130←47065←61576	Both
2017-09-06	00:00 UTC	3130←47065←61576	3130←47065←61576	Both
	02:00 UTC	3130←47065←61576	3130←47065←61576	61576
	04:00 UTC	3130←47065←61576	3130←47065←61576	61576
	06:00 UTC	3130←47065←61576	3130←47065←61576	61576
	08:00 UTC	3130←47065←61576	3130←47065←61576	61576
	10:00 UTC	3130←47065←61576	3130←2914←8283←47065←61575	61575
	12:00 UTC	3130←47065←61576	3130←2914←8283←47065←61575	61575
	14:00 UTC	3130←47065←61576	3130←2914←8283←47065←61575	61575
	16:00 UTC	3130←47065←61576	3130←2914←8283←47065←61575	61575
	18:00 UTC	3130←47065←61576	3130←2914←8283←47065←61575	Both
	20:00 UTC	3130←47065←61576	3130←47065←61576	Both
	22:00 UTC	3130←47065←61576	3130←47065←61576	Both

Table 5.27: Routes for P_R (147.28.254.0/24) and P_E (147.28.255.0/24) exported by a vantage point in AS3130 to collector route-views2. RPKI status of prefix announcement at given time is marked green (valid) and red (invalid).

CHAPTER 6

Tooling

In the course of this work we have designed and implemented a toolchain for data analysis and automated presentation of results. We have also implemented tools for experiment control, which can be used outside this thesis by users of the PEERING testbed and the Dragon Research Lab rpki toolkit. This chapter gives an overview of scripts relevant to the presented work.

6.1 Data Analysis: Uncontrolled Experiments

Throughout this thesis, we have analyzed both uncontrolled and controlled experiments. While both experiments yield the same type of data (BGP RIB entries), the amount of data for the types of experiments varies greatly. Our data set for uncontrolled experiment consist of BGP RIB dumps from all route collectors at *one point in time*, which includes *all* exported routes from *all* available vantage points. This data set is 27GB in size. In contrast, the various data sets for controlled experiments span 7 days but are below 15MB in size. This is because in controlled experiments we are only interested in routes exported for the prefixes announced by us through PEERING. Analyzing the entire 27GB data set could be quite time consuming, so we have paid special attention to keeping the complexity of the analysis as low as possible, ideally linear or faster. Since many parts of the overall analysis were made in an exploratory fashion, we had to often re-run certain scripts with some minor parameters changed to gather new results. In order to avoid having a script analyze the entire data set each time, we decided to create a file which stored some meta-information for each vantage point. In this file we recorded the path diversity of each AS, as seen from each vantage point. This could be used by other scripts to pre-filter the RIB data to only include routes originated by AS that we were interested in. For example, to obtain the divergence point distributions shown in Figure 4.14 we used the path diversity file to exclude AS that were not originating at least one divergent pair of invalid and non-invalid routes. The main scripts used to analyze data from uncontrolled experiments are implemented in Python3:

Path Diversity
Input: Complete Data Set
Output: File `path_diversity.csv`, which contains meta-information about the routes

each vantage point has exported.

Vantage Point Visibility
Input: Complete Data Set
Outputs the amount of prefixes, invalid (broken down by reason) and non-invalid, seen by each vantage point, as well as the amount of origin AS for those prefixes.

Covered Invalids
Input: Complete Data Set, `path_diversity.csv`
Analyzes invalid prefixes which are covered by non-invalid prefixes of the same origin AS. Outputs fraction of covered invalid, as well as divergence between the paths of invalids and the paths of their covering non-invalids.

Origin-Prefix Visibility
Input:Complete Data Set
For each vantage point, outputs the average per-origin prefix visibility.

Replication of Gilad et al.
Input: Complete Data Set, `path_diversity.csv`, AS Relationship Data Set (CAIDA)
A replication of the state of the art methodology of Gilad et al. [39]. Outputs AS marked as non-ROV enforcing, ROV candidates, and ROV enforcing for various subsets of the data as well as the complete data set.

Figure 6.1 shows an overview for those scripts and which figures were produced by them.

6.2 Controlled Experiments

6.2.1 Visibility Overview

Prior to starting our controlled experiments, we needed to understand visibility on the Internet better. For this purpose we decided to implement a script that yield a list of currently active vantage points, *i.e.,* routers that have exported *any* prefixes to a route collector. We compare these vantage points with the AS that we can reach with our experiments through PEERING infrastructure. This yields a list of which AS we can expect to export routes for our prefixes once we conduct experiments. This `vantage_point_visibility` script is essentially an implementation of subsection 5.1.2 (Connectivity and Visibility). To get a list of AS peering with route collectors, it uses `pybgpstream` [15], python extensions for the `bgpstream` [59] framework, to download the entire RIB of all route collectors of the Routeviews and RIPE RIS projects. We run this script periodically and present the results in a "BGP Monitor Visibility Table" which we automatically make available on a website. The website is created with `Jekyll` [11] a framework that can be used to generate static websites. It is hosted on `github`, using `GitHubPages` [8], which comes with built in Jekyll support.

The BGP Monitor Visibility Table can be used to quickly assess which AS is peering with which route collector and how it can be reached through PEERING. This functionality is also useful outside the context of this thesis, specifically for other users of PEERING. The table is updated once a day.

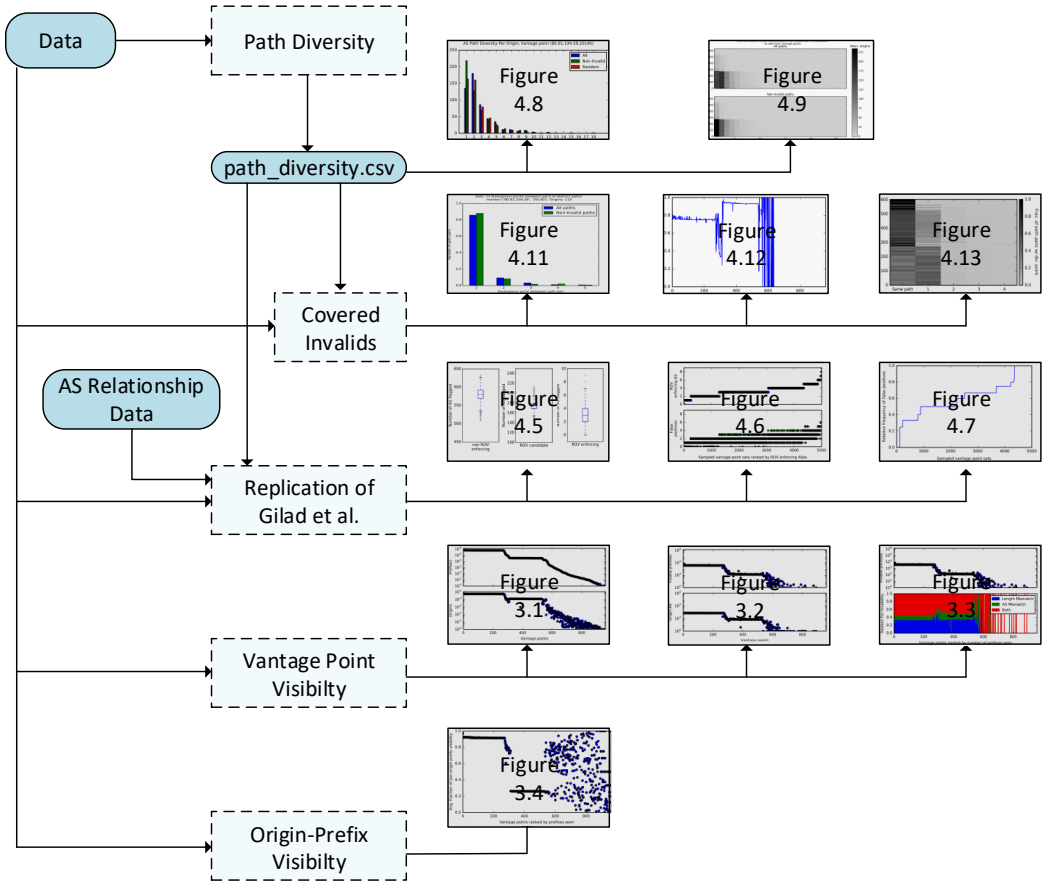


Figure 6.1: An overview over the scripts written for data analysis of uncontrolled experiments.

The BGP Monitor Visibility Table only shows us which AS is actively reachable through PEERING and has a vantage point that is exporting routes to a route collector. While by itself, this is useful for user of PEERING, for our purposes we also wanted to know which AS are exporting routes for prefixes used in our experiments. For this purpose we have written a the `prefix_visibility` script, which yields all AS that *should* export routes for our prefixes (according to the BGP Monitor Visibility Table), and checks whether they actually do. Similarly to the `vantage_point_visibility`, the result of the `prefix_visibility` script are presented in table which is displayed on a website. The `prefix_visibility` scripts is run once per day and the results are then displayed on the website. This way, the website also serves as a kind of archive, which allows us to compare prefix visibility at different points in time. The script also includes some automated data analysis, the results of which are also displayed together with the BGP Prefix Visibility Table. The data analysis identifies AS that do not export routes for our prefixes, or only for a subset of our prefixes, but according to the BGP Monitor Visibility Table *should* be exporting routes for all prefixes. We separate these AS in three categories, (*i*) reachable peers and (*ii*) customers of reachable peers. We also add a third category for situations where an AS and one of its customers both have

prefix visibility, but there is a mismatch, *i.e.,* they are not exporting routes for the same set of prefixes. This analysis gives us a list of AS that we then further investigate. A full explanation of this automated analysis can be found at `http://rpki.github.io/analysis`.

6.2.2 Experimental Facilities

The experiments described in Chapter 5 requires us to inject our own BGP announcements into the Internet as well as to issue our own RPKI objects and publish them within the global RPKI. There exists already specific software to achieve both of these tasks, in the case of BGP announcements it is the `peering-client` [13] and in case of RPKI there is the RPKI toolkit from Dragon Research Labs [7]. However, both these pieces of software were not build for recurring experiments, and do not feature an API that could be used by other programs. They are intended for manual use and are geared towards the use cases of network operators. Some of our experiments require us to periodically withdraw and announce BGP routes, as well as periodically revoke ROAs and issue new ROAs. To accommodate the need for automated control over BGP announcements and ROA issuance, we decided to implement an interface that allows us to specify an experiment in a configuration file and automatically conduct the experiment as well as publish the result on a website: The interface consists of the following components:

Experiment Control

1. Read the provided configuration file and parse experiment parameters
2. Validate the experiment parameters to ensure that the experiment is feasible. By default, this includes checking that the experiment allows for enough propagation time for ROAs and BGP announcements. This behavior can be switched off.
3. Start the experiment by calling the **Experiment Conductor** script with the experiment parameters.
4. After the experiment has ended, fetch the resulting data and run case analysis on it.
5. Post resulting data and analysis classification results to the website.

Experiment Conductor

1. Issue initial ROAs using the **RPKIC Interface**.
2. Schedule announcements and withdrawals of routes using the **Peering-Client Interface**.
3. Schedule issuing and revoking of ROAs using the RPKIC Interface.
4. Schedule clean-up of experiment (withdrawal of all routes, revoking all ROAs).

RPKIC Interface

Interface to issue and revoke ROAs. It is built on the `rpkic` cli tool from the RPKI toolkit of Dragon Research Labs.

Peering-Client Interface

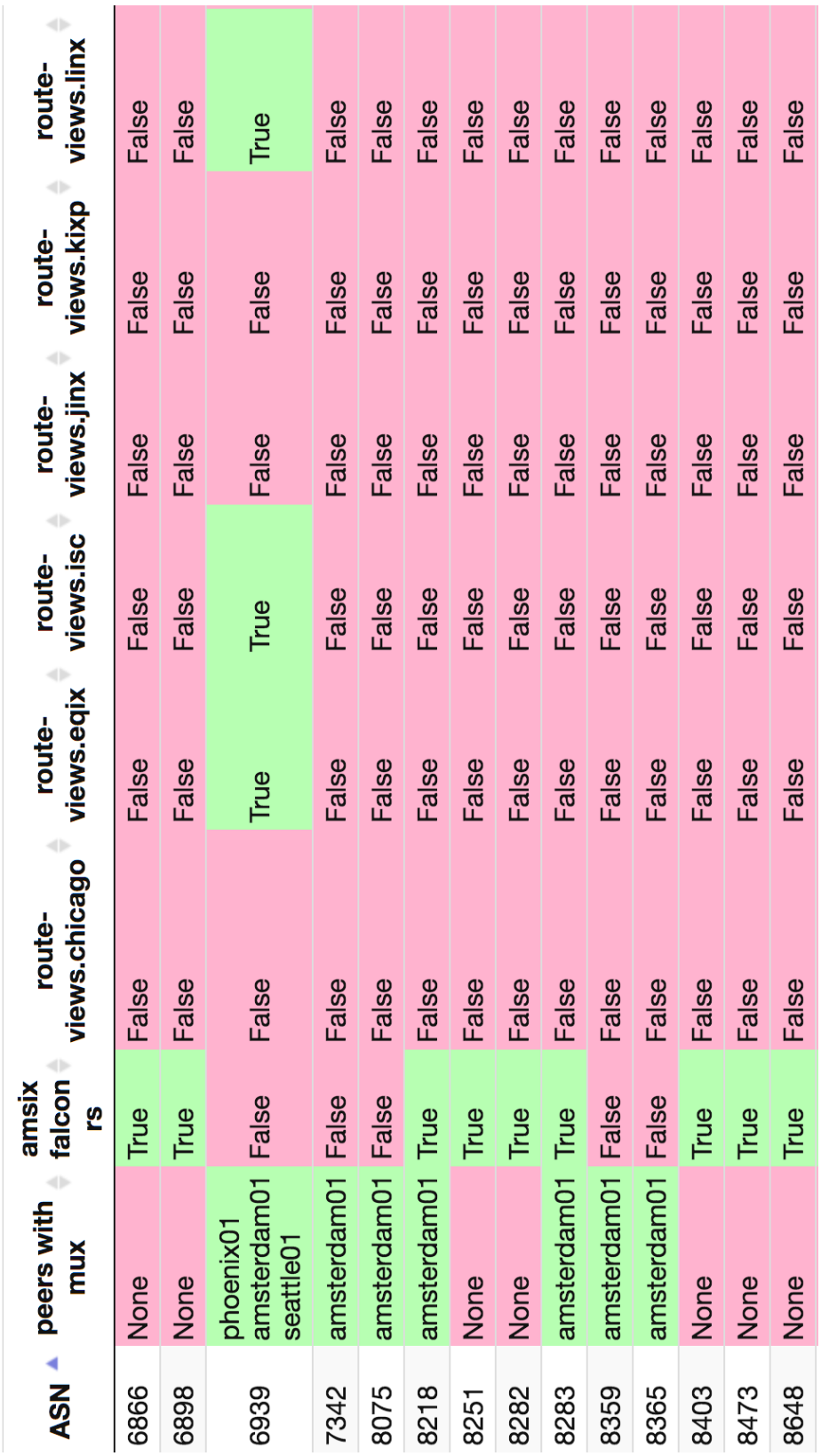
Interface to announce and withdraw BGP routes over the PEERING testbed. It is built

on the `peering-client` [13] cli tool developed by PEERING.

This setup allows us to specify an experiment by its parameters and automatically obtain the related data for further analysis, as well as display prefix visibility results on the website.

6.3 Reproducability

We explicitly support reproducible research [23], and recent efforts to build an ecosystem that incentivizes researchers to support reproducability [64]. We publish the source code for our replication of Gilad et al., as well as the code for the analysis of data obtained from controlled experiments, at `https://github.com/RPKI/rov-measurement-code`.



ASN	peers with mux	amsix falcon rs	route-views.chicago	route-views.eqix	route-views.isc	route-views.jinx	route-views.kixp	route-views.linx
6866	None	True	False	False	False	False	False	False
6898	None	True	False	False	False	False	False	False
6939	phoenix01 amsterdam01 seattle01	False	False	True	True	False	False	True
7342	amsterdam01	False	False	False	False	False	False	False
8075	amsterdam01	False	False	False	False	False	False	False
8218	amsterdam01	True	False	False	False	False	False	False
8251	None	True	False	False	False	False	False	False
8282	None	True	False	False	False	False	False	False
8283	amsterdam01	True	False	False	False	False	False	False
8359	amsterdam01	False	False	False	False	False	False	False
8365	amsterdam01	False	False	False	False	False	False	False
8403	None	True	False	False	False	False	False	False
8473	None	True	False	False	False	False	False	False
8648	None	True	False	False	False	False	False	False

Figure 6.2: Part of the BGP Monitor Visibility Table. For each AS on the left column, it shows how an AS can be reached through PEERING (mux peer or route server), and with which route collectors it peers.

Prefix visibility at RIPE RIS/Routeviews collectors

Time period:
18 Jan 2017 16:00:00 UTC - 18 Jan 2017 16:05:00 UTC

Visibility Table | Analysis

Only show strange cases | 142 items in table

Search for ASN

ASN	Direct peer	Falcon RS peer	Monitor	151.216.32.0/23	151.216.32.0/24	151.216.34.0/23	151.216.34.0/24	147.28.240.0/24
10026	true	false	true	false	true	false	true	true
10310	true	false	true	false	false	false	false	false
10417	true	false	false	false	false	false	false	false
1103	false	true	true	false	true	false	true	true
1200	true	false	false	false	false	false	false	false
12301	false	true	false	false	false	false	false	false
12399	false	true	false	false	false	false	false	false
12414	false	true	true	false	false	false	false	false
12637	true	false	true	false	true	false	true	true
12859	true	false	true	false	true	false	true	true

Figure 6.3: Part of the BGP Prefix Visibility Table. For each AS on the left column, it shows how an AS can be reached through PEERING (mux peer or route server), whether it has a vantage point and whether that vantage point exported routes for our prefixes.

Prefix visibility at RIPE RIS/Routeviews collectors

Time period:
18 Jan 2017 16:00:00 UTC - 18 Jan 2017 16:05:00 UTC

Visibility Table | Analysis

Case 1.1 3 items in table

Search for ASN

ASN	151.216.32.0/23	151.216.32.0/24	151.216.34.0/23	151.216.34.0/24	147.28.240.0/24
1103	false	true	false	true	true
31019	false	true	false	true	true
42541	false	true	false	true	true

Case 1.2 3 items in table

Search for ASN

ASN	151.216.32.0/23	151.216.32.0/24	151.216.34.0/23	151.216.34.0/24	147.28.240.0/24
10026	false	true	false	true	true
10310	false	false	false	false	false
12637	false	true	false	true	true

Case 2.1 0 items in table

Search for ASN

Customer ASN	Provider ASN	151.216.32.0/23	151.216.32.0/24	151.216.34.0/23	151.216.34.0/24	147.28.240.0/24

Case 2.2 0 items in table

Search for ASN

Customer ASN	Provider ASN	151.216.32.0/23	151.216.32.0/24	151.216.34.0/23	151.216.34.0/24	147.28.240.0/24

Case 3.1 0 items in table

p2c pairs where both are monitor: 0

Figure 6.4: Automated analysis included with the BGP Prefix Visibility Table. Classification of AS with limited prefix visibility.

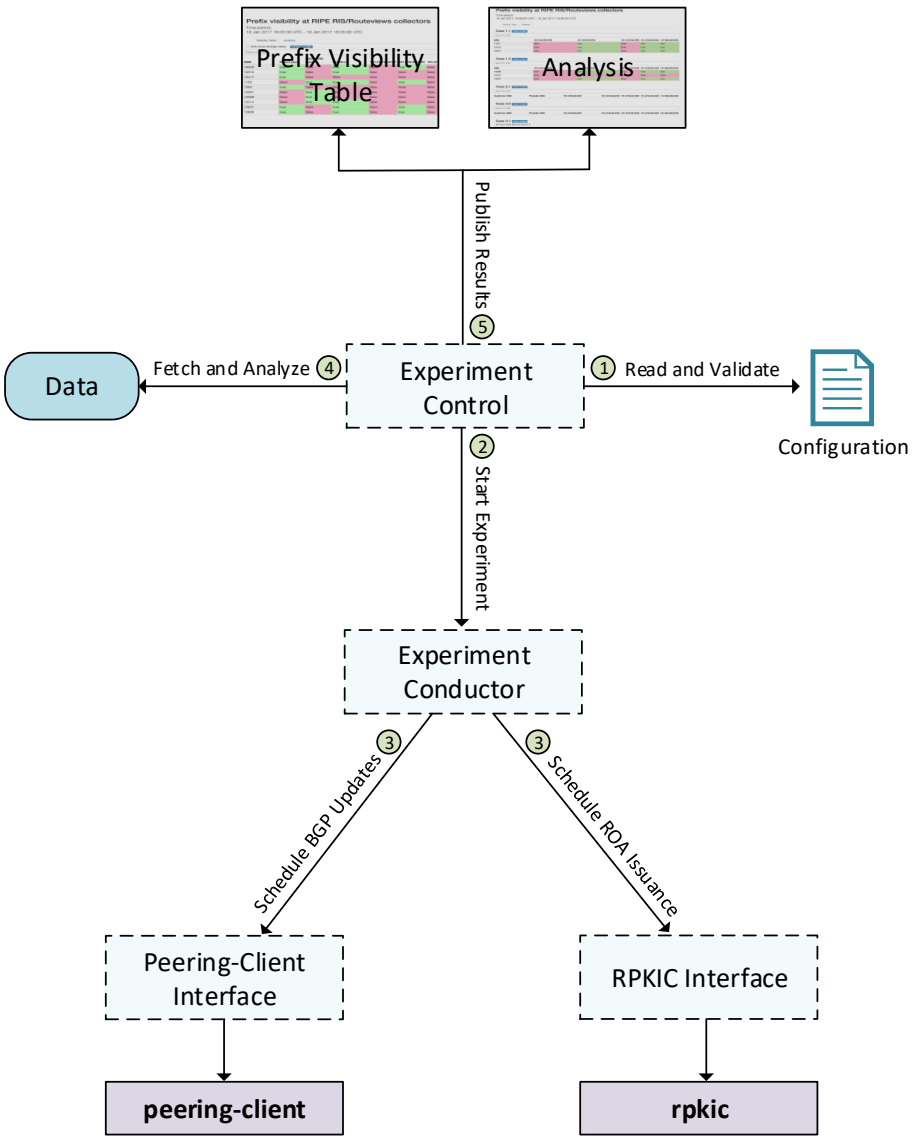


Figure 6.5: Overview of the experiment tooling infrastructure.

CHAPTER 7

Related Work

7.1 Inter-domain Routing

Inter-domain routing, AS-level topology, and BGP measurements, have spawned a plethora of work. Work on routing stability has provided important insight for measurements. While most route announcement and withdrawals will lead to convergence in less than 3 minutes [49], certain routing policies as well as software bugs can increase the delay or even lead to no route convergence [38]. For these reasons, we wait at least 1 hour between announcing/withdrawing a route and measuring the routes adapted by vantage points. The study of BGP route convergence has also resulted in routing guidelines [38] that, if adhered to be a sufficient number of AS, ensure convergence. These guidelines provide a model of AS relationships, with simple categories such as provider-to-customer and peer-to-peer. Models of AS relationships have been discussed further and extended to capture more complex interactions between AS with heuristic-based approaches [36], introduction of new concepts like the customer cone [51], and by combining control plane, data plane, and geolocation data to enhance existing approaches [41]. These findings are helpful to us to understand the propagation of routes for our prefixes. For instance, if we are announcing a route to a peer of PEERING whose customer cone contains an AS with a vantage point, we expect this vantage point to receive a route via this peer.

Using a combination of BGP routing information, obtained from route collectors and looking glasses, and data plane measurements such as traceroutes, mappings of AS topology have been successful in correctly inferring the vast majority of provider-customer links between AS [67, 48, 33]. These findings have been complemented with data plane measurements of the internal structure of IXPs, revealing many previously unknown peer-to-peer links [26]. In a similar vein, work done on the flow data of a large European IXP has shown that the number of peering links inside the IXP is far larger than previously estimated and that AS connect to each other for diverse reasons [24].

Available data has been compared with actual ground truths to further explore the accuracy and completeness of the data [58, 57]. Such work gives perspective on how incomplete inferred AS-level topology using data from route collectors and looking glasses are. For our measurements, this underlines the importance of PEERING connecting to various IXP and

informs the design of our experiments.

The impact of biases in the data obtained from route collectors, looking glasses, and data plane measurements is an important aspect of Internet measurements. These biases have been evaluated and guidelines to design experiments which control for them have been presented [30]. The same work has also provided insight into to the extend of default routes on the Internet and their impact on inferences of AS relationships and routing policies. This will impact our future work, as we plan on augmenting our control plane measurements with data plane measurements which will be affected by default routes.

Many pitfalls and insights of measuring inter-domain routing and inference of topology information have been succinctly expressed in "10 Lessons from 10 Years of Measuring and Modeling the Internet's Autonomous Systems" [63].

The continued research interest in inter-domain routing, and specifically BGP, has produced tools to simplify measurements and easier to deploy at scale. For the work presented here, we have made extensive use of `bgpstream`, a software framework for live and historical BGP data analysis [59] released by CAIDA. Our experimental facilities were provided by PEERING [65], a testbed that offers rich AS-level connectivity and enables active BGP experiments that were previously not feasible to perform for a lot of researchers. PEERING has already provided researchers with important insight into complex routing policies that can not be explained by existing models.

7.2 RPKI

Securing inter-domain routing has been a topic of discussion for many years. The deployment of the RPKI has spawned research that analyzes the resulting data, its evolvement, and the operational challenges that come with it. The relationship between RPKI and web-hosting infrastructure has been empirically explored, showing that less popular websites are more likely to be secured then prominent sites, which are more likely to be hosted within the complex infrastructure of a CDN [70]. An analysis of issued Route Origin Authorizations has shown that the majority of invalid routes are not due to prefix hijack attacks but rather misconfiguration by the operators [69, 47]. This has lead to an increased effort by the Regional Internet Registries to offer further training for operators. The impact of routing policies prioritizing security with existing ROAs has been explored using data from public route collectors [47] as well as with data from simulated environments [52]. Benefits of partial deployment of the RPKI have also been explored in simulated environments [52]. Concerns that policies to drop invalid announcements could be exploited by RPKI authorities to prevent prefixes from being routed for non-security reasons have been a topic of discussion for many years. The potential for such abuse has been explored [35] and possible solutions to this problem have been presented [44]. The deployment of the RPKI has also spawned several tools that, for instance, can be used to explore RPKI repositories [62] or view the RPKI status of a webserver in the browser [71].

7.3 Route Origin Validation

As this work has shown, Route Origin Validation is not widely deployed yet. The only other known work that deals with the deployment of Route Origin Validation does so using uncontrolled experiments [39] and is examined closely in Chapter 4.

CHAPTER 8

Conclusion and Future Work

8.1 Conclusion

In the course of this thesis we evaluate the current state of the art of measuring ROV adoption on the Internet. We identify several limits of the methodology, specifically its inadequate handling of the challenges of limited visibility and limited control. Using our own implementation of the methodology, we demonstrate that these limits are not just theoretical concerns but have a clear impact on the classification output of the methodology. Specifically, we show that the methodology can produce a high number of false positives, as well as contradictory classification when used on different data sets. We argue that these limitations apply to *any* methodology that aims to classify specific AS as ROV enforcing/not ROV enforcing based solely on data from uncontrolled experiments. This does not mean that analysis of data from uncontrolled experiments can not be useful when it comes to understanding ROV adoption measurements. Our analysis shows that invalid routes tend to be routed differently than non-invalid routes, although a likelier explanation for this observation is traffic engineering rather than ROV based filtering.

We improve upon the state of the art with our own methodology, which relies on controlled experiments. By injecting our own routes into the global network, and crafting our own RPKI objects, we are able target specific AS to test whether their routing policy uses ROV. This solves the problem of limited control considerably, since the ability to change our announcements and ROAs means we can distinguish the cause of observed routing decision, which allows us to achieve high accuracy when classifying AS. While the challenge of limited visibility still persists, because there is no complete view of the AS-level Internet, it has no impact on the correctness of classifications anymore. We devise experiments to determine whether (*i*) AS are dropping invalid routes altogether, and (*ii*) AS are preferring valid to invalid routes for the same prefix. We also adapt our experiments to test for faulty ROV implementations. Experiments are conducted using the PEERING testbed. We find only 3 AS that are using a ROV related routing policy, with all 3 of them choosing to drop invalid routes altogether. This is in stark contrast of results produced by the state of the art methodology. We confirm our results with the operators of those AS. We find that at least one AS is using selective filtering, which contradicts the "absolute filtering" assumption on which the state of the art relies on.

Since our experiments rely on visibility and connectivity, we have implemented a website that shows which AS has a vantage point and is reachable via PEERING. As of now, this website is automatically updated, showing which ASes have exported routes for our prefixes. It also displays preliminary analysis of the data resulting from our experiments.

8.2 Future Work

Our automated analysis as of now classifies observed AS according to certain patterns, but does not determine whether an AS is using ROV in its policy. For the future we plan to:

- Expand the automated analysis to determine whether any observed AS is using ROV.
- Set up a longitudinal study with various experiments running in parallel.
- Periodically publish the results of the automated analysis on a website, giving the networking community the means to assess the current state of ROV adoption.

We will also expand the tools we have written for experiment automation. As of now those tools can only handle simple experiments, we will add functionality to handle more complex actions such as:

- Multiple Announcement and Withdrawal cycles in the same experiment.
- Running multiple experiments in parallel.
- Running open-ended experiment that are required for the longitudinal study.

Another improvement to our experiments is the inclusion of routing data from Packet Clearing House [6]. Packet Clearing House operates collectors at a large number of IXPs, which could improve visibility drastically. Future work will also include further investigation into the low per-origin we observe at some collectors. For this, we plan on examining individual vantage points to see if they are really part of the Internet ecosystem at large, or are merely confined to smaller networks within the Internet.

List of Figures

List of Tables

Bibliography

[1] A Chinese ISP Momentarily Hijacks the Internet. http://www.nytimes.com/external/idg/2010/04/08/08idg-a-chinese-isp-momentarily-hijacks-the-internet-33717.html. Accessed: 2017-10-12.

[2] AMS-IX Falcon Class Route Servers. https://ams-ix.net/technical/specifications-descriptions/ams-ix-route-servers/falcon-class-route-servers. Accessed: 2017-07-28.

[3] AMS-IX Route Servers. https://ams-ix.net/technical/specifications-descriptions/ams-ix-route-servers. Accessed: 2017-07-28.

[4] AS Relationships. http://www.caida.org/data/as-relationships/. Accessed: 2017-08-31.

[5] BGP Best Path Selection Algorithm. https://www.cisco.com/c/en/us/support/docs/ip/border-gateway-protocol-bgp/13753-25.html. Accessed: 2017-08-29.

[6] Daily Routing Snapshots. https://www.pch.net/resources/Routing_Data/. Accessed: 2017-11-01.

[7] Dragon Research Labs RPKI Toolkit. https://github.com/dragonresearch/rpki.net. Accessed: 2017-09-28.

[8] GitHub Pages. https://pages.github.com/. Accessed: 2017-10-10.

[9] Global Prefix/Origin Validation using RPKI. https://rpki-monitor.antd.nist.gov/. Accessed: 2017-09-17.

[10] Internet Steering Committee Brazil (IX.br), Belo Horizonte. http://ix.br/particip/mg. Accessed: 2017-07-28.

[11] Jekyll. https://jekyllrb.com/. Accessed: 2017-10-10.

[12] PEERING: ASNs and IP Resources. https://peering.usc.edu/peering/. Accessed: 2017-07-29.

[13] PEERING client controller. https://github.com/PEERINGTestbed/client. Accessed: 2017-09-28.

[14] Phoenix IX. http://www.phoenix-ix.net/. Accessed: 2017-07-28.

[15] PyBGPStream (Python API). https://bgpstream.caida.org/docs/api/pybgpstream. Accessed: 2017-10-10.

[16] Routing Information Service (RIS). https://www.ripe.net/analyse/

internet-measurements/routing-information-service-ris. Accessed: 2017-07-28.

[17] Russian-controlled telecom hijacks financial services' Internet traffic. https://arstechnica.com/information-technology/2017/04/russian-controlled-telecom-hijacks-financial-services-internet-traffic/. Accessed: 2017-10-12.

[18] SIX Route Servers. https://www.seattleix.net/route-servers. Accessed: 2017-07-28.

[19] Understanding BGP Path Selection. https://www.juniper.net/documentation/en_US/junos12.3/topics/reference/general/routing-ptotocols-address-representation.html. Accessed: 2017-08-29.

[20] University of Oregon Route Views Project. http://www.routeviews.org/. Accessed: 2017-07-28.

[21] YouTube - Pakistan Telecom Hijacking. https://stat.ripe.net/events/youtube-pakistan-incident. Accessed: 2017-09-18.

[22] Emile Aben. Propagation of Longer-than-/24 IPv4 Prefixes. https://labs.ripe.net/Members/emileaben/propagation-of-longer-than-24-ipv4-prefixes. Accessed: 2017-06-28.

[23] ACM. Result and Artifact Review and Badging. http://acm.org/publications/policies/artifact-review-badging, Jan., 2017.

[24] Bernhard Ager, Nikolaos Chatzis, Anja Feldmann, Nadi Sarrar, Steve Uhlig, and Walter Willinger. Anatomy of a Large European IXP. In *Proceedings of the ACM SIGCOMM 2012 Conference on Applications, Technologies, Architectures, and Protocols for Computer Communication*, SIGCOMM '12, pages 163–174, New York, NY, USA, 2012. ACM.

[25] Ruwaifa Anwar, Haseeb Niaz, David Choffnes, Ítalo Cunha, Phillipa Gill, and Ethan Katz-Bassett. Investigating Interdomain Routing Policies in the Wild. In *Proc. of ACM IMC*, pages 71–77, New York, NY, USA, 2015. ACM.

[26] Brice Augustin, Balachander Krishnamurthy, and Walter Willinger. IXPs: Mapped? In *Proceedings of the 9th ACM SIGCOMM Conference on Internet Measurement*, IMC '09, pages 336–349, New York, NY, USA, 2009. ACM.

[27] Alexander Band. Github Issue: Update rate. https://github.com/RIPE-NCC/rpki-validator/issues/2. Accessed: 2017-06-28.

[28] R. Bush and R. Austein. The Resource Public Key Infrastructure (RPKI) to Router Protocol. RFC 6810, IETF, January 2013.

[29] Randy Bush. Origin Validation Clarifications. Draft, IETF, October 2017.

[30] Randy Bush, Olaf Maennel, Matthew Roughan, and Steve Uhlig. Internet Optometry: Assessing the Broken Glasses in Internet Reachability. In *Proceedings of the 9th ACM SIGCOMM Conference on Internet Measurement*, IMC '09, pages 242–253, New York, NY, USA, 2009. ACM.

[31] Georg Carle, Jochen Schiller, Steve Uhlig, Walter Willinger, and Matthias Wählisch,

editors. *The Critical Internet Infrastructure (Dagstuhl Seminar 13322)*, volume 3, Dagstuhl, Germany, 2013. Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik.

[32] Ravishanker Chandrasekeran, Paul Traina, and Tony Li. BGP Communities Attribute. RFC 1997, IETF, August 1996.

[33] Kai Chen, David R. Choffnes, Rahul Potharaju, Yan Chen, Fabian E. Bustamante, Dan Pei, and Yao Zhao. Where the Sidewalk Ends: Extending the Internet As Graph Using Traceroutes from P2P Users. In *Proceedings of the 5th International Conference on Emerging Networking Experiments and Technologies*, CoNEXT '09, pages 217–228, New York, NY, USA, 2009. ACM.

[34] D. Cooper, S. Santesson, S. Farrell, S. Boeyen, R. Housley, and W. Polk. Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile. RFC 5280, IETF, May 2008.

[35] Danny Cooper, Ethan Heilman, Kyle Brogle, Leonid Reyzin, and Sharon Goldberg. On the Risk of Misbehaving RPKI Authorities. In *Proceedings of the Twelfth ACM Workshop on Hot Topics in Networks*, HotNets-XII, pages 16:1–16:7, New York, NY, USA, 2013. ACM.

[36] X. Dimitropoulos, D. Krioukov, M. Fomenkov, B. Huffaker, Y. Hyun, k. claffy, and G. Riley. AS Relationships: Inference and Validation. *ACM SIGCOMM Computer Communication Review (CCR)*, 37(1):29–40, Jan 2007.

[37] Aaron Falk. The ietf, the irtf, and the networking research community. *SIGCOMM Comput. Commun. Rev.*, 35(5):69–70, Oct. 2005.

[38] Lixin Gao and Jennifer Rexford. Stable Internet Routing Without Global Coordination. *IEEE/ACM Trans. Netw.*, 9(6):681–692, December 2001.

[39] Yossi Gilad, Avichai Cohen, Amir Herzberg, Michael Schapira, and Haya Shulman. Are we there yet? on rpki's deployment and security. In *Proc. of NDSS*. ISOC, 2017.

[40] V. Giotsas, A. Dhamdhere, and k. claffy. Periscope: Unifying Looking Glass Querying. In *Passive and Active Network Measurement Workshop (PAM)*, Mar 2016.

[41] Vasileios Giotsas, Matthew Luckie, Bradley Huffaker, and kc claffy. Inferring Complex AS Relationships. In *Proc. of ACM IMC*, pages 23–30, New York, NY, USA, 2014. ACM.

[42] Sharon Goldberg. Why is It Taking So Long to Secure Internet Routing? *Commun. ACM*, 57(10):56–63, September 2014.

[43] John Hawkinson and Tony Bates. Guidelines for creation, selection, and registration of an Autonomous System (AS). RFC 1930, IETF, March 1996.

[44] Ethan Heilman, Danny Cooper, Leonid Reyzin, and Sharon Goldberg. From the Consent of the Routed: Improving the Transparency of the RPKI. In *Proceedings of the 2014 ACM Conference on SIGCOMM*, SIGCOMM '14, pages 51–62, New York, NY, USA, 2014. ACM.

[45] G. Huston, G. Michaelson, and R. Loomans. A Profile for X.509 PKIX Resource Certificates. RFC 6487, IETF, February 2012.

[46] Y. Hyun, A. Broido, and k. claffy. Traceroute and BGP AS Path Incongruities. Tech-

nical report, Cooperative Association for Internet Data Analysis (CAIDA), Mar 2003.

[47] Daniele Iamartino, Cristel Pelsser, and Randy Bush. Measuring BGP Route Origin Registration and Validation. In *Passive and Active Measurement - 16th International Conference, PAM 2015, New York, NY, USA, March 19-20, 2015, Proceedings*, pages 28–40, 2015.

[48] Akmal Khan, Taekyoung Kwon, Hyun-chul Kim, and Yanghee Choi. AS-level Topology Collection Through Looking Glass Servers. In *Proceedings of the 2013 Conference on Internet Measurement Conference*, IMC '13, pages 235–242, New York, NY, USA, 2013. ACM.

[49] Craig Labovitz, Abha Ahuja, Abhijit Bose, and Farnam Jahanian. Delayed Internet Routing Convergence. In *SIGCOMM '00: Proceedings of the conference on Applications, Technologies, Architectures, and Protocols for Computer Communication*, pages 175–187, New York, NY, USA, 2000. ACM.

[50] M. Lepinski and K. Sriram. BGPsec Protocol Specification. RFC 8205, IETF, September 2017.

[51] Matthew Luckie, Bradley Huffaker, Amogh Dhamdhere, Vasileios Giotsas, and kc claffy. AS Relationships, Customer Cones, and Validation. In *Conference on Internet Measurement Conference*, IMC'13, pages 243–256, New York, NY, USA, 2013. ACM.

[52] Robert Lychev, Sharon Goldberg, and Michael Schapira. BGP Security in Partial Deployment: Is the Juice Worth the Squeeze? In *Proceedings of the ACM SIGCOMM 2013 Conference on SIGCOMM*, SIGCOMM '13, pages 171–182, New York, NY, USA, 2013. ACM.

[53] Sebastian Meiling. RTRLIB Usage: Examples. `https://github.com/rtrlib/rtrlib/wiki/Examples`. Accessed: 2017-07-30.

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

[55] RIPE NCC. RIPE Atlas. `https://atlas.ripe.net/`. Accessed: 2017-06-28.

[56] RIPE NCC. Router Configuration for Resource Certification (RPKI). `https://www.ripe.net/manage-ips-and-asns/resource-management/certification/router-configuration`. Accessed: 2017-07-30.

[57] Ricardo Oliveira, Dan Pei, Walter Willinger, Beichuan Zhang, and Lixia Zhang. The (in)Completeness of the Observed Internet AS-level Structure. *IEEE/ACM Trans. Netw.*, 18(1):109–122, February 2010.

[58] Ricardo V. Oliveira, Dan Pei, Walter Willinger, Beichuan Zhang, and Lixia Zhang. In Search of the Elusive Ground Truth: The Internet's As-level Connectivity Structure. In *Proceedings of the 2008 ACM SIGMETRICS International Conference on Measurement and Modeling of Computer Systems*, SIGMETRICS '08, pages 217–228, New York, NY, USA, 2008. ACM.

[59] C. Orsini, A. King, D. Giordano, V. Giotsas, and A. Dainotti. BGPStream: a software framework for live and historical BGP data analysis. In *Internet Measurement Conference (IMC)*, Nov 2016.

[60] Jon Postel. Internet Protocol. RFC 791, IETF, September 1981.

[61] Y. Rekhter, T. Li, and S. Hares. A Border Gateway Protocol 4 (BGP-4). RFC 4271, IETF, January 2006.

[62] Andreas Reuter, Matthias Wählisch, and Thomas C. Schmidt. RPKI MIRO: Monitoring and Inspection of RPKI Objects. In *Proc. of ACM SIGCOMM, Demo Session*, pages 107–108, New York, August 2015. ACM.

[63] Matthew Roughan, Walter Willinger, Olaf Maennel, Debbie Perouli, and Randy Bush. 10 Lessons from 10 Years of Measuring and Modeling the Internet's Autonomous Systems. *IEEE Journal on Selected Areas in Communications*, 29(9):1810–1821, 2011.

[64] Quirin Scheitle, Matthias Wählisch, Oliver Gasser, Thomas C. Schmidt, and Georg Carle. Towards an Ecosystem for Reproducible Research in Computer Networking. In *Proc. of ACM SIGCOMM Reproducibility Workshop*, New York, NY, USA, August 2017. ACM.

[65] Brandon Schlinker, Kyriakos Zarifis, Italo Cunha, Nick Feamster, and Ethan Katz-Bassett. PEERING: An AS for Us. In *Proceedings of the 13th ACM Workshop on Hot Topics in Networks*, HotNets-XIII, pages 18:1–18:7, New York, NY, USA, 2014. ACM.

[66] J. Scudder, R. Fernando, and S. Stuart. BGP Monitoring Protocol (BMP). RFC 7854, IETF, June 2016.

[67] Lakshminarayanan Subramanian, Sharad Agarwal, Jennifer Rexford, and Randy H. Katz. Characterizing the Internet Hierarchy from Multiple Vantage Points. Technical report, Berkeley, CA, USA, 2001.

[68] Matthias Wählisch, Fabian Holler, Thomas C. Schmidt, and Jochen H. Schiller. RTRlib: An Open-Source Library in C for RPKI-based Prefix Origin Validation. In *Proc. of USENIX Security Workshop CSET'13*, Berkeley, CA, USA, 2013. USENIX Assoc.

[69] Matthias Wählisch, Olaf Maennel, and Thomas C. Schmidt. Towards Detecting BGP Route Hijacking using the RPKI. In *Proc. of ACM SIGCOMM, Poster Session*, pages 103–104, New York, August 2012. ACM.

[70] Matthias Wählisch, Robert Schmidt, Thomas C. Schmidt, Olaf Maennel, Steve Uhlig, and Gareth Tyson. RiPKI: The Tragic Story of RPKI Deployment in the Web Ecosystem. In *Proc. of 14th ACM Workshop on Hot Topics in Networks (HotNets)*, pages 11:1–11:7, New York, Nov. 2015. ACM.

[71] Matthias Wählisch and Thomas C. Schmidt. See How ISPs Care: An RPKI Validation Extension for Web Browsers. In *Proc. of ACM SIGCOMM, Demo Session*, pages 115–116, New York, August 2015. ACM.

[72] Matthias Wählisch, Thomas C. Schmidt, Markus de Brün, and Thomas Häberlen. Exposing a Nation-Centric View on the German Internet – A Change in Perspective on the AS Level. In *13th Passive and Active Measurement Conference (PAM)*, volume 7192 of *LNCS*, pages 200–210, Berlin Heidelberg, 2012. Springer-Verlag.

[73] Walter Willinger and Matthew Roughan. Internet Topology Research Redux. In Hamed Haddadi and Olivier Bonaventure, editors, *Recent Advances in Networking*, pages 1–59. Licensed under a CC-BY-SA Creative Commons license, 2013.