Working Around BGP: An Incremental Approach to Improving Security and Accuracy of Interdomain Routing

Geoffrey Goodell, William Aiello, Timothy Griffin,
John Ioannidis, Patrick McDaniel, Aviel Rubin
goodell@eecs.harvard.edu, {aiello, griffin, ji, pdmcdan, rubin}@research.att.com

Abstract

BGP is essential to the operation of the Internet, but is vulnerable to both accidental failures and malicious attacks. We propose a new protocol that works in concert with BGP, which Autonomous Systems will use to help detect and mitigate accidentally or maliciously introduced faulty routing information. The protocol differs from previous efforts at securing BGP in that it is receiver-driven, meaning that there is a mechanism for recipients of BGP UPDATE messages to corroborate the information they receive and to provide feedback. We argue that our new protocol can be adopted incrementally, and we show that there is incentive for network operators to do so. We also describe our prototype implementation.

1. Introduction

There are tens of routing protocols; they can be broadly split into two categories: *intra*domain, or internal, routing protocols, and *inter*domain, or external, routing protocols. Organizations under cohesive administrative control (companies, universities, Internet service providers) use intradomain routing protocols to exchange information about how to reach machines within their own purview. Interdomain routing protocols are used to exchange and propagate reachability information *between* such organizations. This split reflects the coarse structure of the Internet: many networks connected to each other. It also reflects the different needs and requirements for routing protocols for use in intra- versus interdomain routing. While there are several internal routing protocols in use today, there is only one interdomain routing protocol: the Border Gateway Protocol (BGP) [18, 20].

BGP views the Internet as a collection of interconnected *Autonomous Systems*. An Autonomous System (AS) is a portion of the network under single administrative control (at least as far as routing is concerned). Each AS connects to other ASes; the routers in each AS that connect to their counterpart in other ASes are called *border* routers. These neighboring border routers connect *directly* to each other, that is, there are no routers between them. (This is not strictly true, nor is the assertion that only neighboring routers speak BGP to each other, but the details are beyond the scope of this paper.) Over this direct connection, border routers establish *BGP sessions*; there may be many BGP sessions over each link, but there are (almost) never BGP sessions between non-neighboring routers. BGP sessions are used to exchange network reachability information — each router tells its neighbor what address ranges (also known as address prefixes, or just prefixes) it knows how to route to, along with ancillary information that is used to make the decision of whether this router will actually be used to route that part of the address space.

As BGP provides information for controlling the flow of packets between ASes, the protocol plays a critical role in Internet efficiency, reliability, and security. The Internet can be severely impacted by BGP failures. Accidental misconfigurations have resulted in serious routing problems and loss of service [13]. However, failures are not always accidental — attacks intended to cause widespread outage on the Internet will (and do) target BGP [16, 19]. Denial of service is not the only concern; an attacker might redirect the flow of some traffic through his network so that he can eavesdrop on it.

BGP has several well-known vulnerabilities. Neither the originating announcement of a route, nor the information attached to it as it traverses ASes are guaranteed to be correct. Moreover, BGP does not provide any way of identifying the source of bad data. Hence, misconfigured or malicious routers can, among others things, force other ASes to accept bad or inefficient routes, hijack address ranges, or simply flood the network with useless route information.

The security limitations of BGP are compounded by the fact that the protocol itself does not always converge [22]. Because BGP is potentially unstable at any time, it is particularly difficult to analyze. Complexity is always at odds with security. Getting the routing system to work at an acceptable level has taken huge effort in terms of design-

ing, implementing, and deploying protocols. Moreover, as the nature of the Internet changes, these protocols have been required to provide functionality not originally envisioned. It comes as no surprise that security has not been the first priority of designers, implementers, or even operators; it is this lack of security that makes the routing system, and hence the entire Internet, susceptible to an increasing number of both accidental failures and malicious attacks.

In this paper, we present the Interdomain Route Validation (IRV) service, a new protocol that acts as a companion to BGP. IRV defines a service that protects against rogue, subverted, or grossly misconfigured ASes, and is used to identify and diagnose routing configuration problems. We have designed IRV as a separate protocol because of the difficulty in changing widely deployed protocols such as BGP. This design allows fast and minimally disruptive deployment. For similar reasons, IRV is meant to be incrementally deployable, and we argue that even small groups of ASes will see immediate benefit from deployment. Moreover, such deployment does not interfere with the operation of non-participants. We describe the uses of IRV through real world examples, and consider its use as a replacement for and in conjunction with other routing services and protocol extensions. We have implemented a prototype, which we also describe.

2. Related Work

Murphy [16] outlines and categorizes many of the security vulnerabilities present in BGP and in the infrastructure used to propagate the route announcements between ASes. She describes a threat model that includes not only outsiders, but misconfigured and malicious routers as well. In a related document [15], Murphy characterizes some solutions needed to rectify many of the most significant vulnerabilities. Some of the potential attacks on BGP, such as simple replay attacks and denial of service attacks that involve shutting down BGP sessions prematurely, can be solved by securing the channels between BGP speakers. For this class of vulnerabilities, Murphy recommends use of IPsec between BGP routers. Furthermore, Murphy recommends that originator information be authenticated via digital signature (*e.g.*, signed association between origin and prefix).[1] Finally, Murphy suggests that each AS sign the Autonomous System Number (ASN)[2] of the next AS in the path (thus authenticating the complete AS path).

The Secure Border Gateway Protocol (S-BGP) [12] addresses many of the issues presented in Murphy's work. To protect the actual BGP sessions, S-BGP uses IPsec [10]. For the fundamental routing vulnerabilities, S-BGP introduces the concept of *attestations*, which are digitally signed statements used to verify the authenticity of route announcements. *Address attestations* are statements signed by a well-known authority that map an ASN to a prefix or prefixes, verifying that the speaker who originated the route announcement was eligible to do so for the indicated prefixes. *Route attestations* are statements signed by an AS that list the next AS in the path; they are used by each AS in the path to verify that the following AS along the path legitimately received the announcement and the privilege to forward it. When used together, these two forms of attestations create a well-defined chain of evidence for most route announcements.

Despite its many advantages, S-BGP has not been widely deployed among autonomous systems on the Internet. Reasons for this may include factors such as the computational cost of sending the larger and more complex *UPDATE* messages, as discussed by the authors of [11], not to mention concomitant costs of upgrading existing routing firmware. Also, implementation of S-BGP requires fundamental changes to BGP itself, which means that routers along the path from the source of an announcement to the destination need the ability to forward S-BGP messages. Also, in order to achieve the benefit of route attestations, all ASes in the path between the announcer of an *UPDATE* message and a given recipient must run an implementation of S-BGP.

Huston [9] argues that BGP may already be too *monolithic* a protocol in that it simultaneously performs multiple distinct functions — exchanging reachable prefixes, learning about (local) topology, binding prefixes to paths, and implementing routing policy. He argues that interdomain routing might be more scalable if these functions where performed by separate protocols. We would note that adding security and authentication to BGP, as S-BGP does, only increases complexity of the protocol and will likely diminish its scalability in the long run.

The Internet Routing Registry (IRR) [3] provides a non-invasive alternative strategy to ensuring reasonable routes by providing a set of routing policy databases. The IRR model introduces a third party capable of collecting and publishing AS-specific policy information. This approach to ensuring route validity is a response to the need described in RFC 1787 [17] for improving global consistency by sharing policy data among providers. Each participating AS submits policy data, encoded using the Routing Policy Specification Language (RPSL) [5, 14]. Interested parties may contact the registry to determine the stated policies for a particular AS, including what ASes (and possibly prefixes) are suitable for import or export.

[1] This presumes that there exists some infrastructure for mapping ASes to prefixes, and that an appropriate authority can create and distribute these statements.

[2] Each AS is assigned a (generally) unique *AS Number* by the authorities that govern Internet addressing (*e.g.*, ARIN [1], RIPE [4], APNIC [2]).

Additional information provided to the IRR by an AS often includes policy concerning the configuration of BGP communities; this information is generally most useful to neighboring ASes.

In practice, the IRR contains information from a substantial number of ASes, and it serves a useful purpose in debugging policy-related errors. However, the utility of the IRR for securing routing is quite limited. First, the IRR does not provide information about current routes, but only about potential routes. Some potential routes may be legal according to the IRR, but undesirable from a more global point of view. Next, the IRR has many security vulnerabilities concerning the integrity of registry contents and authorization of changes to the registry. Moreover, some policy information concerning agreements between peering ASes is sensitive and not to be shared with everybody; such information will therefore not be in the IRR. Some of these concerns are addressed in a proposal to make the IRR more secure [23], but until problems of authorization of database queries are addressed, the IRR will not be useful in conveying policy data other than that which are safely world-readable. In addition, the nature of the IRR provides ASes with little incentive to keep their own records in the policy databases up-to-date, further reducing its usefulness.

Mahajan *et al.* [13] argue that misconfigurations are responsible for a substantial portion of the errors and instability that plague interdomain routing. They describe several different forms of misconfiguration that result in the unintentional advertisement of incorrect prefixes, and they describe some of the reasons for these incidents, many of which are logistical and managerial in nature rather than technical.

3. IRV Architecture

Existing BGP security approaches have not been widely deployed; the reasons include limited ability to be incrementally deployed, high computational costs, and the infeasibility of modifying the vast installed base of BGP implementations. Recognizing these limitations, we propose the Interdomain Routing Validation (IRV) architecture. Used in conjunction with BGP, IRV is used to validate BGP data and acquire additional routing information relevant to an AS. IRV has the following goals:

- Allow ASes to acquire and validate both static (*e.g.*, policy) and dynamic (*e.g.*, current route advertisements) interdomain routing information.
- Be incrementally deployable; the system must provide substantial benefit even with limited adoption.
- Allow ASes to securely differentiate the requesters of routing information, in order that responses be tailored to the recipient.
- Not be tightly coupled with BGP; the protocol must operate independently of the reception of BGP messages, and ASes must be free to validate and acquire routing information whenever they desire.
- Allow ASes to passively receive routing-relevant information from remote entities; this will permit collections of participating ASes to cooperatively monitor and debug the routing infrastructure.

3.1. Approach

IRV combines features of S-BGP and the Internet Routing Registry. Like S-BGP, IRV allows autonomous systems to confirm (attest) that they announced or propagated particular routes. Unlike S-BGP, validation information is not carried in *UPDATE* messages. Instead, we introduce the notion of an Interdomain Routing Validator (IRV). Each participating AS designates an IRV responsible for answering queries from other ASes. Users of the system query the IRV to validate received BGP data or to acquire additional route-relevant information. IPsec or TLS can be used to ensure the integrity, authenticity, and timeliness of the queries and responses. Figure 1 illustrates this; when the Network Management Element (NME) in AS3 wants to verify an *UPDATE* message concerning AS1 that AS3 got from one of its neighbors, it queries the IRV located in AS1.

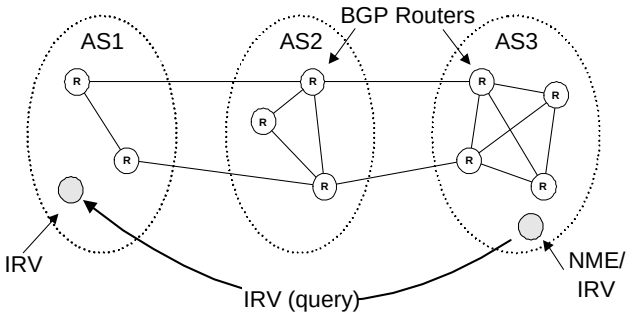

Figure 1. The IRV Protocol – BGP messages are validated and additional routing information is acquired (generally by another IRV or network management element – NME) via querying the IRV representing the relevant AS.

The essence of the IRV architecture is a decentralized query system. Participating ASes establish an IRV to speak authoritatively (through query interfaces) about the local network status and configuration. The IRV provides access to static and dynamic data through a query interface. Static data can include routing policy (such as that currently exchanged using RPSL and centralized registries), community information, or data relating to peer-

ing. The dynamic data can include received BGP route announcements, the current routing tables, or a description of advertised routes. Queries are used to implement features not currently present in BGP.

3.2. The IRV

Each IRV is a server, deployed as a dedicated machine (or set of machines) within the AS that it represents. However, this need not be the case. The only requirement is that those that wish to contact the IRV corresponding to a particular AS can do so (we consider the design of a distributed service that maps ASNs to IRVs in Section 3.4). Regardless of how the IRV is contacted, it must provide an interface by which external entities can query routing data.

The IRV architecture does not mandate the kinds of data supported by an IRV. ASes are free to include or omit support for any routing information. By design, ASes can extend the query/repository schema easily (see Section 4). We expect that different schemata will evolve within the communities in which IRV is deployed. The following includes data that are likely to be supported by IRVs:

- RPSL records indicating import or export policies for an AS. This can simply duplicate the format and semantics of what is provided by the Internet Routing Registry.
- Configuration information for BGP communities. Presently, much of this information is distributed via RPSL *remarks* fields stored in the Internet Routing Registry. For example, ASes use *remarks* fields to indicate that a particular *UPDATE* should not be propagated, or that an *AS_PATH* should be padded. These (common) uses of BGP communities could be standardized and interfaces simplified through schema extensions.
- Contact information for the maintainers of an AS. The contact information found in the Internet Routing Registry and *whois* database is often out of date, incorrect, or missing entirely.
- Received route advertisements and withdrawals. This feature exports an AS-centric view of the routing infrastructure. This view may be extremely useful in debugging routing inconsistencies and detecting failures.
- Route advertisements sent to neighbor ASes. These records serve as route attestations: the IRV states the set of *UPDATE* messages that it is currently using to advertise routes (and to whom they are sent).
- Sensitive or recipient-specific information that is not appropriate to broadcast in a BGP *UPDATE* message. For example, an AS may want to restrict access to data concerning private peering relationships.

Note that the IRV need not respond uniformly to all requesters; an IRV may be configured to restrict access to particular data to a list of authorized requesters. While we do not mandate that the IRV authenticate the source of queries, we view authentication as essential to future routing environments. The BGP model requires that an *UPDATE* message be propagated virtually unmodified as it traverses routers. As a result, the originator of a route advertisement has little control over the set of entities that have access to the advertisement. Our work is partly motivated by the fact that BGP alone cannot distinguish between recipients. By offering a second stage in which recipients of an *UPDATE* can request additional information, we provide the ability for an announcer of a route to provide discretionary information to authorized recipients.

3.3. Using IRV

The main use for IRV is as a way of validating BGP data. Origin information can be (naïvely) validated by querying the origin AS identified in each *UPDATE* message at the time it is received. However, the costs of validation can be amortized by queuing sets of *UPDATE* messages originating from a single AS, which can later be validated in a single bulk validation query. We expect that the decision to query a downstream IRV will be based on the disparity between the *UPDATE* and a baseline mapping of the address space and known AS connectivity.

The authenticity of a received *AS_PATH* by can be verified by querying each node in the path. However, it may be beneficial to cache previously acquired policy and route information. These cached values can be used to avoid revalidation of stable information (*e.g.*, origin information) associated with frequently changing routes

IRV does not mandate when queries are sent. However, the algorithm chosen by an AS will determine the cost associated with validation. A significant benefit of IRV is that parties do not need validation information with each *UPDATE*. For example, an AS may choose to query routes at random intervals, which may reduce local load. Another approach might be to vary the frequency of queries by ASN. ASes that are topologically closer or deemed more relevant may be queried more frequently. Querying other ASes based upon a random sampling at periodic intervals may be effective at identifying problems.

It is possible to use IRV to only provide static information such as routing policy. In many respects, this is functionally equivalent to using the IRR. However, since the IRV and the requesters communicate directly, the former can still tailor its responses to the latter.

3.4. Finding the IRV for an AS

An important consideration in the IRV architecture is the way by which the IRV associated with each AS is located. An obvious approach would embed a *hint* address of each AS's IRV within *UPDATE* messages. The address would be authenticated during subsequent communication with the IRV, *e.g.*, via known certificate. Because this design would require modification to existing BGP implementations, we view it as highly undesirable. An alternative approach would institute a well-known registry to store and distribute authoritative IRV contact information. This IRV contact registry need only store IRV location information (*e.g.*, IP addresses) for each AS.[3] The implementation of the registry itself need not be complex. For example, the central registry may use HTTP redirection or DNS records to communicate IRV location.

While the preceding sections have implied that each AS has a globally unique ASN, this is not always true. RFC 2270[21] specifies how a set of singly-homed ASes (with the same upstream provider) can share a single ASN.[4] Having a unique IRV associated with each AS is essential for certain uses of the IRV architecture (*e.g.*, origin validation). Hence, there must exist a way of disambiguating requests to IRVs for data associated with RFC 2270 prefixes. We propose to use the provider's IRV to redirect these requests to the IRV of the appropriate AS. Determining that an AS specified in an *AS_PATH* actually refers to an RFC 2270 AS is non-trivial. RFC 2270 specifies that the ASN used for a singly-homed AS shall be either a number previously assigned to its provider or a private ASN (64512–65535). When private ASNs are used, the ASN is stripped from the *AS_PATH* by the provider, so the requester might (incorrectly) treat the provider's IRV as authoritative. In either case, the provider may configure its IRV to require the requester to specify a prefix along with the ASN when making queries. Alternatively, an IRV could use the **aslocator** field (described in Section 4.1) to return a pointer to the location of the proper IRV.

3.5. Authentication and Secure Communication

While IRV does not require a security infrastructure, one is essential in countering the threats against interdomain routing. When necessary, the IRV must authenticate queries to prevent unauthorized access to sensitive data (*i.e.*, enforce access control over routing information). Responses must be authenticated to prevent forgery. Confidentiality may also be a concern. For example, peering relationships are often closely guarded secrets. In this case, peers may desire confidentiality to prevent exposure of these relationships. Confidential communication may enable new kinds of AS interaction: the parties are now free to share changing data deemed more sensitive than what can be advertised publicly. Of course, where the queried information is not deemed sensitive, authentication and confidentiality are optional.

Queries and responses can be authenticated using digital signatures. This approach requires some means of distributing public keys, and may consume significant computational resources [11]. Note that these costs can be reduced by caching and later reusing frequently used requests and responses. Caches need to be carefully designed, as incorrect implementations may introduce vulnerabilities to replay attack; timeliness bounds must be established and enforced on cached requests and responses.

The computational costs associated with digital signatures can place a significant burden on the already resource-limited routers. Unlike BGP or the IRR, the validating entity in IRV is in direct communication with a representative of the relevant AS. Hence, we recommend that existing security protocols (*e.g.*, IPsec or TLS) be used to establish long term security associations. These associations are maintained over potentially long periods during which many requests and responses are exchanged.

3.6. Extending IRV Queries

As described in the preceding sections, IRV provides a clear benefit to receivers of routing information. However, it also presents an opportunity to benefit the provider of that information. Users (requesters) can actively supply information to ASes about themselves through query interfaces. This information can be used to identify failures, monitor network connectivity, and improve the quality of the advertised routes.

We extend the notion of IRV queries to include submissions of routing *reports*. Reports are voluntary, and may include received announcements, connectivity data, changing policy, topology data, or any other information relating to interdomain routing. The provider IRV will decide how much and when report information is submitted. For example, reports can be used to flag BGP misconfigurations that result in bad route announcements.

Tools that permit ASes to automatically share network health and performance data are likely to improve connectivity. Hence, we expect network operators to view bidirectional sharing of routing information as beneficial. There is a great incentive to deploy IRV where best practices dictate information sharing. ASes that wish to profit from reports must provide an IRV. Hence, requesters supplying reports will also have the opportunity to make use of the other services provided by IRV. This balance works

[3] Clearly, location information must be authenticated. One or more location service certificates could be configured at each router for this purpose.

[4] RFC 2270 notes that additional software at the provider can be used to obviate the need for distinct ASNs across the provider's different singly-homed customers.

as an incentive to provide IRV support: ASes are likely to receive preferential treatment from those ASes to which they consistently provide routing data.

4. Implementation

IRV allows interested parties to query routing and policy information from participating ASes. For reasons described in the preceding sections, such a system must be simple, robust, and built on widely deployed technology. Because HTTP easily fulfills these requirements, our prototype implements IRV as a web-based service. Solutions that provide security to web-based services are well known and widely available (SSL/TLS and, to a lesser extent, IPsec). Hence, ASes are free to implement IRV security as is appropriate for their environment. Finally, the administrative costs of running an IRV web server are such that it will not serve as a deterrent to adoption. It goes without saying that a general-purpose web server should probably not be used because of the obvious security implications of very complex software.

The traditional way of expressing routing policy, RPSL, provides structures that describe import policies, export policies, forwarding defaults, route preferences, route flap dampening measures, and various other AS-specific policy declarations. While RPSL provides a starting point for our language, we seek a way of expressing queries and exchanging more generalized policy information. The potential uses of the protocol, combined with the need for extensibility, suggest that a flexible, convenient language for expressing routing-relevant information is required. For these reasons, we chose XML[7], largely for its modular structure and widespread deployment. We define an XML schema[8] to express classes of data (called *sections*, see next section) supported by an IRV. All data associated with an AS are stored in a collection of data objects confirming to this schema in an AS-local IRV database.

We have implemented a prototype IRV and its accompanying front-end user interface. The interface allows users to make queries through an HTML form. User-supplied forms-data are submitted to the web-server through the HTTP POST method. The web server passes the query data to a CGI script that queries the IRV database. Query results are returned as an XML document. Users specify the categories of data to query through the HTML form. For example, RPSL data are queried by accessing the *policy* section of the database.

We have chosen the XQuery[6] language to encode the queries made to the IRV database. XQuery provides a convenient way of retrieving data from XML documents. The query schema is designed such that it is possible to make multiple queries within a single request. The structure of the query response ensures that returned data are unambiguous.

4.1. The Prototype IRV Schema

The schema supported by the IRV prototype roughly corresponds to the taxonomy presented in Section 3.2. This schema defines a set of independent *sections* representing broad categories of routing information. The following briefly describes each section:

- The **policy** section provides policy information as it appears in the Internet Routing Registry (IRR). Policy is presented as sets of RPSL attributes[5]. Our decision to use RPSL rather than XML to express the policy is one of compatibility: current tools that make use of policy registries require RPSL.
- The **config** section extends RPSL by standardizing BGP community information. This section assigns semantic meaning to various community values (*e.g.*, as is currently done through RPSL "remarks" attributes). Our implementation recognizes two commonly used, but nonstandard, BGP communities: "*no announce*" and "*prepend*". A "*no announce*" community indicates that a particular route is not to be announced to a specific peer. A "*prepend*" community indicates that a specified ASN should be prepended to the *AS_PATH*. A field indicating the number of times the ASN should be prepended is supplied.
- The **contactinfo** section provides human-readable text stating how the administrators of an AS can be contacted. This information may also include text describing procedures for reporting and tracking problems relating to the AS.
- The **aslocator** provides location information for the IRV of other ASes. The current prototype maps each ASN to the appropriate IRV through local configuration. Because "location" can be defined many ways (see Section 3.4), a "type" field is used to indicate how the location field is interpreted. Location information may include URLs, IP addresses, or any other data indicating how an IRV can be contacted. In addition, **aslocator** mappings can be further refined by prefix (rather than solely by ASN). This is used to resolve the IRV location where multiple ASes share an ASN (according to RFC 2270).
- The **bgproute** and **bgpreceived** sections form the structure for dynamic routing queries. **bgproute** information records the current route announcements and withdrawals made by an AS. **bgpreceived** information records the route announcements and withdrawals recently heard from neighboring ASes. Both record types contain BGP *UPDATE* messages. **bgproute** records associate each *UPDATE* message with a set of recipient ASes. **bgpreceived** records

associate each *UPDATE* message with the single AS from which it was received.

Note that our prototype schema is not intended to be fixed. Those ASes encountering new requirements can arbitrarily extend or modify any part of the schema. This extensibility allows ASes to explore new facilities and services within the existing interdomain routing infrastructure (*e.g.*, interdomain load balancing, traffic engineering). As it exists today, BGP does not allow this exploration. We expect that the schemata used in real networks will reflect the needs of those communities which deploy IRV, and evolve as those needs change.

5. Examples

This section illustrates how IRV is used to mitigate interdomain routing failures through four representative examples. The first two examples discuss failures resulting from misconfiguration, and the latter two discuss failures resulting from malicious behavior (attacks). We begin by considering the most common failures resulting from BGP misconfiguration (as identified by Mahajan *et al.* [13]).

1. Origin Misconfiguration. Origin misconfiguration occurs when an AS inadvertently inserts a prefix into the global BGP tables. Failures of this type can be classified as *self-deaggregation*, *related-origin propagation*, and *foreign-origin propagation*. In self-deaggregation, an AS announces prefixes that should have been aggregated but were not, unnecessarily advertising more-specific prefixes). Related-origin failures occur when prefixes that should remain local are propagated. Foreign-origin propagation occurs when an AS claims ownership of some prefix that it does not own. These misconfigurations often result from configuration errors, poor router synchronization, application of incorrect route attributes, or unwarranted reliance on an upstream AS filtering.

One way an AS can detect related and foreign origin misconfigurations is by requiring proof of ownership from the appropriate IRV. These proofs may consist of digitally signed statements binding prefixes to ASes. These statements would be issued by the appropriate prefix authority (*e.g.*, ARIN, RIPE, or APNIC). The developers of S-BGP have explored various architectures that provide these proofs [11].

By design, external observers have incomplete information of how an AS manages prefixes and routes. For this reason, detecting and preventing self-deaggregation is very difficult. We currently do not provide a specific mechanism that addresses these failures, but expect that IRV supplied information stating maximal permissible deaggregation (*i.e.*, prefix handling policy) will aid in detecting these failures.

2. Export Misconfiguration. Failures resulting from export misconfigurations occur when a route that violates policy is propagated [13]. These misconfigurations can result in sub-optimal routes or violation of AS agreements. Routing registries are designed to reduce the incidence of errors resulting from export misconfiguration: ASes use registries to verify that *UPDATE* messages are consistent with the policy of the AS from which they are received.

IRV mitigates export misconfiguration by acting as an AS-local route registry. For example, assume that an AS receives a route that contains AS202. The receiving AS can retrieve the RPSL policy from the IRV associated with ASN 202 (after locating it using **aslocator** interface of some known IRV). The resulting RPSL records would be used to validate the route information using existing verification procedures [5]. The following XQuery expression is used to acquire the relevant policy from AS202:

```
for $f in $doc/framework
return
  <framework> {
    for $p in $f/policy
    return
      <policy> {
        for $r in $p/rpsl
        return $r
      } </policy>
  } </framework>
```

Note that it may be advantageous to delay multiple XQuery requests sent to an IRV. A single request can be issued once a threshold of queries has been reached. Hence, request costs can be amortized over potentially many queries. For similar reasons, it may be advantageous to cache the results of queries (and serve future requests from that cache).

3. Announcement forgery. Route announcements are potentially sensitive. ASes along the propagation path of a BGP *UPDATE* message are able to modify advertised and withdrawn routes and their corresponding path attributes. ASes along the path have no means of determining that these modifications have taken place.

Announcement forgeries can be detected by obtaining direct verification from the origin AS. For example, suppose that AS301 receives an announcement for the prefix *12.244.0.0/16*, which originated from AS302. AS301 obtains verification by requesting all announcements associated with that prefix from AS302. This request is communicated in the following XQuery expression:

```
for $f in $doc/framework
return
  <framework> {
    for $p in $f/bgproute
    return $p[update[equal(nlri,
        "12.244.0.0/16")]]
  } </framework>
```

If AS302 did not intend to send any announcements

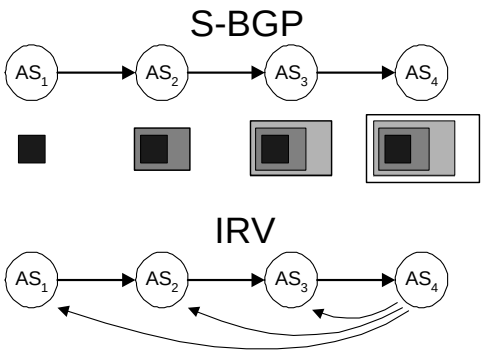

Figure 2. Verification of successor information using S-BGP and IRV

for this prefix, then the response contains no records, and AS301 can safely ignore the announcement. Otherwise, AS302 sends a record of the *UPDATE* messages that it is currently forwarding to its neighbors. Note that this solution is only secure if communication between the ASes is authenticated (*e.g.,* digitally signed responses, or by authenticated transport such as SSL/TLS or IPsec).

4. Forged propagation. Even if the BGP origin information were unforgeable, a malicious AS would still be able to alter the advertised *AS_PATH* path. Malicious entities may alter the path to increase or decrease the desirability of a route, to assert control over the flow of traffic, or simply to deny service.

Users of IRV can detect forged AS paths by seeking confirmation from each AS along the path. A query similar to the one given in the previous example is sent to the IRV of each AS along the path. The requesting AS verifies that the ASN of the next AS in the *AS_PATH* has a **recipient** entry in the appropriate **bgproute** record.

Illustrated in Figure 2, IRV path verification is semantically equivalent to S-BGP *route attestations*. However, paths are validated as needed via direct communication with the issuing AS, rather than from data carried in each BGP *UPDATE*. Hence, not only does IRV allow the recipient to decide when to perform verification, but also ensures that an AS can achieve partial benefit from partial adoption.

6. Discussion

It is important to consider how IRV fits in the universe of tools used to support Internet domain routing. In particular, one must assess how IRV relates to other techniques for interdomain policy and security services. The following illustrates the similarities and differences between this work and related BGP security proposals by comparing IRV with S-BGP. A discussion of more general policy distribution is also presented here.

The central goal of S-BGP is to support the validation of the crucial data upon which interdomain routing is based: path properties (route information) and prefix ownership (origin information). Route advertisements in S-BGP provide authenticating information using signatures. The AS signature over the route contents commits the AS to the *UPDATE*. The use of the signature, and indirectly the supporting PKI, prevents forgery. However, the costs associated with generating, distributing, and validating signatures for *UPDATE* messages can be prohibitively high. IRV considers a different model in which the originating AS commits resources to validation only where the verifying AS requests confirmation. Because such interaction is session-oriented and may represent long term associations, validation may be amortized over many *UPDATE* messages and may use low cost symmetric cryptography (over a long-term IPsec security association or over a persistent TLS session). However, such services must be prepared to support a potentially large community of users.

As proposed by S-BGP, cryptographically supported validation of prefix ownership requires the existence of a governing body (*e.g.,* ICANN as certificate issuer). Where available, IRV can make use of validating block governance by advertising ownership-proving credentials through the query interfaces, although some infrastructure is necessary. However, existing governing bodies do not provide such validation infrastructure, and the technical challenges to doing so indicate that deployment in the near term is highly unlikely. This is further complicated by incomplete knowledge of address ownership. Hence, in the near term, flagging and investigating inconsistencies, rather than validating ownership, may provide the best means of ensuring correct origin information.

Note that while IRV cannot provide strong prefix ownership validation without a governing authority, it can mitigate route misconfiguration. Detection of a misconfigured BGP speaker (*UPDATE* for a prefix not belonging to the AS) is detected during the associated *UPDATE* validation. In this case, the ownership is cross-referenced with the data advertised at the administrative server. Of course, where both sources are misconfigured or the AS is malicious, the error would not be detected.

Routing Registries (RRs) provide access to the routing policies of participating ASes. RR consumers alter the routing behavior infrastructure based on the policy content. Note that such infrastructure is inherently egalitarian: every user of the RR has access to the same policy data. Uniform access is not always desirable. Peering relationships, identity and state of routing infrastructure, and AS connectivity are frequently considered highly sensitive. As a result, ASes only provide non-sensitive policy to their RR, reducing its usefulness.

Each AS provides is own IRV server. Hence, each AS may exert control over what, to whom, and how policy is distributed. For example, an AS may wish to expose more information to ASes carrying their traffic. Other ASes need not (and should not) be provided information about how the routing infrastructure supports this relationship. These facilities may open to door to new kinds of cooperative behavior between BGP neighbors.

6.1. Accessing Dynamic Data

Many of our arguments in the previous sections assume that IRV servers have access to dynamic data, and the question of how IRV servers gain access to dynamic data is an important one. In this section, we describe a readily deployable architecture that achieves this goal. Adoption of this particular approach is not required by our system, but we believe that it will provide the necessary functionality.

According to our specification, IRV servers have access to two kinds of dynamic routing information:

- Cached BGP *UPDATE* messages *received* from neighboring BGP speakers. Each message is associated with the particular AS and BGP speaker that most recently forwarded it.
- A set of currently valid *UPDATE* messages, as they are to be *sent* to neighboring BGP listeners, including both *UPDATE* messages originating locally and those to be forwarded. Each message is associated with a set of neighboring ASes to which that message is to be sent.

To access BGP *UPDATE* messages *received* from neighboring BGP speakers, an IRV server can simply establish I-BGP sessions with all the border routers of the AS; this way, all BGP messages received from peer ASes are also given to the IRV. Figure 3 illustrates this; dark lines are I-BGP sessions propagating routing information received from routers D, E, F, and G. The IRV is now able to determine, from the identity of its I-BGP peer, the corresponding AS and foreign BGP speaker that propagated the message to the local AS. In the event that a router (in this case, A) has BGP sessions with more than one other router (in this case, D and E), it may be necessary to configure the E-BGP listener locally to use private *community* fields or other path attributes to signal to the IRV the identity of the foreign speaker.

Clearly, malicious/compromised routers can manipulate AS-local IRV services by arbitrarily omitting, delaying, or modifying I-BGP messages. However, protection of an AS from its own routers is explicitly outside the domain of IRV. We assume that ASes will employ additional infrastructure to detect and disable faulty or compromised routers. Note that any AS-centric (rather than router centric) solution must contend with these same issues.

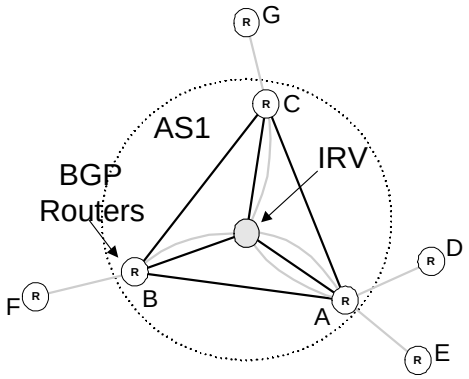

Figure 3. Dynamic Data acquisition by the IRV.

Maintaining current information regarding BGP *UPDATE* messages *sent* to foreign ASes is more challenging. In particular, there is no convenient way to intercept outbound E-BGP sessions, since they traverse a direct physical link between the E-BGP speaker and listener. For each outbound E-BGP session, we recommend configuring the corresponding border router to establish a second outbound E-BGP session with the IRV, configured to send the same data as the corresponding session with the remote AS. The gray lines in Figure 3 reflect this. As such, border routers that serve as BGP speakers in multiple BGP sessions establish multiple BGP sessions with the IRV. The IRV, then, is an E-BGP listener, configured to treat each E-BGP session to which it is privy to as authoritative with respect to the original outbound E-BGP session that it mirrors.

A related issue involves the question of who issues requests on behalf of an AS. It is possible for operators to query the IRV manually or via scheduled scripts that have no access to current routing data. However, in order to take full advantage of the system, an AS must be able to systematically form queries in response to received *UPDATE* messages. For this purpose, we recommend establishing an IRV client system, called the Network Management Element (NME) in Figure 1. The IRV client listens to I-BGP messages from E-BGP listeners at the border of the AS, and collects them for use in forming and sending useful queries. Since the IRV server already listens to I-BGP messages, it may make sense to collocate the IRV client and server on the same physical machine. However, this is not strictly necessary in order to achieve the desired results.

6.2. System Limitations

The fundamental limitation of any supervisory system is that it is only as good as the data in it. Since some of the data in IRV are configuration and policy data, which are maintained by human operators, there is always the chance that the data in the IRV are different from the configuration in the actual routers, even though the configuration may be correct. Unfortunately, many operators still configure routers by entering configuration commands at the console prompt; this is a source of many problems, as shown in [13]. To address this problem, a front end needs to be developed where new configurations are developed, then atomically transferred to both the IRV and the affected routers. Such a front end would, of course, be useful regardless of the adoption of IRV.

Part of the reason why operators enter configurations directly on the routers is that it is easy to do so, and the router configurations are ultimately the authoritative source of policy. Much of the data in the RRs is outdated, incomplete, or incorrect; operators derive no immediate benefit from updating the RRs, and thus fail to do so; it is just extra work. It might be argued that the same would hold true for IRV; why would an ISP go through the trouble of maintaining two syntactically different copies of its configuration? Would not that lead to the same divergence and uselessness of data as with the IRR? The answer hinges upon the distributed nature of IRV; whereas with RRs a central authority has to be updated, the information in IRV is created and managed by the originator of that information, namely, the network operator, and thus it is easier (and hence more likely) to keep the information correct.

Even with transactional semantics for the updating of the routers and the IRV, there will always be a short interval between the actual update and when the update propagates to the entire Internet (subject to policy filtering, of course). It is thus conceivable that a query from a remote AS could be initiated right after a change in the IRV was committed, but before the change propagated over BGP; the query concerns an older, and possibly just as valid, routing state of the network. Absent a lot of operator experience, it is not clear what the proper solution to this problem would be. The IRV could store some historical data — prior versions of its database — and be able to furnish them on demand, but this adds complexity to the protocol. On the positive side, the IRV itself could keep track of how frequently it gets requests or reports that point to inconsistencies, and deduce potential BGP propagation problems.

Another consideration that may delay the initial deployment of IRV is the so-called "network effect": the usefulness of a particular piece of technology being proportional to the number of people actually using it. Arguably, an operator deploying IRV in their own network does not initially gain much; a single IRV server running by itself is not very useful. Still, it can be used by just the AS deploying it in the following fashion: the ISP sets up multihop BGP sessions with routers of other ISPs, and uses the information thus gathered to perform sanity checks on itself. Whether this has any advantages over alternate ways of accomplishing the same goal is left for future work. Once more than just one AS start deploying IRV and using it, they can check each other's configurations. Again, there are a great many unknowns, such as deployment cost and value of the benefits to make a reasonable approximation of where the break-even point is.

7. Conclusions

BGP is the dominant protocol for interdomain routing, but current implementations of BGP provide little security. Emerging standards attempt to address this limitation by augmenting the existing protocol with security infrastructure. Such infrastructure frequently assumes universal deployment (within a vast collection of heterogeneous and often embedded software), requires significant computational resources, or provides limited ability to communicate policy. Exploitation of weaknesses within the present interdomain routing infrastructure could result in significant costs, financial or otherwise, to networks relying on external connectivity.

We have introduced the Interdomain Routing Validation (IRV) system. Used in conjunction with BGP, IRV provides interfaces through which BGP data can be validated and additional routing information can be acquired. Participating ASes designate an IRV that processes requests received from remote users. The requests consist of queries used to implement features not currently present in BGP. For example, an AS can validate an *UPDATE* message by querying the originating AS. Network security protocols are used to ensure the integrity, authenticity, and timeliness of the queries and responses.

Ultimately, the value of IRV is determined by its effectiveness in increasing an AS's ability to correctly obtain and manage interdomain routing information. Providing a common interface is a key means by which we achieve this goal. IRV is a receiver-driven architecture, providing the users of routing announcements with a role in obtaining the information they need to function correctly. The ability of IRV speakers to tailor responses to the requester affords greater control over how and to whom route information is shared. The ability for requesters to share information about received announcements with originators of those announcements provides originators with a degree of introspection by demonstrating how their announcements appear to the world.

The routing facilities supported by a AS are specific to

its administration. Hence, we view services such as IRV as a necessary and natural progression of interdomain routing. Each AS should provide data and interfaces tailored to its operational needs. Services such as IRV allow future enhancements to be quickly implemented, tested, and deployed within the interested communities.

The importance of incremental deployability to a system providing BGP security cannot be understated. Any new BGP feature is unlikely to receive quick or universal deployment. Hence, any solution should be of demonstrable value in the presence of partial adoption. This is true of IRV: the security and accuracy of the interdomain routing information is increased precisely within the community in which it is deployed.

The number and frequency of clients requesting information from the administrative server will be determined by the extent to which IRV is adopted throughout the Internet. In future work, we plan to systematically characterize and evaluate this cost. Such information will be used to design highly-scalable server implementations. We have yet to fully explore the potential uses of IRV. Other future plans we have include the extension of IRV schemata to other services, such as quality of service, load balancing, and congestion control. Centrally, this work will seek to use the IRV to communicate service-specific requirements between ASes, and ultimately influence interdomain routing.

Acknowledgements

We wish to thank Steve Bellovin, Matt Blaze, Howard Karloff, Fabian Monrose, and the staff at a230.com.

References

[1] American Registry of Internet Numbers. `http://www.arin.net/`.

[2] Asia Pacific Network Information Centre. `http://www.apnic.net/`.

[3] Internet Routing Registry. `http://www.irr.net/`.

[4] Réseaux IP Européens. `http://www.ripe.net/`.

[5] C. Alaettinoglu, C. Villamizar, E. Gerich, D. Kessens, D. Meyer, T. Bates, D. Karrenberg, and M. Terpstra. Routing Policy Specification Language (RPSL). *Internet Engineering Task Force*, June 1999. RFC 2622.

[6] S. Boag, D. Chamberlin, M. Fernandez, D. Florescu, J. Robie, J. Siméon, and M. Stefanescu. XQuery 1.0: An XML Query Language. W3C Working Draft, April 2002.

[7] T. Bray, J. Paoli, C. Sperberg-McQueen, and E. Maler. Extensible Markup Language (XML) 1.0, Second Edition. W3C Working Draft, October 2000.

[8] A. Brown, M. Fuchs, J. Robie, and P. Wadler. XML Schema: Formal Description. W3C Working Draft, September 2001.

[9] G. Huston. Scaling interdomain routing. *Internet Protocol Journal*, 4(4), Dec. 2001.

[10] S. Kent and R. Atkinson. Security architecture for the internet protocol. Request for Comments (Proposed Standard) 2401, Internet Engineering Task Force, November 1998.

[11] S. Kent, C. Lynn, J. Mikkelson, and K. Seo. Secure Border Gateway Protocol (S-BGP) — Real World Performance and Deployment Issues. In *Proceedings of Network and Distributed Systems Security 2000*. Internet Society, February 2000.

[12] S. Kent, C. Lynn, and K. Seo. Secure Border Gateway Protocol (Secure-BGP). *IEEE Journal on Selected Areas in Communications*, 18(4):582–592, April 2000.

[13] R. Mahajan, D. Wetherall, and T. Anderson. Understanding BGP Misconfiguration. In *Proceedings of ACM SIGCOMM '02*, pages 3–16. ACM, September 2002.

[14] D. Meyer, J. Schmitz, C. Orange, M. Prior, and C. Alaettinoglu. Using RPSL in Practice. *Internet Engineering Task Force*, August 1999. RFC 2650.

[15] S. Murphy. BGP Security Protections (*Draft*). Internet Research Task Force, February 2002. (`?????`).

[16] S. Murphy. BGP Security Vulnerabilities Analysis (*Draft*). Internet Research Task Force, February 2002. (`draft-murphy-bgp-vuln-00.txt`).

[17] Y. Rekhter. Routing in a Multi-provider Internet. *Internet Engineering Task Force*, April 1995. RFC 1787.

[18] Y. Rekhter and T. Li. A Border Gateway Protocol 4 (BGP 4). *Internet Engineering Task Force*, March 1995. RFC 1771.

[19] S. Staniford, V. Paxson, and N. Weaver. How to 0wn the Internet in Your Spare Time. In *Proceedings of the 11th USENIX Security Symposium*, August 2002.

[20] J. Stewart. *BGP4: Interdomain Routing in the Internet*. Addison-Wesley, 1998.

[21] J. Stewart, T. Bates, R. Chandra, and E. Chen. Using a Dedicated AS for Sites Homed to a Single Provider. *Internet Engineering Task Force*, January 1998. RFC 2270.

[22] K. Varadhan, R. Govindan, and D. Estrin. Persistent Route Oscillations in Inter-Domain Routing. *Computer Networks*, 32(1):1–16, 2000.

[23] C. Villamizar, C. Alaettinoglu, D. Meyer, and S. Murphy. Routing Policy System Security. *Internet Engineering Task Force*, December 1999. RFC 2725.