Jumpstarting BGP Security with Path-End Validation

Avichai Cohen
The Hebrew University of Jerusalem
avichai.cohen@mail.huji.ac.il

Yossi Gilad
Boston University and MIT
yossigi@mit.edu

Amir Herzberg
Bar-Ilan University
amir.herzberg@gmail.com

Michael Schapira
The Hebrew University of Jerusalem
schapiram@huji.ac.il

ABSTRACT

Extensive standardization and R&D efforts are dedicated to establishing secure interdomain routing. These efforts focus on two mechanisms: *origin authentication* with RPKI, and *path validation* with BGPsec. However, while RPKI is finally gaining traction, the adoption of BGPsec seems not even on the horizon due to inherent, possibly insurmountable, obstacles, including the need to replace today's routing infrastructure and meagre benefits in partial deployment. Consequently, secure interdomain routing remains a distant dream. We propose an easily deployable, modest extension to RPKI, called "path-end validation", which does not entail replacing/upgrading today's BGP routers. We show, through rigorous security analyses and extensive simulations on empirically derived datasets, that path-end validation yields significant benefits even in very limited partial adoption. We present an open-source, readily deployable prototype implementation of path-end validation.

1. INTRODUCTION

The Internet infrastructure was not designed with security in mind, and is consequently alarmingly vulnerable. We focus on the arguably most acute problem: *securing interdomain routing*, that is, routing between the administrative domains, or "Autonomous Systems" (ASes), which comprise the Internet. As highlighted by many high-profile configuration errors and attacks (e.g., [1, 2, 6]), the Border Gateway Protocol (BGP), today's de facto interdomain routing protocol, is hazardously insecure [10]. However, adoption of BGP security solutions is difficult and is proceeding slowly [21].

The current prevalent paradigm for securing interdomain routing, as advocated, for instance, by the IETF's Secure Inter-Domain Routing (SIDR) group, consists of two steps: (1) *origin authentication* with the Resource Public Key Infrastructure (RPKI) [30], followed by (2) *path validation* through replacing BGP with BGPsec [11], a security-enhanced interdomain routing protocol.

RPKI [30] certifies records binding an IP-prefix to the number and public key of its *origin AS*, i.e., the AS that "owns" that prefix. RPKI certificates allow BGP routers to perform *origin authentication* [36]: detect and discard *prefix hijacks*, BGP route advertisements where an IP prefix is announced by an AS that is not its legitimate owner. Prefix hijacks happen frequently (e.g., see [1, 2, 5, 9]), motivating the adoption of RPKI [37].

Origin authentication (via RPKI) provides an important first step towards securing interdomain routing, yet it is insufficient to prevent path-manipulation attacks. In particular, even with RPKI fully deployed, the attacker can still perform the *next-AS* attack, i.e., announce a fake link between himself and the victim AS. To address this and other path-manipulation attacks, the IETF is standardizing BGPsec [11], which uses digitally-signed BGP announcements. BGPsec prevents a BGP-speaking router from announcing a path that is not a legitimate extension of a valid path that is announced to it. To ensure this, BGPsec requires each AS to sign every path advertisement that it sends to another AS, and to validate all the signatures of previous ASes along the path. Unlike RPKI, integration of BGPsec necessitates changes to BGP routers [21]. Worse yet, recent work on adoption of BGP security [33] shows that in partial deployment, BGPsec is expected to achieve disappointingly meager security benefits over RPKI, while potentially even leading to *less* security and other undesirable phenomena (e.g., routing instabilities).

The above serious, arguably insurmountable, obstacles facing the adoption of BGPsec, beg the question: are there alternative security measures that are easier to deploy than full-fledged BGPsec, share the deployability advantages of RPKI (no need to replace routers, no online crypto, etc.), yet provide comparable security benefits to BGPsec? We argue below that this is indeed achievable via a modest extension to RPKI termed *path-end validation*.

SIGCOMM '16, August 22–26, 2016, Florianopolis, Brazil
 ISBN 978-1-4503-4193-6/16/08…$15.00
DOI: http://dx.doi.org/10.1145/2934872.2934883

Path-end validation has a much more modest goal than (full) path validation with BGPsec: it only attempts to ensure that the last AS hop on the advertised BGP path is valid, i.e., that the origin AS has approved reaching it via the previous AS along the path. What level of security can a check that merely validates that the 1-AS-hop suffix of a BGP path is valid provide? Our simulations show that even with relatively few adopters, path-end validation suffices to achieve a level of security that is close to the security guarantees of BGPsec in full deployment (before BGP is officially depracated, see [33]).

In retrospect, these surprisingly good news are easy to comprehend. An attacker cannot fool neighboring ASes regarding the business relationship between him and them, and so should intuitively advertise as short a path to the victim's prefix as it can get away with. However, the attacker is now at a big disadvantage: he cannot pretend to own the prefix or even claim to be directly connected to the victim AS without being detected. Consequently, the path to the victim announced by the attacker to his neighbors must be of length at least 2 (and this path length is increased as the announcement is further propagated in the Internet). This, combined with the fact that BGP paths are typically short (consistently about 4-hops-long on average [35]), intuitively implies that the vast majority of ASes will not fall victim to the attack.

Importantly, path-end validation constitutes a radical departure from BGPsec's design philosophy, which focuses on achieving "rigorous AS path protection" [41] and does not distinguish between paths that are partially validated and paths that are not validated at all. Our results rely on the insight that while partial validation of paths does not *always* benefit security, there is significant benefit in validation of *path suffixes*, which forces the attacker to announce longer paths. Moreover, the shortness of interdomain routes implies that even validating 1-hop suffixes (that is, path-end validation) provides significant benefits, as discussed above.

Our contributions:

Path-end validation (Section 2). We identify path-end validation as a target security objective that is both achievable without modifying the routing infrastructure and can significantly improve interdomain routing security even in partial deployment.

Path-end validation is safe (Section 3). We show that path-end validation is provably guaranteed to never destabilize the routing system and also to never worsen security as more ASes adopt the mechanism. While seemingly two obvious prerequisites for the adoption of any routing security mechanism, these are not met by BGPsec [33]

Evaluation of security benefits (Sections 4 and 5). We perform extensive simulations to evaluate the security impact of adopting path-end validation, for different adoption rates. The results are encouraging: significant impact is obtained even with very limited partial deployment. We also identify the potential for regional adoption of routing security mechanisms, possibly government sponsored/driven. Specifically, we analyze the impact of adoption of path-end validation in geographical/national regions, and its potential to protect local communication within these regions.

Handling other attacks (Section 6). We present and evaluate feasible and easily-deployable extension of path-end validation to validate more than the 1-AS-hop suffix of a BGP path. We also present a different simple extension to RPKI can prevent certian "route leak" incidents [28, 3, 4]. We point out that these extensions, similarly to path-end validation, are but modest extensions to RPKI and do not involve any changes to BGP routers.

Implementation (Section 7). We present an open-source implementation that involves grappling with operational issues such as BGP router configuration complexity.

We discuss related work and conclude in Sections 8 and 9, respectively.

2. PATH-END VALIDATION

We argue that any *deployable* and *effective* improvement to BGP security must satisfy two constraints:

Not involve replacing today's BGP routing infrastructure. One of the main concerns with BGPsec and similar proposals is that they require replacing/upgrading BGP routers and, consequently, the associated monetary and operational costs. Worse yet, these proposals often require validation of (multiple) signatures when processing BGP advertisements, as well as signing upon sending a BGP advertisement. Such requirements make deployment very challenging [21]. We therefore seek a security solution that, unlike BGPSec, only entails configuring today's BGP routers (using their existing capabilities and interfaces).

Provide significant security benefits in partial deployment. Deployment of a new mechanism for securing interdomain routing, which spans tens of thousands of independently administered ASes, is expected to be a long process in which ASes gradually adopt the new mechanism. We hence seek solutions that provide tangible security benefits in the realistic partial deployment scenario, i.e., in the long interim period of time in which the new mechanism is not ubiquitously adopted. This should be contrasted with BGPsec, which provides meagre benefits over RPKI under partial adoption [33].

We propose path-end validation as a means to achieve the two above desiderata.

2.1 Design

Path-end records. Path-end validation extends RPKI. An adopting AS must first authenticate through RPKI ownership of its resources: IP address blocks and AS numbers. The AS uses its private RPKI-authorized key to sign a *path-end record* which includes a list of approved adjacent ASes through which it can be reached. These records, received from different ASes, are stored in a database.

Path-end records can be extended to allow an AS to specify a different set of approved adjacent ASes for different IP prefixes (if that AS so desires). We discuss how this can be supported in Section 7.

Path-end filtering. Path-end validation extends RPKI's *offline* mechanism, which periodically syncs local caches at adopting ASes to global databases, and pushes the resulting whitelists to BGP routers [12]. Any BGP router anywhere can thus use the path-end records registered in the system to discard BGP path advertisements where the AS before last does not appear in the list specified by the origin (i.e., last) AS on the advertised path ("path-end forgery"). This filter extends RPKI to prevent not only prefix-hijacking and sub-prefix hijacking, but also next-AS attacks.

Example. Consider the network in Figure 1. AS 1 is the "victim", i.e., the AS whose traffic the attacker, AS 2, attempts to hijack. Suppose that AS 1, and also ASes 20, 200, and 300, are adopters. Path-end validation protects AS 1 from falling victim to next-AS path-manipulation attacks, where AS 2 announces the bogus route $2-1$ to the prefix 1.2.0.0/16, i.e., pretends to be directly connected to AS 1. Path-end validation does not protect against the "2-hop attack", in which AS 2 pretends to be directly connected to a neighbor of AS 1 (say, announces the bogus route $2-40-1$). However, as shown below, such attacks turn out to be quite ineffective as the attacker is forced to announce a 2-AS-hop path whereas BGP paths are typically short (about 4-hops-long on average [35]). We discuss, in Section 6, how path-end validation can be extended to validation of longer path-suffixes.

Path-end validation vs. BGPsec. While path-end validation might seem, at first glimpse, as a restricted variant of BGPsec, this is not so. As discussed above, in contrast to BGPsec, path-end validation does not rely on *online* cryptographic signing of BGP advertisements but instead on an *offline* syncing mechanism. Beyond avoiding the need for changes to BGP routers, this has important implications from a security perspective. Specifically, this allows validation of BGP advertisements even when there are intermediate legacy routers along the path. To illustrate this point, let us revisit Figure 1. Path-end validation enables protecting *all* adopters, including AS 20, from next-AS attacks against other adopters, e.g., AS 1. Moreover, even an isolated adopter on the path, such as AS 20, can protect the non-adopters "behind" it by preventing malicious routes from disseminating and, in particular, a malicious advertisement will not reach AS 30. We show in Section 4 that this valuable trait indeed leads to significant security benefits even in very partial deployment, greatly improving over the meagre benefits achievable with BGPsec [33].

Privacy-preserving mode. To accommodate ISPs reluctant to disclose the identities of their neighbors (more specifically, customers) for fear of competitors, our design supports a "privacy-preserving mode", where an ISP deploys the path-end filters, but does not register its neighbors in the database. This protects privacy-concerned ISPs from falling victim to next-AS attacks against others, without compromising privacy (and increases protection for the other ASes).

We point out, however, that: (1) Over 85% of ASes are not ISPs. In particular, a large fraction of Internet traffic is originated at and destined for ASes with no customers like Google, Netflix, etc. (We present in Section 4 the security benefits of path-end validation for large content providers.) While some non-ISPs might be interested in keeping the identities of the neighbors private, the vast majority do not have a business interest in keeping this information secret (see, e.g., PeeringDB [43]). (2) Even if an ISP does not reveal the identity of a customer, that customer can choose to reveal its connection to this ISP so as to protect itself. (3) As our results in the following sections indicate, validating the last 1-hop suffix of an AS path is sufficient to provide significant security guarantees, and so non-ISPs can achieve a high level of protection even without ISPs' cooperation. (4) A lot of information about the list of neighbors of an AS can easily be deduced from examining BGP advertisements from multiple (publicly available) vantage points. Hence, even an ISP concerned about the privacy of its list of neighbors might, in practice, not enjoy substantial privacy.

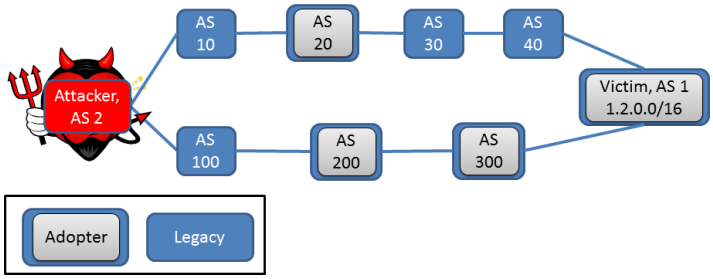


Figure 1: Partial deployment example

3. PREREQUISITES

As shown in [33], beyond BGPsec's meagre security benefits in partial adoption, it does not satisfy what we view as two natural prerequisites for the adoption of any inter-domain routing security mechanism: (1) *never to destabilize* the routing system, and (2) *never to worsen security*, i.e., an attacker's ability to attract traffic should never improve as more ASes adopt the security mechanism. We refer to the latter prerequisite as security monotonicity.

We prove below that path-end validation, unlike BGPsec, satisfies these requirements in any deployment scenario. We show in Section 4 that path-end validation also significantly outperforms BGPsec in terms of security benefits in partial adoption.

3.1 Model

We briefly overview the standard model for reasoning about BGP dynamics, namely, the Gao-Rexford model [17], and then present the threat model as well as path-end validation deployment model. We then present our results for stability and security-monotonicity within these models.

The network. The network is modeled as an undirected graph $\mathcal{G} = (\mathcal{V}, E)$, where the vertices represent the ASes, labeled by the corresponding AS numbers, and the edges represent the communication links between them. A *route* in our model is a sequence of vertices in $\mathcal{V}$ that ends with the

destination prefix π. Each link in E is annotated with one of the following two *business relationships*: *customer-provider* (directed from customer to provider), indicating that the customer pays the provider for connectivity, or *peer-to-peer*, where two neighboring ASes agree to transit each other's customer traffic at no cost.

The Gao-Rexford conditions. A network where each vertex runs the BGP protocol to select a route to each destination IP prefix belonging to another AS is called a *BGP system*. We refer the reader to [17] for a detailed exposition of the BGP routing process model. We assume that the three following so called "Gao-Rexford conditions" hold:

- **Topology Condition:** No customer-provider cycles exist in the AS-level graph.
- **Preference Condition:** Prefer customer-learned routes to peer- and provider-learned routes.
- **Export condition:** Only export provider and peer-learned routes to customers.

Gao and Rexford prove that the combination of these three conditions is sufficient to guarantee the convergence of the BGP system to a stable routing configuration [17].

Threat model. We model attackers as a set $Adv \subset V$. An attacker must advertise a single, "fixed" route to a prefix π to each of its neighbors, yet may announce different routes to different neighbors. As an attacker cannot lie about its identity, i.e., AS number, to its neighbors, each route announced by an attacker must begin with the attacker's vertex (AS number). The attacker can, however, present himself as the owner of a prefix (prefix/subprefix hijack), a direct neighbor of the victim (next-AS attack), launch a 2-hop attack, etc..

Path-end validation in partial adoption. To model a BGP system in which path-end validation is partially (or fully) adopted we include in our model the set $Adpt \subseteq V$, of vertices that perform path-end filtering. Recall that path-end validation is deployed on top of RPKI and so adopters are protected from both prefix and subprefix hijacking (by RPKI) and next-AS attacks (by path-end validation). Hence, if a vertex $d \in \mathcal{V}$ registered a path-end record, any vertex $v \in Adpt$ discards routes to π that do not end with $d - \pi$ or $n - d - \pi$, where the prefix π is owned by d and n is an authorized neighbor of d.

3.2 Stability

In [32] Lychev et al. show that, under the Gao-Rexford conditions, BGP is not only guaranteed to converge to a stable state, but this is so even in the presence of fixed-route attackers [32]. Hence, misconfigurations/attacks such as prefix hijacks and next-AS attacks cannot destabilize BGP routing. We show that this statement holds true also in *any* deployment scenario of path-end validation. We point out that this should be contrasted with the risk of routing instabilities resulting from inconsistent BGPsec route selection across different ASes, even when *not* under attack (see [33] for details).

THEOREM 1. *Under the Gao-Rexford conditions, a BGP system where any set of vertices Adpt adopts path-end validation is guaranteed to converge to a stable routing configuration even in the presence any set Adv of fixed-route attackers.*

Our proof, which extends the proof in [32], is omitted due to space constraints.

3.3 Security-Monotonicity

Informally, an interdomain routing security mechanism is *security monotone* if an attacker's ability to attract traffic to its network is never enhanced as more and more ASes adopt the mechanism. We show that this indeed holds in the context of path-end validation.

THEOREM 2. *Under the Gao-Rexford conditions, for any BGP system, attacker AS a, and victim AS v, if traffic from some source-AS x destined for v does not reach a when the set of adopters is $Adpt \subseteq V$ then this also holds for any set of adopters that is a super-set of Adpt.*

Again, the proof is omitted due to space constraints.

4. PATH-END VALIDATION *VS.* BGPSEC

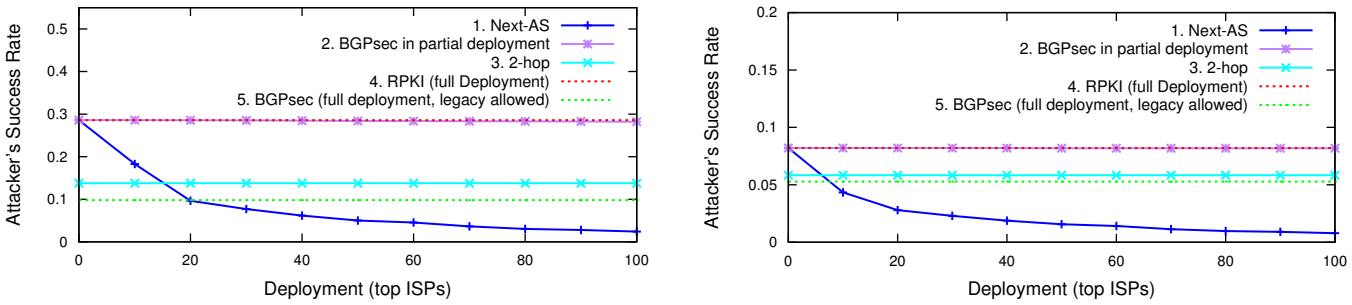
In this section we assume that RPKI is globally adopted and compare the level of security provided by path-end validation and BGPsec in *partial* deployment. We show that even under very limited adoption, path-end validation provides security benefits close to those of BGPsec in *full* deployment, whereas, as already discovered in [33], BGPsec provides meagre benefits in partial deployment. We later show (Section 5) how path-end validation fares when RPKI is also only *partially* deployed.

4.1 Methodology

Simulation framework. We evaluate different attack strategies and quantify the attacker's success by the fraction of ASes he is able to attract, as in [18, 33]. Our simulations apply the BGP route-computation framework presented in [18, 19, 23] to the empirically-derived CAIDA AS-level graph [8] from January 2016 (links are annotated with inferred bilateral business relationships and contain previously hidden peering links within IXPs [20][1]). Our simulations, similarly to [18, 19, 23], compute the BGP routing outcome reached when each AS selects BGP paths to other ASes according the following simple routing policy:

1. **Local preferences over routes**: Prefer routes in which the next-hop AS is a customer over routes in which the

[1]For example, in the induced AS-graph, Google has 1325 peers, and each of the 5 largest content providers (specified in [33]) has over 850 peers. We point out that while many peering links might still remain hidden, this new CAIDA dataset contains over many more peering connections than datasets used in past studies.



(a) Global security evaluation

(b) Protection of top content providers

Figure 2: Attacker success rate for different strategies as function of adopters.

next-hop is a peer. Prefer routes in which the next-hop AS is a peer over routes in which the next-hop is a provider.

2. **AS-path length**: Prefer shorter routes (in terms of AS-hop count) over longer routes.

3. **Tie break**: Break ties between routes based on the number of the next-hop AS on the BGP path. (Since our simulation framework treats an AS as an atomic entity, as in [18, 19, 23], this step substitutes tie-breaking based on intradomain routing considerations.)

4. **Export**: If the selected route is learned from a customer AS, advertise the route to all neighboring ASes. Otherwise, the route is exported to customers only.

When an AS adopts path-end validation, the following route-filtering step is added *before* the above-specified steps in its BGP decision process (as with today's origin authentication via RPKI).

- **Security**: When a BGP advertisement from a neighbor is incompatible with the path-end records in the RPKI, discard the advertisement.

We averaged our measurements over 10^6 combinations of attacker-victim ASes, where the method for selecting the attacker and victim pairs depends on the particular scenario we investigate below.

Who are the best adopters? Ideally, our security evaluation results would quantify the maximum security benefits from path-end validation achievable from any given number of adopters. We prove, however, that identifying the "best" set of adopters of a given size $k > 0$ is computationally hard.

We consider the "Max-k-Security" problem from [33], adapted to our context: Given an AS graph, a specific attacker-victim pair (a, v), and a parameter $k > 0$, find a set S of size k of path-end validation adopters that minimizes the total number of ASes whose paths reach the attacker. We show that, as in the case of BGPsec [33], this is computationally hard.

THEOREM 3. *Max-k-Security is NP-hard.*

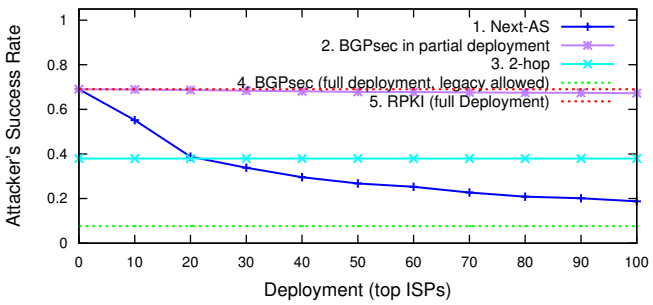
The proof, which resembles that in [33], is omitted due to space constraints. To quantify the security benefits of path-end validation in partial deployment we must apply reasonable heuristics for identifying "good" adopters (from a global security perspective). Specifically, our experiments evaluate the security benefits from adoption of path-end validation by the top ISPs, i.e., the ASes with largest numbers of AS customers. Our results demonstrate that this can indeed yield significant security benefits even when very few large ISPs adopt. We validate this result with robustness tests.

4.2 Security Evaluation

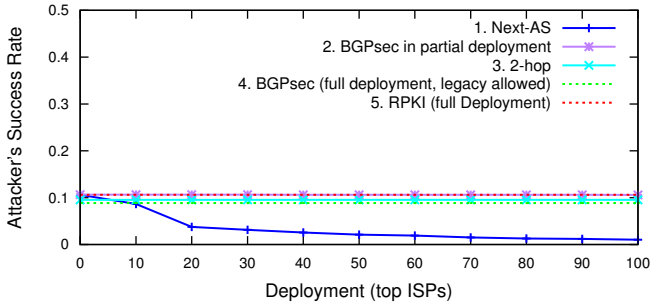
Consider the graphs in Figure 2. The x-axis describes 11 deployment scenarios corresponding to adoption of path-end validation by the set of $0, 10, \ldots, 100$ largest ISPs. The y-axis is the average fraction of ASes on the Internet whose traffic the attacker is able to attract to his network. Since deployment of RPKI implies that ASes can detect and block prefix and subprefix hijack attempts, we consider two remaining path-manipulation strategies for the attacker when the victim registers a path-end record: the next-AS attack, and the 2-hop attack (we later show that other path-manipulation attacks provide lower success rates). Path-end validation implies that adopters can always detect and ignore the first attack (on other adopters), while the second attack goes unnoticed by the defense. BGPsec, on the other hand, only allows adopters to validate a path where *all* ASes on the path are adopters (regardless of its length, see discussion in Section 2).

The graphs also present two dashed reference lines: (1) the attacker's success rate when RPKI is fully deployed, when launching the next-AS attack (which RPKI does not block), and (2) the attacker's success rate when BGPsec is fully deployed, but legacy BGP is not deprecated and so the attacker can launch "protocol downgrade attacks" by advertising legacy BGP paths, as studied in [33] (our simulations also confirm the simulation results in [33]).

Figure 2a compares the Internet-wide security benefits of the different mechanisms for uniformly-selected attacker-victim pairs. We show later that the results for *specific* choices of attacker and victims (e.g., victim CDNs, attacker-victim pairs from past incidents) exhibit the same trends. As shown in the figure, path-end validation is remarkably effective at



(a) Attacker is a large ISP, victim is a stub



(b) Attacker is a stub, victim is a large ISP

Figure 3: Attacker success rate for different victim/attacker classes

thwarting the next-AS attack, in contrast to the meagre improvement that BGPsec achieves over RPKI under the same partial deployment scenario (compare lines 1 and 2 in the figure). Even with only 20 adopters, the attacker is better off resorting to the 2-hop attack to avoid detection, resulting in success rate of 13.7% (see line 3)—a big improvement over RPKI in full deployment (28.5% success rate, see line 4) and not far from BGPsec in full deployment (roughly 10% success rate, see line 5). In fact, with 100 adopters performing path-end validation the next-AS attack vector is almost completely blocked as the attacker's success rate goes below 3%, while the mirror image for BGPsec is 28.2% (a mere 0.3% improvement over RPKI, see line 2).

Protection for content providers. Figure 2b describes the protection that path-end validation provides to large content providers (Google, Amazon, Netflix, etc. (list taken from [33]), as these generate a significant fraction of data traffic on the Internet. We evaluated, for each victim content provider, the success rate of an attacker drawn uniformly at random. Our results show that path-end validation provides significant security benefits, reducing the attacker's success rate with his best strategy (2-hop attack) to 5.8% with only 20 adopters, improving over RPKI (8.3%) and partial deployment of BGPsec (8.2% when all top 100 ISPs adopt). In fact, path-end validation almost reaches the best we could hope for before BGP is deprecated, i.e., 5.3% attacker success rate when BGPsec is fully deployed (but the attacker can still advertise legacy BGP routes).

Results for specific classes of attackers and victims. Our results above are for the scenario of random attackers and/or victims. Since most ASes on the Internet (over 85%) are stubs, i.e., ASes with no customer ASes, this greatly biases the results towards stub attackers and victims. To see how an attacker/victim's size and location within the ISP hierarchy affect the results we ran the same simulations for different classes of attackers and victims. We consider four classes: (1) large ISPs (250+ customers), (2) medium ISPs (250 $>$ customers $\geq$ 25), (3) small ISPs (25 $>$ customers $\geq$ 1), and (4) stubs (no customers). We generated results, similarly to the results in Figure 2a, for all 16 combinations of attackers and victims in these categories.

Our results for all 16 attacker-victim scenarios reveal that

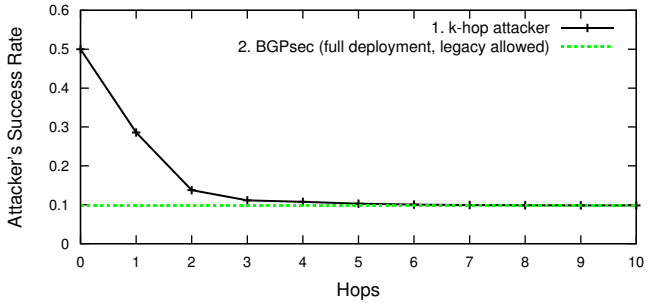
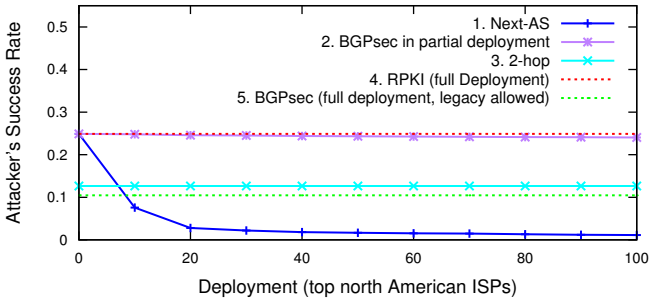


Figure 4: Attacker success rate as a function of hops in malicious BGP advertisement

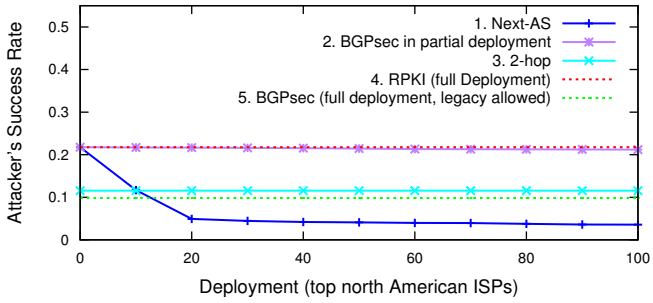
with relatively few adopters (the numbers vary from less than 10 to 100, depending on the categories of attacker and victim), path-end validation renders the next-AS attack less effective than the 2-hop attack, as before. We present, due space constraints, our results for the two extremes: the most central ASes (transit-wise), namely the large ISPs, and the least central ASes, i.e., stubs. See Figures 3a and 3b for results in the scenario that the attacker is a large ISP and the victim is a stub, and vice versa, respectively. Not surprisingly, large ISPs are very powerful attackers, as indicated by the attacker's success rate, whereas stubs are quite weak attackers. Observe, however, that path-end validation has the same qualitative effect in both scenarios: the attacker is better off, even with quite few adopters, bypassing path-end validation and announcing a longer (2-hop) path.

Effectiveness of k-hop attacks. Figure 4 plots the success rate of an attacker launching a "k-hop" path-manipulation attack when no defense is deployed, i.e., the attacker announces a bogus k-hop path to the victim, for different values of k (attacker and victim pairs picked uniformly at random). The x-axis specifies the number of hops k and the y-axis specifies the average success rate of a randomly selected attacker launching a k-hop attack on a randomly selected victim. We also show, as reference, the success rate of the attacker when BGPsec is fully deployed but legacy BGP advertisements are allowed [33] (line 2).

We view this figure as capturing the key idea behind path-end validation. When $k = 0$, the attacker is simply launching

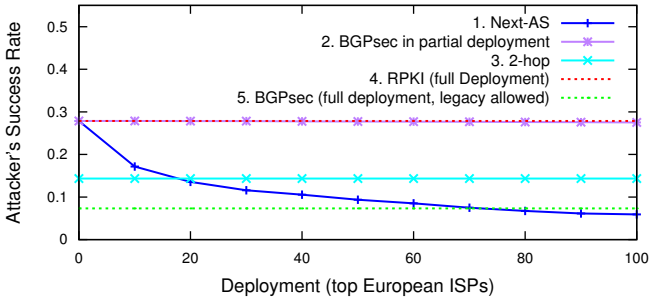


(a) Internal attackers

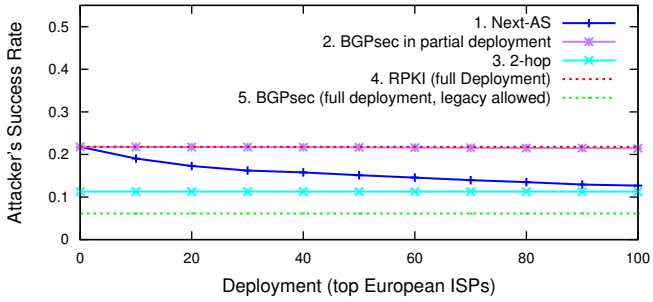


(b) External attackers

Figure 5: Protection for North-American ASes by local adopters



(a) Internal attackers



(b) External attackers

Figure 6: Protection for European ASes by local adopters

a prefix hijack (i.e., announcing itself to be the destination of the prefix). Observe, that this is much more effective than the 1-hop attack, i.e., the next-AS attack. Indeed, this attack (and subprefix hijacking) is precisely what RPKI is meant to prevent. Observe also that while the 1-hop attack, i.e., the next-AS attack, is significantly more beneficial to the attacker than the 2-hop attack, the 2-hop attack does not fare significantly better than the 3-hop attack. We conclude that path-end validation indeed gets the most "bang for the buck" in terms of security against path-manipulation attacks, striking the right balance between deployability and efficiency.

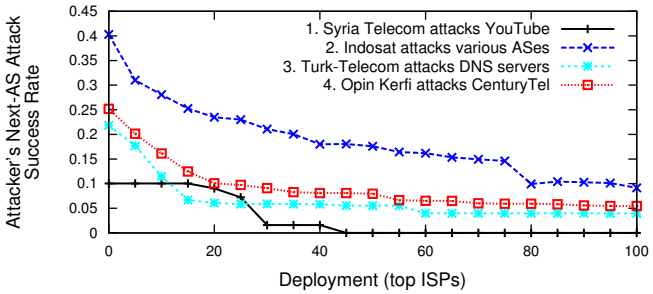
4.3 Geography-Based Deployment

One possible strategy for boosting initial deployment of path validation mechanisms is for governments to incentivize large ISPs in their countries to adopt (i.e., install the corresponding filtering rules in their routers). We investigate whether such *local* adoption can protect *l*ocal communication, i.e., protect the ASes in that geographical region, and compare between path-end validation and BGPsec as potential mechanisms to perform path validation. This is important, for instance, since many end-users retrieve content from servers in their geographic region due to the popularity of content delivery networks, and communicate with local services (banking, healthcare etc.), and also to ensure the availability of critical national infrastructures.

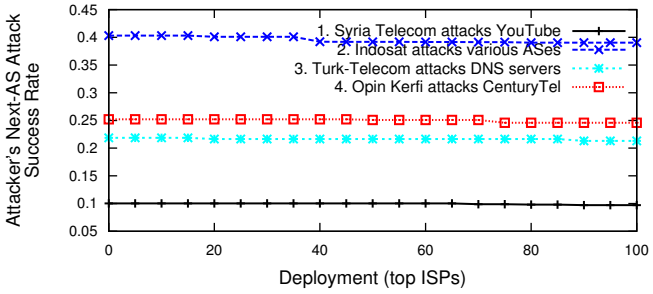
We used the Regional Internet Registries (RIRs) division of the world into five geographic regions and considered adoption only by ISPs in a particular region. We then measured how many benign ASes *in the region* are fooled to take a malicious route to a victim *in the region* advertised by attackers (internal and external to the region).

Figure 5 shows the fraction of North-American ASes that an attacker can attract when trying to capture traffic to a North-American AS. We find that with only 10 *adopters*, path-end validation protects the communication between two ASes in North America, even if the attacker is co-located in North America (Figure 5a), reducing the attacker's success rate to just above 13% with his best strategy (2-hop attack). The corresponding results for Europe, presented in Figure 6, show that the top 20 European ISPs need to adopt the protocol to achieve a similar effect. We also find that RPKI provides relatively a high level of security for European ASes against external attackers (see Figure 6b), yet Europe still benefits from deployment of path-end validation, which gradually decreases the attacker's success rate until eventually when the top 100 ISPs adopt the defense the attacker's best strategy becomes the 2-hop attack, yielding 11.2% success rate. These results provide significant motivation for adoption, considering recent attacks on ASes from Iceland and Belarus [2], and the emphasis on routing security of the European Union. We also contrast them with the results for BGPsec under the same partial deployment which provides only marginal benefits over RPKI, line 2 in both figures.

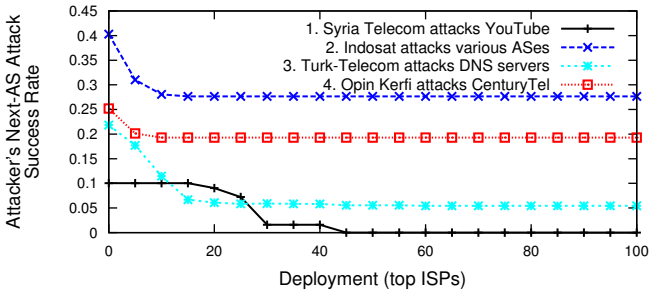
Intuitively, these positive results are not at all surprising since much of path-end validation's success can be attributed to the shortness of BGP routes, and routes within a particular



(a) Path-end validation



(b) BGPsec



(c) *Best* strategy against path-end validation

Figure 7: High profile past incidents: attacker's success rate as function of the number of adopters.

region are expected to be shorter on average (in our AS-level graph [8] simulations, routes within North-America and Europe regions are, on average, 3.2 and 3.6 hops long, respectively, while routes on the global Internet are about 4-hops long on average [35]). Also, traffic within country-/region-level geographies is likely to traverse fewer large and central ISPs, and so adoption of path-end validation by the region's largest ISPs can yield significant benefits.

4.4 Revisiting High-Profile Past Incidents

Our results in Section 4.2 quantify the *average* success rate of random attackers and victims. What about *specific* attacker-victim cases?

We revisit four fairly recent high-profile prefix-hijack incidents to illustrate the immediate security benefits of deploying path-end validation with RPKI: (1) Syria-Telecom hijacks YouTube [9] on December 9th, 2014 (2) Indosat hijacks over 400,000 prefixes on April 3rd, 2014 [1]; (3) Turk-Telecom hijacks DNS resolvers in Google, OpenDNS and Level3 on March 29, 2014 [5]; and (4) Opin Kerfi's (an ISP in Iceland) repeated prefix-hijacks [2] in December 2013. Some of these incidents are attributed to benign configuration errors while others are suspected attacks. Because this section considers the scenario that RPKI is fully deployed, such prefix-hijacks can be detected and mitigated. We thus consider the next-AS attack, which is not prevented by RPKI, with respect to the same attacker-victim pairs.

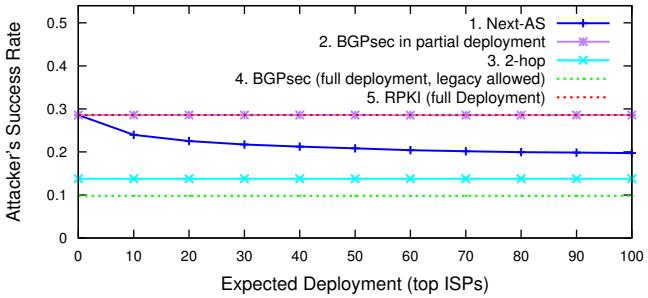
Clearly, no simulation framework is rich enough to capture all intricacies of interdomain routing (e.g., ASes' actual routing policies). Our aim is therefore not to predict the routing outcome, but to get a high-level idea of path-end validation's potential influence in these concrete scenarios. We computed, for every attacker-victim pair, a next-AS attacker's success rate with X adopters from the largest ISPs, where $X = 0, 5, \ldots, 100$. Figure 7 describes the attacker's success rate for path-end validation and for BGPsec in partial deployment. Observe that even with a modest number of path-end validation adopters, the attacker's best strategy is to launch the 2-hop attack and avoid detection by the path-end validation mechanism (see Figure 7a). In contrast, BGPsec exhibits far inferior security benefits (see Figure 7b).

Figure 7c plots the attacker's success rate in each deployment scenario ($X = 0, 5 \ldots, 100$ adopters) for his *best* attack strategy among the two (next-AS and 2-hop) against the path-end validation defense. Consider, for example, the Turk-Telecom case. Before any AS adopts path-end validation, the attacker's best strategy is the next-AS attack, resulting in a success rate of almost 25%, i.e., attracting nearly a quarter of the Internet. As more and more large ISPs adopt the path-end validation defense, the success of the next-AS attack is significantly decreased. Indeed, even with 15 adopters, the attacker is better off switching to the 2-hop attack to bypass the path-end validation mechanism, and so the attacker's success rate remains fixed at about 5% henceforth.

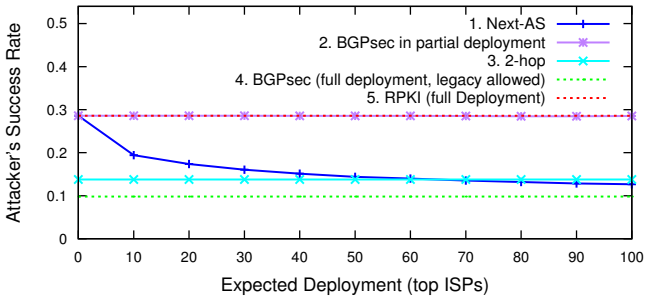
4.5 Robustness Tests

Our evaluation thus far considered deployment by a small set of the very top ISPs. We now turn to evaluate the security benefits where path-end validation or BGPsec are only adopted by *some* of the top ISPs. The results in Figure 8 compare path-end validation to BGPsec under *probabilistic* deployment scenarios. The x-axis describes 11 deployment scenarios, where for each value x the corresponding deployment scenario is as follows: We consider, for a specified probability p ($p = 0.25, 0.5, 0.75$) the set of $\frac{x}{p}$ top ISPs. We then select each of these ISPs as an adopter with probability p. Hence, in deployment scenario x, the *expected* number of adopters is $\frac{x}{p} \cdot p = x$. The measurement for each deployment scenario is repeated 20 times and Figure 8 plots the averaged results.

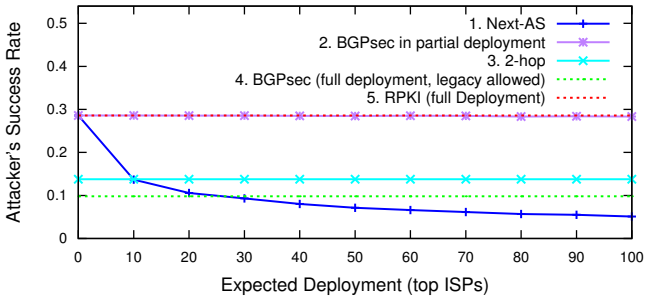
We observe that as path-end validation benefits from the adoption of the top ISPs on the Internet, the attacker's success rate naturally grows as the probability for adoption reduces. However, even in this probabilistic deployment scenario path-end validation yields good results, significantly outperforming BGPsec. For example, for adoption rate of 50%, path-end validation still provides significant benefits



(a) 75% adoption rate



(b) 50% adoption rate



(c) 25% adoption rate

Figure 8: Security benefits under probabilistic adoption by the top ISPs

and the attacker is better off switching to the 2-hop attack with only 60 adopters. BGPsec under a similar scenario provides only marginal 0.2% improvement over RPKI.

This difference between the two defenses against path-manipulation attacks under probabilistic adoption is particularly important when considering the difference in effort required for deployment (see discussion in Section 2): while path-end validation was designed to be very easy to deploy and only introduces changes the BGP router's configuration, BGPsec requires new hardware and is therefore expected to have a long period until it is extensively deployed [21].

5. RPKI IN PARTIAL DEPLOYMENT

RPKI deployment is still far from ubiquitous. Indeed, only about 6% of IP prefixes advertised in BGP are in the RPKI repository [37]. Understanding the root causes for RPKI's slow adoption is beyond the scope of this work and is an important direction for future research. Our focus in this section is on showing that path end validation can provide benefits without waiting for extensive deployment of RPKI. We use the simulation framework presented in Section 4 to evaluate interdomain routing security when adopters deploy RPKI with path-end validation, but all other ASes deploy neither mechanism.

Prefix hijacks. Since in this section we consider the case where RPKI is very partially deployed, attackers may perform prefix hijacks, i.e., advertise the same prefix as the victim AS, with the plausible excuse of benign misconfiguration error (when detected) [7]. Although hijacking attacks are blocked by RPKI adopters, they can be very effective when RPKI is insufficiently deployed.

Security evaluation. In Figure 9 we plot the attacker's success rates when launching a prefix hijack, which is filtered by adopters. (We already quantified path-end validation's effectiveness against next-AS attacks in Section 4.) The dashed reference line describes next-AS attacker's success when RPKI is fully deployed, but with no path-end validation deployment. Figure 9a plots the average attacker's success rate for uniformly chosen attacker and victim pairs. We find that when 20 large ISPs adopt RPKI, the attacker is better off launching a next-hop attack than a prefix hijack so as to circumvent RPKI. This is precisely where the benefits of path-end validation start to kick in. We conclude that even in early stages of RPKI adoption, path-end validation can already provide tangible security benefits over RPKI. This should be contrasted with deployment of BGPsec, which relies on extensive deployment of RPKI [33]. Similar measurements focusing only on the security benefits for large content providers, described in Figure 9b, show the same trends.

6. HANDLING OTHER ATTACKS

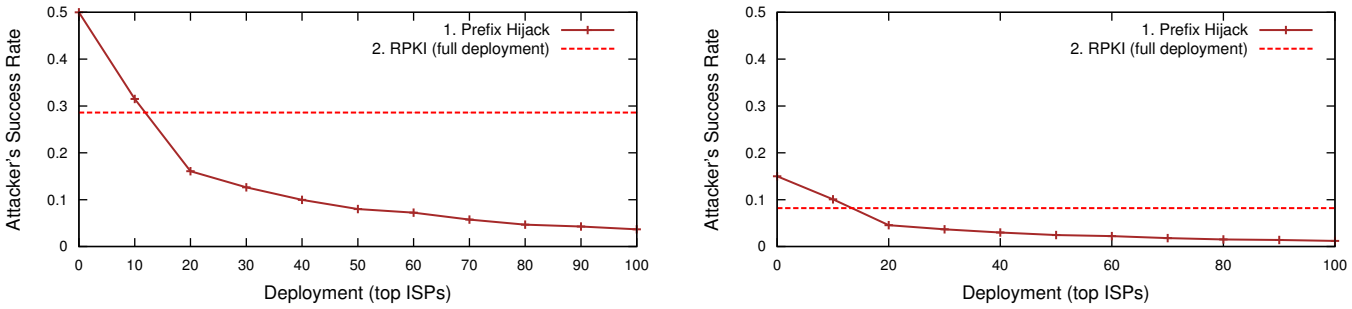
Path-end validation focuses on authentication of the 1-AS-hop suffix of a BGP advertisement so as to protect against next-AS attacks. We next describe two simple extensions to RPKI, designed to defend against other path-manipulation attacks. These extensions, similarly to path-end validation, satisfy the constraints, discussed in Section 2, on providing a deployable and effective defense. In particular, these extensions do not involve any changes to BGP routers.

We first show that at no additional deployment cost, path-end validation can be extended to validate longer path-suffixes. We then show how the addition of a single-bit field to the path-end record can mitigate certain types of "route-leakage" incidents. We incorporated these extensions into our open-source implementation, presented in Section 7.

We last discuss which path-manipulation attack strategies remain at the attacker's disposal despite these extensions and discuss their effectiveness.

6.1 Validating Longer Path-Suffixes

We introduced path-end validation as a means to ensure that the last AS-hop on a BGP path is valid. Observe, however, that the adjacency list in the path-end record, allows to verify whether *any link* to/from an adopting AS on a path is consistent with the path-end record for that AS, even if the



(a) Global security evaluation.

(b) Protection of top content providers.

Figure 9: Attacker success rate for different strategies as function of adopters under partial RPKI deployment.

adopter is *not* the origin (i.e., last) AS. Consider, for example, the scenario that both the last and the next-to-last ASes on a BGP path are adopters. Any other adopter should be able to validate the 2-AS-hops suffix of that path by examining the path-records of these two ASes. In general, if the last k ASes on a BGP path published path-end records, then all adopters can validate the last k AS-hops on that path.

Let us revisit the deployment scenario illustrated in Figure 1, in the event that AS 2 chooses to launch the 2-hop attack against AS 1 by advertising the false route 2-300-1, this attack will be detected since AS 300 is an adopter and the attacker is not one of its valid neighbors. Instead, the attacker, AS 2, may exploit AS 1's only legacy neighbor, AS 40, and announce the route 2-40-1 to avoid detection. As adoption gains traction, however, the attacker's options are incrementally limited. Specifically, when AS 40 will also adopt, AS 1 will become protected from 2-hop attacks.

We describe, in Section 7, how to enforce path-end validation rules without changes to today's router configuration interface. We show that filtering BGP paths with invalid links, as explained above, has the exact same complexity as enforcing these filters only for the last AS-hop on the path. Hence, extending path-end validation to validate longer path-suffixes comes at no extra cost.

We point out, however, that as our results in Section 4 indicate, k-hop attacks, for $k > 1$, are not very effective. Hence, while validating path-suffixes longer than the 1-AS-hop (i.e., validation beyond the last hop) can help in reducing the effectiveness of attacks is specific scenarios, this cannot, on average, significantly improve over path-end validation even if ubiquitously adopted.

6.2 Mitigating Route-Leaks

Route leakage is an incident where an AS propagates a BGP path advertisement in a manner that violates *its own* path-export policy, e.g., announces a BGP path from one provider to another provider. This can be the result of a misconfiguration or, alternatively, an attack that leverages a compromised BGP router. Route leaks circumvent even full deployment of the most powerful BGP security mechanisms, e.g., BGPsec [22], and are a bothersome security vulnerability in practice, as evidenced by recent inadvertent incidents [3, 4, 28].

The vast majority of ASes on the Internet (over 85%) are stubs, i.e., have no AS customers and so do not provide transit services to other ASes. We observe that RPKI can easily be extended to protect from route-leak attacks (e.g., due to configuration errors or compromised equipment) by stubs. Specifically, path-end validation can be extended to allow a stub to specify that its AS number should *only* appear at the end of a BGP path (since stubs, by definition, do not transit traffic to external destinations). To this end, a Boolean flag is added to path-end records to indicate whether the origin provides may transit traffic or should only appear at the end of the route. Observe that this implies that in the event of a route-leak from a stub adopter, all other adopters will indeed not fall victim to the attack. Consider again the network in Figure 1 and assume that AS 1's router is compromised and propagates a BGP advertisement received from its provider AS 40 to its other provider AS 300 (e.g., advertising a popular service such as Amazon as in a recent incident [28]). The new "non-transit indicator" flag allows AS 300 to discard this BGP advertisement, preventing further dissemination into the network (e.g., to its own provider, AS 200).

We point out that this can be viewed as a non-local analogue of defensive filtering [22], i.e., ISPs policing the BGP advertisements of their stub AS customers.

To measure how deployment of this simple extension to RPKI can aid in mitigating route leaks we use the simulation framework in Section 4. We evaluate our defense under two scenarios: first when the victim is a randomly chosen AS (chosen uniformly), and second when the victim is a large content provider from the list in [33]. The route leaker in both cases is a multi-homed stub AS that advertises a BGP path it learns to the victim to all of its neighbors (excluding the one it obtained that route from) in a manner that violates the Gao-Rexford export condition (see Section 3).

Figure 10 describes our results both for the general case and for content providers. We observe that since the leaked route must contain at least two hops (often more), route leakage typically has lower success rate than route-manipulation attacks (e.g., the next-AS attack), unless those of the latter type are prevented by a defense mechanism (path-end validation, BGPsec, etc.). We find that our extension allows to mitigate the route-leakage threat, halving its effect already with 10 adopters (both when the victim is a random AS and

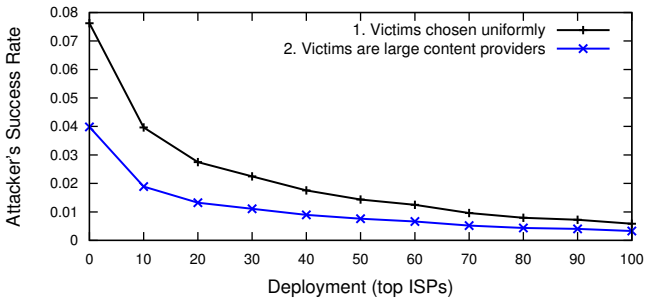


Figure 10: Path-end validation as route-leaks defense

when it is one of the content providers) and eventually reaching 0.5% success rate when the top 100 ISPs adopt.

6.3 What is Left?

We discussed above two extensions to RPKI designed to prevent path-manipulation attacks beyond next-AS attacks. We now investigate which path-manipulation attacks are *not* eliminated by path-end validation and these two extensions, even when *fully* deployed. Understanding how such attacks can be completely eliminated in a deployable manner (e.g., without resorting to heavyweight online cryptography or blowup in configuration complexity) remains an open question.

Advertising existent, yet unavailable paths. Observe that the first of the two extensions to RPKI discussed above, when ubiquitously adopted, prevents an attacker from advertising a nonexistent path without being detected. This does not, however, prevent the attacker AS from advertising an *existent* path, even though that path was never advertised to it by a neighboring AS. Importantly, however, unless the attacker is directly connected to the victim, such an attack involves announcing a path of length at least 2 and therefore, as shown in Section 4, is not very effective.

Colluding attackers. As in essentially all past research on BGP security, our threat model does not consider the possibility that multiple attackers collude. An attacker AS a_1 can, for instance, approve a colluding attacker AS a_2 as a neighbor in its path-end record, thus enable a_2 to advertise the AS-path (a_2, a_1, v) to a victim v that neighbors AS A. However, this attack, too, results in a path of length 2 or more, and so is significantly less harmful (on average) than next-AS attacks (Section 4).

Route leaks by ISPs. Our second extension to RPKI prevents (in full deployment) all route leakage from stubs, yet does not prevent route leaks by ISPs. We point out, however, that, as before, unless the route leaker happens to be the direct neighbor of the victim, this attack too cannot be more effective than the 2-hop attack.

Data-plane attacks. Similarly to BGPsec and other proposals for securing the BGP *control plane*, path-end valiation does not prevent *data-plane attacks*, in which a malicious or faulty router advertises a legitimate BGP path but *forwards* data traffic in a manner that is incompatible with its BGP advertisements. Even though such an AS can always break connectivity by simply dropping all packets, the authenticity and confidentiality of the communication can be protected using cryptographic protocols such as IPsec [26] and TLS [16].

We conclude that eliminating (sub)prefix hijacking and next-AS attacks indeed significantly limits the ability of the attacker.

7. PROTOTYPE

We present a prototype implementation, which complements RPKI and allows to deploy path-end validation with today's routing infrastructure. We envision this prototype as a first step, providing an immediate defense against path-manipulation attacks, before path-end validation is integrated into RPKI. To allow the community to deploy and experiment with path-end validation, our implementation is open-source and available at https://github.com/yossigi/pathend.

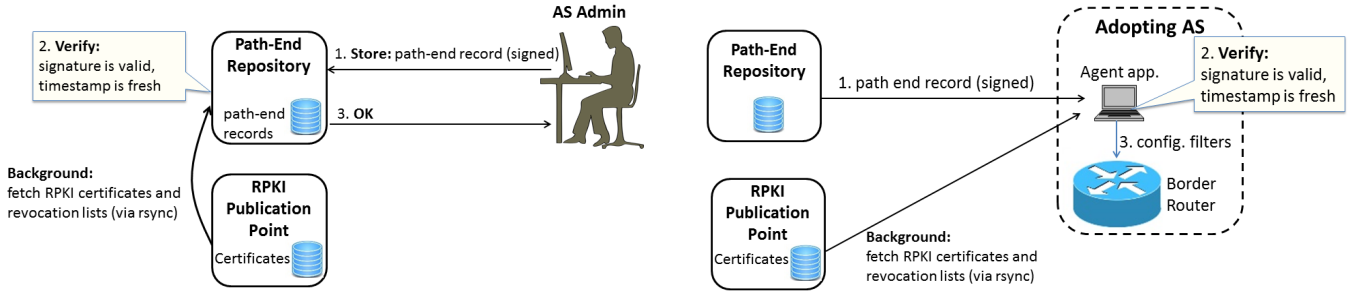
7.1 Implementation Details

Our implementation uses path-end records, where an origin AS holding an RPKI certificate specifies a list of approved neighboring ASes and whether it provides transit services. We use the following ASN.1 syntax to define the record format:

```
PathEndRecord ::= SEQUENCE {
        timestamp Time,
        origin  ASID,
        adjList SEQUENCE (SIZE(1..MAX)) OF ASID,
        transit_flag BOOLEAN
    }
```

Path end records are stored in public repositories, similar to RPKI's publication points [24]. We envision that the two repositories of both mechanisms may be co-located to avoid establishment of additional services, however, to support path-end validation on today's Internet, i.e., before RPKI publication points support distribution of path-end records, we implement the system's repositories. When a repository receives an AS's path-end record to store (via HTTP POST), along with a signature from the origin AS, it uses that AS's RPKI certificate to verify the signature over its path-end record (we utilize RPKI's certificate revocation lists to remove records in case the signing key was revoked), as well as validates that the timestamp specified in the record is not before an already existing entry for the same origin. See Figure 11a. An AS can update or delete its path-end records using a signed announcement sent to the repositories, similarly to Route Origin Authorization records (ROAs) in RPKI.

Since BGP routers do not yet accept path-end records, we also implement an "agent application" that updates periodically from the repositories and configures BGP routers in the adopter's network with path-end-filtering policies. See illustration in Figure 11b. To avoid trusting the path-end record repository (until the trusted RPKI publication points support distribution of path-end records), the agent retrieves the corresponding record signature, which path-end repositories store along with the records. It then verifies the signature



(a) AS Administrator stores path-end record

(b) Agent application configures filtering rules

Figure 11: Path-end validation deployment

using the RPKI certificates retrieved from RPKI's publication points (similarly to the verification process performed by the repository). The agent retrieves each update from a random path-end repository, so as to ensure that a compromised repository cannot remove a record or provide an obsolete image of the database, i.e., prevent "mirror world attacks" (see [29] for discussion on such attacks within RPKI). The agent application supports an automated mode, where the network administrator provides the credentials needed to configure a BGP router, and agent automatically connects to the router to deploy filtering rules according to the path-end records it retrieves. The agent also supports a manual mode, where it only outputs the filtering policies to a router configuration file, which the administrator can later apply.

7.2 Deploying Path-End Filtering Rules

We next describe how the agent configures filtering rules for BGP advertisements on *today's* routers. For each AS, the agent deploys at most *two* filtering rules. This results in less than a fifth of the rules required for origin authentication with RPKI, which involves a filtering rule per IP-prefix, origin-AS pair (there are roughly 53K ASes advertising over 590K prefixes). We therefore believe that path-end validation can scale to support the entire set of ASes.

We note that if path-end validation were fully integrated into RPKI (as advocated here), then it could piggyback RPKI's existing filtering mechanism, i.e., extend RPKI's origin authentication policies, to support validation of last-hop. This would allow to establish fine-grained path-end filtering mechanism on a per-prefix, rather than per-AS granularity, without adding to the number of filtering rules established by vanilla RPKI.

To illustrate the filtering rules we use the network described in Figure 1, and present the routing policy for protecting AS1, whose adjacent ASes are $40, 300$. Our description uses the Cisco IOS command-line interface. We verified that routers from other vendors (e.g., Juniper Networks) support the same functionality.

We first use AS1's path-end record to create the following access list (named as1), which blacklists routes containing (invalid) links to AS1 from non-adjacent ASes and, in case AS1 is a stub, i.e., does not provide transit services, routes where AS1 is an intermediate hop on the path.

```
// disallow any AS but 40 or 300 to
// advertise a link to AS1
ip as-path access-list as1 deny _[^(40|300)]_1_

// if AS1 is a stub, deny routes where
// AS1 is an intermediate hop
ip as-path access-list as1 deny _1_[0-9]+_
```

The agent creates another access list (named allow-all) to allow all other routes. This access list is global, i.e., created once rather than for every adopting AS.

```
ip as-path access-list allow-all permit
```

Finally we apply these policies in order, i.e., first blocking invalid routes, then allowing all others.

```
route-map Path-End-Validation permit 1
    match ip as-path as1
    match ip as-path allow-all
```

8. RELATED WORK

The security vulnerabilities of today's interdomain routing system motivated many proposals for securing BGP routing. Due to length restrictions, we only discuss below the main proposals for how to *prevent path manipulation attacks*. We refer the reader to the survey in [13] for an extensive discussion, which includes a description of important complementary directions such as *detection* of attacks (e.g., [25, 31, 46])

Past research analyzed various important aspects of proposals for securing interdomain routing against path manipulations, including security guarantees [10, 23] and adoptability [14, 18]. Several proposals focused on using cryptography to prevent route manipulation attacks, including S-BGP [27], psBGP [44], and BGPsec [34]. These proposals require changes to the BGP protocol, and upgrading routers to support online cryptographic computations, which are significant challenges to adoption. In contrast, path-end validation extends RPKI's offline approach, and can be deployed on top of the current Internet infrastructure, with only router configuration changes and offline, off-router cryptography.

Secure-origin BGP (soBGP [45]) was another proposal for securing BGP using offline validation of inter-AS connections. It was proposed to the IETF as a general framework allowing for many possible realizations (from essentially RPKI to variants that require significant changes to

routers and BGP message format), but was abandoned in favor of S-BGP and BGPsec, and its properties were thus not sufficiently analyzed.

Path-end validation is carefully engineered to be easily deployable and to provide significant security benefits even in very limited partial deployment. We regard path-end validation as a specific realization of soBGP designed to realize these objectives. Specifically, guiding path-end validation's design are two insights with important implications for its adoptability: (1) that validating the 1-AS-hop suffix of the BGP path is already very beneficial in terms of performance, and (2) that utilizing RPKI's *offline* approach provides significant benefits for adoption (e.g., avoiding the need to change BGP message format or the decision process at routers, no need to trust intermediate ASes to forward security-related information). Our simulation results establish that path-end validation indeed provides a very attractive "return on investment" in partial adoption.

Subsequent to the publication of our results on path-end validation in [15], a talk at NANOG [38] and several blog posts [39, 40] presented a related effort by Linkedin and several vendors and providers to rethink path validation. We believe that path-end validation is a good direction to explore in this context and that our results can be of value in informing such an effort.

RLP is a recent proposal from the SIDR working group [42] that suggests mitigating route-leaks by annotating hops in BGP advertisements with provider-to-customer and peering links. This allows routers to detect "valley" routes, i.e., BGP path advertisements that violate the Gao-Rexford export condition [17]. Thus, RLP can protect from all route-leak attacks, whereas our solution, described in Section 6, protects only the stubs (over 85% of ASes). However, while our mechanism follows the two design guidelines in Section 2 to simplify deployment, the RLP proposal violates both: it introduces modifications to routers, for attaching annotations to their BGP advertisements, and therefore also changes the BGP message format, which can limit value under partial adoption.

Interestingly, [14] examines path-end validation (termed "first-hop authentication" therein) from a very different angle. Specifically, [14] studies the diffusion of BGP security solutions from an economic/game-theoretic perspective. Our foci, in contrast, are on the *deployability* of path-end validation (e.g., avoiding the need to upgrade/replace BGP routers) and on quantifying its security benefits in partial adoption. Indeed, while [14] mentions online cryptography as a viable approach to realizing path-end validation, this would eliminate all deployability and security benefits discussed here.

Our preliminary workshop paper [15] studied the security guarantees of path-end validation, but did not address (1) the prerequisites to deployment discussed in Section 3, namely, stability and security monotonicity, (2) path-end validation's security benefits when RPKI is only partially deployed, as discussed in Section 5, and (3) path-manipulation attacks beyond next-AS attacks, as discussed in Section 6.

9. CONCLUSION

We presented path-end validation as a means for improving interdomain routing security while avoiding the hurdles en route to deployment of BGPsec. Our security evaluation shows that path-end validation provides a surprisingly high level of security even with a modest number of adopters, and an open-source implementation shows the feasibility of its deployment on top of today's routing infrastructure.

We hence believe that path-end validation provides a tangible path to significant improvements in interdomain routing security in the seemingly very long interim period before BGPsec is fully deployed. Our findings motivate the standardization of path-end validation and its integration into RPKI. We propose that governments and industry groups concentrate regulatory efforts and/or financial incentives on convincing large ISPs in their countries to adopt path-end validation (on top of RPKI).

Importantly, the success of path-end validation, as an extension of RPKI, is dependent on the extent to which RPKI is adopted, especially by the large ISPs. While RPKI is slowly gaining traction amongst network operators, its deployment rate remains slow and only rougly 6% of IP prefixes advertised in BGP are protected by RPKI. Hence, analyzing the root causes for RPKI's sluggish adoption and eliminating obstacles to adoption is an important direction for future research and standardization efforts.

Acknowledgments

This work was supported by ISF grants 420/12 and 1354/11, Israel Ministry of Science grants 3-9772 and 3-10884, the Israeli Center for Research Excellence in Algorithms, and an ERC Starting Grant. We thank Aditya Akella, Steve Bellovin, Randy Bush, Sharon Goldberg, Joel Halpern, Hezi Moriel, and Alvaro Retana for their helpful comments and suggestions.

10. REFERENCES

[1] Hijack Event Today by Indosat. BGPmon.
[2] New Threat: Targeted Internet Traffic Misdirection. Renesys blog.
[3] Routing Hiccup Briefly takes Google Down Worldwide. Thousand Eyes blog.
[4] Spotify Route Leak. Thousand Eyes blog.
[5] Turkey Hijacking IP addresses for popular Global DNS providers. BGPmon.
[6] Pakistan Hijacks YouTube. Renesys Blog, Feb. 2008.
[7] BGP Routing Incidents in 2014, Malicious or Not? http://www.bgpmon.net/bgp-routing-incidents-in-2014-malicious-or-not, 2015. BGPMon.
[8] CAIDA AS Relationships Dataset. http://www.caida.org/data/as-relationships/, Jan. 2016.
[9] Andree Toonk. BGP Hijack Incident by Syrian Telecomunications Establishment. BGPmon, 2015.
[10] H. Ballani, P. Francis, and X. Zhang. A Study of Prefix Hijacking and Interception in the Internet. In *proc. of ACM SIGCOMM*, pages 265–276, 2007.

[11] S. Bellovin, R. Bush, and D. Ward. Security Requirements for BGP Path Validation. RFC 7353 (Informational), Aug. 2014.

[12] R. Bush and R. Austein. The Resource Public Key Infrastructure (RPKI) to Router Protocol. RFC 6810 (Proposed Standard), Jan. 2013.

[13] K. R. B. Butler, T. R. Farley, P. McDaniel, and J. Rexford. A Survey of BGP Security Issues and Solutions. *proc. of the IEEE*, 98(1):100–122, 2010.

[14] H. Chan, D. Dash, A. Perrig, and H. Z. 0001. Modeling Adoptability of Secure BGP Protocols. In L. Rizzo, T. E. Anderson, and N. McKeown, editors, *SIGCOMM*, pages 279–290. ACM, 2006.

[15] A. Cohen, Y. Gilad, A. Herzberg, and M. Schapira. One Hop for RPKI, One Giant Leap for BGP Security. In J. de Oliveira, J. Smith, K. J. Argyraki, and P. Levis, editors, *HotNets*, pages 10:1–10:7. ACM, 2015.

[16] T. Dierks and E. Rescorla. The Transport Layer Security (TLS) Protocol Version 1.2. RFC 5246 (Proposed Standard), Aug. 2008. Updated by RFCs 5746, 5878, 6176, 7465, 7507, 7568, 7627, 7685.

[17] L. Gao and J. Rexford. Stable Internet Routing without Global Coordination. *IEEE/ACM Transactions on Networking*, 9(6):681–692, 2001.

[18] P. Gill, M. Schapira, and S. Goldberg. Let the Market Drive Deployment: A Strategy for Transitioning to BGP Security. In *SIGCOMM*, pages 14–25, 2011.

[19] P. Gill, M. Schapira, and S. Goldberg. Modeling on Quicksand: Dealing with the Scarcity of Ground Truth in Interdomain Routing Data. *Computer Communication Review*, 42(1):40–46, 2012.

[20] V. Giotsas, S. Zhou, M. J. Luckie, and kc claffy. Inferring Multilateral Peering. In K. C. Almeroth, L. Mathy, K. Papagiannaki, and V. Misra, editors, *CoNEXT*, pages 247–258. ACM, 2013.

[21] S. Goldberg. Why is it Taking so Long to Secure Internet Routing? *Commun. ACM*, 57(10):56–63, 2014.

[22] S. Goldberg, M. Schapira, P. Hummon, and J. Rexford. How Secure are Secure Interdomain Routing Protocols. In *SIGCOMM*, pages 87–98, 2010.

[23] S. Goldberg, M. Schapira, P. Hummon, and J. Rexford. How Secure are Secure Interdomain Routing Protocols? *Computer Networks*, 70:260–287, 2014.

[24] G. Huston, R. Loomans, and G. Michaelson. A Profile for Resource Certificate Repository Structure. RFC 6481 (Proposed Standard), Feb. 2012.

[25] J. Karlin, S. Forrest, and J. Rexford. Pretty Good BGP: Improving BGP by Cautiously Adopting Routes. In *ICNP*, pages 290–299. IEEE Computer Society, 2006.

[26] S. Kent and K. Seo. Security Architecture for the Internet Protocol. RFC 4301 (Proposed Standard), Dec. 2005. Updated by RFCs 6040, 7619.

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

[28] N. Kephart. Route Leak Causes Amazon and AWS Outage. Thousand Eyes blog, 2015.

[29] X. Lee, X. Liu, Z. Yan, G. Geng, and Y. Fu. RPKI Deployment Considerations: Problem Analysis and Alternative Solutions. Internet Draft, Jan. 2016.

[30] M. Lepinski and S. Kent. An Infrastructure to Support Secure Internet Routing. RFC 6480 (Informational), Feb. 2012.

[31] J. Li, T. Ehrenkranz, and P. Elliott. Buddyguard: A Buddy System for Fast and Reliable Detection of IP Prefix Anomalies. In *ICNP*, pages 1–10. IEEE, 2012.

[32] R. Lychev, S. Goldberg, and M. Schapira. Brief Announcement: Network-Destabilizing Attacks. In D. Kowalski and A. Panconesi, editors, *PODC*, pages 331–332. ACM, 2012.

[33] R. Lychev, S. Goldberg, and M. Schapira. BGP Security in Partial Deployment: Is the Juice worth the Squeeze? In *SIGCOMM*, pages 171–182. ACM, 2013.

[34] E. M. Lepinski. BGPsec Protocol Specification. RFC 1, Oct. 2014.

[35] Mirjam Kuhne. AS Path Lengths Over Time. https://labs.ripe.net/Members/mirjam/update-on-as-path-lengths-over-time, 2012.

[36] P. Mohapatra, J. Scudder, D. Ward, R. Bush, and R. Austein. BGP Prefix Origin Validation. RFC 6811 (Proposed Standard), Jan. 2013.

[37] NIST. RPKI Monitor. http://rpki-monitor.antd.nist.gov/, 2016.

[38] Russ White. Rethinking Path Validation. NANOG 66, Feb. 2016.

[39] Russ White. Rethinking Path Validation: Pt. 1, New Requirements. LinkedIn Engineering Blog, https://engineering.linkedin.com/blog/2016/03/rethinking-path-valid-pt1, Mar. 2016.

[40] Russ White. Rethinking Path Validation: Pt. 2. LinkedIn Engineering Blog, https://engineering.linkedin.com/blog/2016/03/rethinking-path-validation--pt--2, Mar. 2016.

[41] K. Sriram. BGPSEC Design Choices and Summary of Supporting Discussions. Internet draft, https://tools.ietf.org/html/draft-sriram-bgpsec-design-choices-08, July 2015.

[42] K. Sriram, D. Montgomery, B. Dickson, K. Patel, and A. Robachevsky. Routing Hiccup Briefly takes Google Down Worldwide. Internet Draft.

[43] R. Steenbergen. PeeringDB. http://www.peeringdb.com/, July 2015.

[44] P. C. van Oorschot, T. Wan, and E. Kranakis. On Interdomain Routing Security and Pretty Secure BGP (psBGP). *ACM Trans. Inf. Syst. Secur*, 10(3), 2007.

[45] R. White. Deployment Considerations for Secure Origin BGP (soBGP), June 2003.

[46] K. Zhang, A. Yen, X. Zhao, D. Massey, S. F. Wu, and L. Z. 0001. On Detection of Anomalous Routing Dynamics in BGP. In *NETWORKING*, volume 3042 of *LNCS*, pages 259–270. Springer, 2004.