BGP-iSec: Improved Security of Internet Routing Against Post-ROV Attacks

Cameron Morris
University of Connecticut
cameron.morris@uconn.edu

Amir Herzberg
University of Connecticut
amir.herzberg@gmail.com

Bing Wang
University of Connecticut
bing@uconn.edu

Samuel Secondo
University of Connecticut
samuel.secondo@uconn.edu

***Abstract*—We present BGP-iSec, an enhancement of the BGPsec protocol for securing BGP, the Internet's inter-domain routing protocol. BGP-iSec ensures additional and stronger security properties, compared to BGPsec, without significant extra overhead. The main improvements are: (i) *Security for partial adoption:* BGP-iSec provides significant security benefits for early adopters, in contrast to BGPsec, which requires universal adoption. (ii) *Defense against route leakage:* BGP-iSec defends against route leakage, a common cause of misrouting that is not prevented by BGPsec. (iii) *Integrity of attributes:* BGP-iSec ensures the integrity of *integrity-protected attributes*, thereby preventing announcement manipulation attacks not prevented by BGPsec. We argue that BGP-iSec achieves these goals using extensive simulations as well as security analysis. The BGP-iSec design conforms, where possible, with the BGPsec design, modifying it only where necessary to improve security or ease deployment. By providing stronger security guarantees, especially for partial adoption, we hope BGP-iSec will be a step towards finally protecting inter-domain routing, which remains, for many years, a vulnerability of the Internet's infrastructure.**

I. Introduction

The Border Gateway Protocol (BGP), now in its 4th version [1], is the primary method used by Internet Service Providers (ISPs) to exchange routing information in the Internet. Security was not a design goal of BGP. Until BGP-4 [1], the standard merely stated that 'Security issues are not discussed', and even BGP-4 mostly refers to an informational RFC [2] for analysis of BGP vulnerabilities. BGP vulnerabilities have been a known serious concern at least since 1989 [3], and yet, only partial defenses have been standardized and deployed so far.

Currently, the most impactful misrouting incidents are *prefix/sub-prefix hijacks* [4] and *route leaks* [5]. In prefix/sub-prefix hijack, the attacker falsely originates a route to a prefix it is not authorized to announce, while in a route leak, a rogue transit-service provider exports an announcement which conflicts with its supposed business model. Such attacks can result in major disruptions and improper interception of traffic; e.g., a sub-prefix hijack can intercept nearly 100% of the sub-prefix's traffic, unless proper defenses are deployed, since the most-specific route preference in IP routing will cause routers to forward to the attacker all traffic destined for the hijacked sub-prefix.

In recent years, there has been considerable progress with standardizing and deploying defenses against prefix/sub-prefix hijacks, mainly based on the Resource Public Key Infrastructure (RPKI) [6]. RPKI allows the owner of prefix p to identify authorized origin Autonomous Systems (ASes), using a signed *Route Origin Authorization (ROA)*. BGP routers receiving announcement of p or a sub-prefix of p can perform *Route Origin Validation (ROV)*, validating that the origin of the announcement was authorized by a (valid) ROA and dropping announcements that have invalid origins or prefix length. Adoption of ROAs and ROV has been steadily increasing [7]–[9]. Currently, over 42% of IPv4 address space is protected by ROAs [7], and measurements show a steady increase in the number of ASes applying ROV to filter announcements with invalid ROAs with some estimating as many as 37% of ASes now filter [9]–[15]. Recently proposed extensions to ROV [16] can significantly improve the defense against sub-prefix and prefix hijacks under partial adoption.

The increasing adoption of RPKI/ROV will make prefix and sub-prefix hijacks less effective. As a result, attackers will resort to *post-ROV attacks*, i.e., attacks that ROV does not defend against. In this paper, we present the *BGP-iSec* protocol that has significantly ***i***mproved security over *BGPsec* [17]. BGP-iSec protects against three types of post-ROV attacks: *route leaks*, *path manipulations* and *attribute manipulations*.

Route leaks can be accidental or intentional. RFC 9234 [18] defines the Only-To-Customer (OTC) mechanism against accidental route leaks; a currently developed draft, ASPA [19] should protect also against some intentional route leaks.

Path manipulations involve sending announcements with valid origin, but which were *not* relayed along the path of the ASes indicated in the announcement, as per the BGP specifications. Path manipulation can be abused in different ways, most notably, to *intercept* traffic sent to a victim destination, obtain *Man-in-the-Middle (MitM)* capabilities for traffic sent to a victim destination, perform Denial-of-Service (DoS) attacks, or make *stealthy* (hard to detect) attacks.

Attribute manipulations. The attributes in BGP announcements can have a significant impact on routing, and can be abused in different attacks. For example, a rogue AS may remove the Only-To-Customer (OTC) anti-leakage attribute [18] to cause route leaks, or add a fake OTC attribute to

Network and Distributed System Security (NDSS) Symposium 2024
26 February - 1 March 2024, San Diego, CA, USA
ISBN 1-891562-93-2
https://dx.doi.org/10.14722/ndss.2024.241035
www.ndss-symposium.org

cause *stealthy disconnections*. See other attribute manipulation attacks in [20], [21].

While BGPsec is the IETF standardized protection against path manipulation attacks, it is not deployed. Deployment of BGPsec faces two formidable obstacles. The first obstacle is that *BGPsec has high computational requirements* that necessitate cryptographic co-processors, even with multiple proposed optimizations [22]–[25]. The second obstacle is that simulations of BGPsec show only *limited benefits in partial adoption* [26]–[29]. Even under full adoption, BGPsec does not prevent route leaks and attribute manipulation. Both obstacles were mentioned in responses to the recent FCC inquiry into Internet routing vulnerabilities [30], e.g., from Cisco and Juniper [31], [32].

We design BGP-iSec to address only the second obstacle, i.e., improving security benefits, especially under partial deployment. This obstacle is more urgent than the obstacle of high computational requirements for two reasons. First, techniques have been developed to address the high computation requirements, by designing router hardware and software to efficiently support BGPsec [23], [25], [33], and practitioners have expressed increasing optimism about the feasibility of meeting BGPsec's performance requirements, e.g., [34]. Second, the limited benefits from partial deployment of BGPsec make early-adopters unlikely; and since route leaks and attribute manipulations are possible even with full adoption of BGPsec, the incentive to adopt would remain limited.

Interoperability and reuse. BGP-iSec is fully interoperable with BGP, namely, its implementation uses standard BGP attributes, and we believe that it will not require changes to the basic BGP processing. We hope that this will allow to implement BGP-iSec as an extension of BGP, unlike solutions that change the protocol processing, including BGPsec and [35]–[39]. On the other hand, the design of BGP-iSec reuses, where possible, elements from the BGPsec design, allowing us to take advantage of the significant efforts of standardizing, optimizing and developing implementations of BGPsec [23], [25], [40]. In §IV-F, we show that BGP-iSec has similar computation complexity as BGPsec for high (over 50%) adoption rate.

MAIN CONTRIBUTIONS:

• **BGP-iSec**, a security extension to BGP which is based on BGPsec but with much improved security, especially under partial adoption. BGP-iSec effectively prevents path manipulations, announcement manipulations, and route leaks. BGP-iSec has comparable efficiency to BGPsec, is interoperability with BGP, and reuses BGPsec mechanisms when possible.

• **Revisiting transitive signatures to defend against path and attribute manipulations (§III).** BGP-iSec revisits the use of transitive signatures, proposed already in S-BGP [41], but abandoned by BGPsec[1]. The design of BGP-iSec builds upon RPKI, and contains important aspects not in S-BGP, including mechanisms that address the concerns that led to BGPsec's abandonment of transitive signatures. Using simulations and analysis, we show, for the first time, the dramatic loss in security due to the use of non-transitive signatures, which results in the limited security of BGPsec in partial adoption.

[1]BGPsec signatures are sent as attributes with the transitive bit set to zero, thus are non-transitive even though the signatures are passed along to other neighbors running BPGsec as if they were transitive.

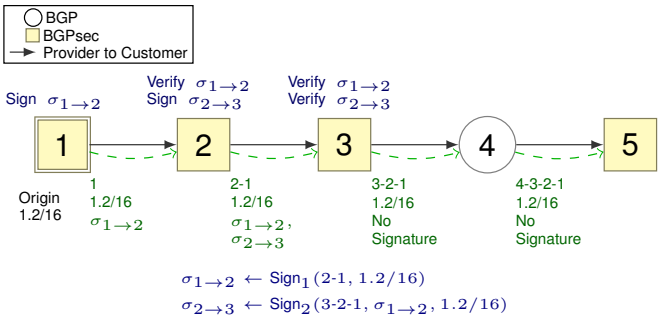


Fig. 1: Illustration of signing, verification and announcement update in BGPsec.

• **Effective defenses against route leaks (§III).** BGP-iSec deploys three effective defenses against route leaks, *Protected-OTC, UP attribute*, and *ProConID*. Only the first defense, Protected-OTC, was proposed (in a similar form) earlier [42], [43]; the other two are novel defenses that we develop. These three defenses are complementary to each other, providing different tradeoffs in security and complexity.

• **Experimental security evaluation (§IV).** We evaluate BGP-iSec using extensive simulations over an empirical Internet topology [44], against two strong attacker models (Global Attacker and Full Attacker) and different attacker strategies, showing the significant benefits of BGP-iSec. For example, with BGP-iSec, the attacker interception rate declines rapidly (from about 27% at no adoption, to about 3% at 50% adoption), while with BGPsec, even with 50% adoption, interception rate remains near 27%. Above 80% adoption, interception rates for BGP-iSec are negligible, while even with 99% adoption, interception rate for BGPsec is about 22%.

• **Security analysis (§V).** We prove that BGP-iSec has several security properties, including announcement integrity, preventing route leaks, and no false positives under strong attacker models.

While the results of BGP-iSec are very encouraging, more evaluation, design, and efforts from the community are needed to ensure we have effective, deployable defenses against path manipulations and intentional route leaks (§VII).

II. BACKGROUND AND MODELS

A. Background: BGPsec

We briefly review the functionalities in BGPsec that are related to this paper; more details are found in [17]. Suppose that origin X_1 is the owner of prefix p and sends an announcement to neighbor X_2, which forwards it to X_3, and so on until X_n. Let us first consider the full deployment scenario where all the ASes adopt BGPsec. Let $\sigma_{i\to i+1}$ denote X_i's signature for its announcement to X_{i+1}. The announcement from X_i to X_{i+1} is signed by X_i using its private key over the following attributes: AS numbers (ASNs) from the origin up to itself, i.e., $X_1, \ldots, X_i$, the next AS X_{i+1}, and the signatures $\sigma_{1\to2}, \ldots, \sigma_{i-1\to i}$, and p. This signing process is illustrated in Fig. 1, where for simplicity, we set $\sigma_{i\to i+1} = \text{Sign}_{X_i}(X_{i+1} - X_i - \cdots - X_1, \sigma_{1\to2}, \ldots, \sigma_{i-1\to i}, p)$. AS X_{i+1}

declares that the announcement it receives from X_i as valid if and only if all the signatures in the announcement are valid and the (origin, prefix) pair is valid based on RPKI.

During partial deployment, when an adopting AS sends an announcement to a non-adopting neighbor, it *downgrades* to regular BGP, i.e, does not send the signatures [45]. For instance, in Fig. 1, AS 3 does not include any signature in its announcement to non-adopting AS 4. Consequently, as specified in [46] (section 3.2) and [17] (section 7.9), and justified in [45] (section 6), BGPsec verification is only for announcements sent within a 'BGPsec deployment island', i.e., a contiguous group of ASes that all deploy BGPsec.

To summarize, the signatures in BGPsec are *non-transitive* attributes, and are not relayed to non-adopting ASes. In addition, BGPsec does not address route leaks. Furthermore, it only verifies 'the authenticity of the AS path info received in a BGP update' [47]; other attributes are not protected or verified, even within BGPsec deployment islands.

B. Routing Model

1) Valley-free routing model: To define and analyze route leaks, and to perform experimental evaluation, we assume *valley-free routing* [48], as in other studies [10], [26], [49], [50]. Specifically, we model the Internet as an AS-Graph, where mutually agreed upon inter-AS relationships can be characterized either as *customer-provider*, where the customer pays its provider for the transit of traffic, or as *peer-to-peer*, where traffic is exchanged between the two ASes without monetary compensation[2]. In this model, a benign AS never relays the announcements that it received from non-customer ASes to non-customer neighbors, hence the name 'valley-free'.

While routing is not always valley-free in practice [51]–[54], RFC 7908 [5] defines six types of route leaks, only two are not in the form of violating the valley-free model, and both can be prevented by RPKI/ROV, not relevant to post-ROV security that we focus on in this paper. Therefore, we adopt the valley-free model when designing defenses against route leaks. The cases where violating the valley-free model are not route leaks need special treatment, which is left as future work.

2) Path-selection and export policies: In BGP, each AS has a *path-selection policy* that selects the best path to use for each IP prefix, and an *export policy* that determines what routes (if any) to forward to a neighbor.

The announcements that BGP-iSec detects as invalid are discarded, regardless of the path-selection policy. For the announcements that are not discarded, we consider two policies, *security-third* as in [26], and *security-never*. Security-never is easy to implement—it has no further consideration of security, and simply follows Gao-Rexford model [48] with the following rules. First, an AS prefers paths from customers, then from peers, and lastly providers. That is, 'relationship first' or 'local preference first'. Second, if two paths have the same relationship, e.g., both are from customer, peer or provider, then the AS prefers the shorter path. That is, length second. Third, break ties. Security-third policy differs from the above in the third rule: if two paths have the same relationship and length, a benign adopting AS prefers the path where all ASes are adopting BGP-iSec, and hence the name 'security-third'. Note that partial secure paths, i.e., where only some of the ASes adopt BGP-iSec, are *not* preferred; see §III-A.

Our evaluation in §IV shows that security-never achieves similar performance as security-third, indicating the primary benefits come from discarding invalid announcements that is in both policies. In addition, security-never is much easier to implement than security-third (see discussion in Appendix C-A). We therefore recommend security-never, instead of security-third, in practice.

As to export policy, only in our simulations, we adopt the widely-used and simplifying *export-to-all* policy. Namely, for an AS, the preferred announcements are sent to *all customers*; and if the preferred announcement for a prefix was received from a customer, then it is sent to *all neighbors*, including providers and peers.

C. Known Adoption and Public Keys (KAPK) Assumption

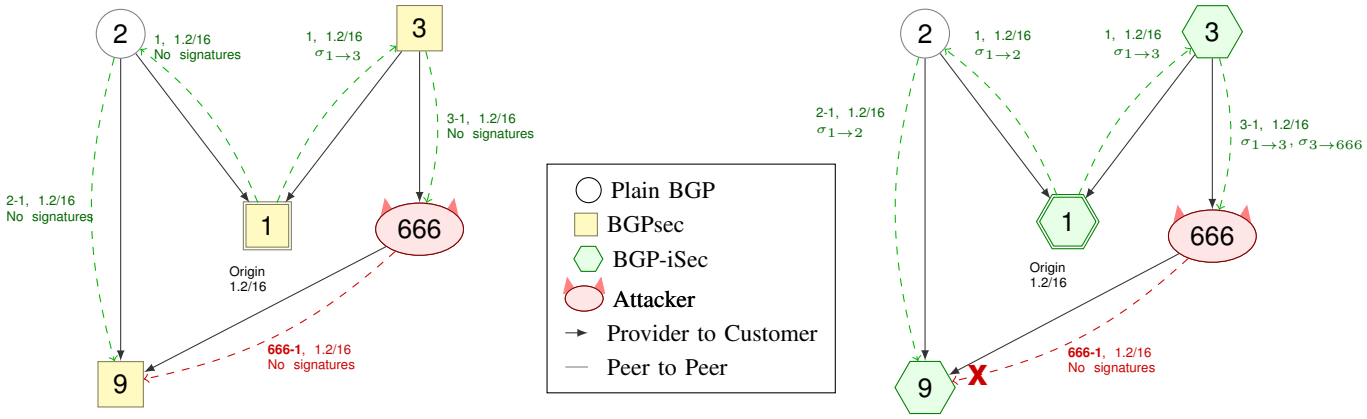
BGP-iSec, like BGPsec and the ROV standard [55], relies on RPKI [6]. RPKI is a public key infrastructure (PKI) designed to support improved security of Internet routing, by defining relevant public-key certificates and other signed objects, as well as *RPKI distribution points* that facilitate distribution of the certificates and signed objects, and related protocols.

Specifically, BGP-iSec uses BGPsec router certificates, defined in [56]. The BGPsec router certificates of an AS define the public key associated with the routers of that AS. When an AS, say X, adopts BGP-iSec, then X will issue new BGPsec router certificates that will also indicate that X supports BGP-iSec (from all its routers), and distribute these certificates via the RPKI repositories. BGP-iSec-adopting ASes, like other RPKI-deploying ASes, periodically download updated versions of the RPKI repositories [6], [57]. This ensures that each BGP-iSec-adopting AS will know all other BGP-iSec-adopting ASes and their public keys, soon after adoption. Hence, we assume the ASes that adopt BGP-iSec and their public keys are known, referred to as *Known Adoption and Public Keys (KAPK) assumption*. This assumption is equivalent to the assumptions made by other mechanisms using the RPKI, e.g., the already significantly deployed ROV mechanism [55], which depends on timely knowledge of new ROAs and certificates.

One convenient way to signal support for BGP-iSec is by including an additional KeyPurposeID value in the certificate's *Extended Key Usage* extension. This KeyPurposeID is in addition to or instead of the BGPsec KeyPurposeID defined in Section 3.1.3.2 in [56]. In this way, BGP-iSec can take advantage of the existing RPKI distribution points, which will allow efficient distribution of BGP-iSec certificates, including identification of BGP-iSec-adopting ASes.

Recent works [15], [58]–[62] pointed out vulnerabilities in the current RPKI. However, countermeasures were proposed, and some have already been implemented. Therefore, we believe that BGP-iSec can securely use RPKI and its distribution mechanisms, and hence KAPK assumption holds. In §IV-H, we investigate the impact when KAPK assumption is violated and show that it does not need to be strictly satisfied.

[2]The model ignores other relationships, e.g., siblings.



$\sigma_{1\to3} \leftarrow \mathsf{Sign}_1(3\text{-}1, 1.2/16)$

(a) BGPsec.

$\sigma_{1\to2} \leftarrow \mathsf{Sign}_1(2\text{-}1, 1.2/16)$
$\sigma_{1\to3} \leftarrow \mathsf{Sign}_1(3\text{-}1, 1.2/16)$

(b) BGP-iSec, transitive signature.

Fig. 2: Example of path manipulation attacks in partial adoption, BGPsec vs. BGP-iSec. (a) BGPsec. AS 666 does not adopt BGPsec, so AS 3 will send it a BGP announcement without any signatures. AS 9 cannot detect that the announcement from 666 is fake and may choose the fake path via 666, falling victim to the hijack. (b) BGP-iSec. AS 666 attempts to remove AS 3 from the AS-path it announces to AS 9. However, the missing signature is observed by AS 9 and the announcement is rejected as invalid.

D. Attacker Models

In the design, analysis and evaluation of BGP-iSec, we consider three attacker models: *Global Attacker*, *Full Attacker*, and *MitM*. All of them are models for strong attackers. In particular, in all three models, the attacker is given the global routing topology and the relationships between ASes, e.g., provider, customer, or peer.

The MitM attacker is the strongest model. It gives the adversary complete control over the communication between all ASes: the attacker can intercept, modify, block and impersonate BGP announcements. Theorem 1 (§V) shows that BGP-iSec ensures important security properties even against the MitM attacker.

However, the MitM attacker model is too strong for evaluating the performance of routing protocols. First, in practice, MitM capabilities are rarely available to attackers against inter-domain routing. Specifically, RFC 7132 [47] mandates authentication mechanisms between neighboring BGP routers, using IPsec [63], TLS or TCP Authentication Option [64]. Second, a MitM attacker can drop legitimate announcements, making it trivial to disconnect the legitimate origin, and giving the attacker an unreasonable edge in intercepting traffic. Indeed, if one really assumes such strong MitM capabilities, the attacker can directly attack traffic in the data plane, and may not even need to manipulate routing.

Therefore, in our simulations, we evaluate BGP-iSec against the Full Attacker and Global Attacker models. Both of them receive *all BGP announcements sent by any AS*. However, the Full Attacker receives the BGP announcements with *all attributes*, while the Global Attacker receives the BGP announcements with *all attributes except the BGP-iSec attributes*. In practice, attackers rarely have direct eavesdropping capabilities on announcements exchanged between a pair of directly connected ASes. Instead, attackers may obtain (some) routing information shared by (some) ASes. In particular, the public BGP collectors such as RouteViews [65] and RIPE RIS [66] expose BGP announcements from a subset of ASes, motivating the Full Attacker model. However, we found that many (probably most) of these BGP collectors only expose limited set of attributes, hence will not expose the BGP-iSec attributes, motivating the Global Attacker model. Note that both models allow access to announcements from *all* ASes, while in reality, only some ASes expose their routing information (via public BGP collectors or other mechanisms). Hence, the attack success rates obtained from our simulations are likely higher than the rates expected in practice.

III. BGP-iSec Design

In §III-A, we discuss *mandatory signatures* to defend against path and attribute manipulations. In §III-B, we discuss *route-leak prevention defenses* to prevent benign and malicious route leaks. We also designed and evaluated a mechanism for preventing path length shortening (see Appendix A), but its improvement turned out to be insufficient and hence it is not included in BGP-iSec.

A. Mandatory Signatures for Announcement Integrity

BGP-iSec ASes sign and verify every announcement whose origin adopts BGP-iSec. In addition, the signatures are *transitive attributes*: a BGP-iSec-adopting AS exports an announcement to *both* adopting and not adopting neighbors (following export policy). This is in contrast to BGPsec, which includes signatures only when exporting to a BGPsec-adopting AS, making it vulnerable to downgrades. As an example, Fig. 2a shows that in BGPsec, AS 666 sends to AS 9 a manipulated announcement, with no BGPsec signatures and with a fake AS-path (666-1); AS 9 prefers this shorter, unsigned path, and its traffic is hijacked by AS 666. In contrast, Fig. 2b shows that in BGP-iSec, non-adopting AS 2 propagates the signature $\sigma_{1\to2}$ as part of the announcement it sends to AS 9, allowing AS 9 to verify the partial path from AS 1 to AS 2 in the announcement, although AS 2 is non-adopting. Similarly, AS 9 detects that the announcement it receives from AS 666 is invalid, since it does *not* contain a valid signature $\sigma_{1\to666}$.

As a result, AS 9 accepts the announcement from AS 2 and rejects the announcement from AS 666, and hence does not fall victim to the path manipulation by AS 666.

BGP-iSec uses the RPKI repository to identify adopting ASes and their public keys following the KAPK assumption (see §II-C). Signatures are *mandatory*; an announcement with a BGP-iSec-adopting AS is considered invalid, unless it includes a valid signature by all BGP-iSec-adopting ASes in the AS-path. We confirmed that BGP-iSec's effectiveness is robust to reasonable failures of the KAPK assumption; see §IV-H. While S-BGP [67] also uses transitive signatures, it predates RPKI and does not discard announcements 'missing' signatures.

As in BGPsec, BGP-iSec does *not* prefer announcements based on the number of ASes signing (or not signing). BGP-iSec only instructs ASes to discard announcements where some BGP-iSec-adopting AS did not sign properly. BGP-iSec-adopting ASes *may* give preference to fully-signed paths, e.g., as in security third' [26], which however does not provide much benefits, as shown by our experimental results (see §IV).

Note that the BGP specifications [1], [68] instruct ASes to propagate, without modifications, transitive attributes when they export announcements. In a recent study [69], about 2% of the ASes seem to have dropped an unknown transitive attribute. Based on the above measurement results, we assume that only a small fraction of the ASes drop BGP-iSec transitive signatures before forwarding announcements. In §IV-G, we confirm that reasonable percents of non-compliant ASes corrupting the attribute will not significantly reduce the impact of BGP-iSec.

Alternative: out-of-band delivery of signatures. BGP-iSec can also be deployed by sending signatures out-of-band, e.g., over HTTP, rather than as transitive attributes. This can be used if a significant fraction of the ASes fail to forward the BGP-iSec transitive attributes, a concern raised in [45], which motivated abandoning transitive in BGPsec. The retrieval mechanism (URI) can be specified in the RPKI signed object associated with the origin AS (as in §II-C).

Secure downgrade. The other concern with mandatory signatures raised in [45] is that an adopting AS may sometimes be unable to sign due to the computational load [45]. If this would indeed be a concern, we can allow an adopting AS X to perform a secure downgrade, which signals that AS X stopped signing announcements (typically, due to computational load), as follows. Every AS will include in its BGP-iSec certificate a *downgrade ticket* $h(x)$, where h is a one-way hash function and x is a random *downgrade preimage*. To stop signing, an AS appends its downgrade preimage x to its announcements, as a transitive attribute. Each adopting AS that receives the attribute containing x will apply h to confirm that x is the correct preimage of $h(x)$; (only) when this holds, the announcement will be accepted even without X's signature.

Integrity-protected attributes. There is a challenge in using signatures to protect announcements: the announcements are modified as they are relayed (exported) by ASes, and a signature can only validate the exact string which was signed. BGPsec deals with this challenge by signing only the AS-path attribute (and the identity of the next-AS). The AS-path attribute and the next-AS also change when exporting the announcement, but given the AS-path in an announcement A received by AS Y, it is easy to compute the AS-path and next-AS of the announcement when purportedly forwarded by any AS X in the AS-path, allowing BGPsec in AS Y to validate a signature by AS X over these values.

BGP-iSec generalizes this approach of processing the attributes as received (e.g., by AS Y) to recover the contents of these attributes when the announcement was exported and signed by previous BGP-iSec-adopting ASes along the path (e.g., X); we say that such transitive attributes are integrity-protected. In addition to protecting AS-path (and next-AS) as in BGPsec, integrity-protected attributes include the *OTC* and *UP attributes*, used by BGP-iSec against route leaks (see §III-B). Other attributes can also be integrity-protected; e.g., *fixed attributes*, i.e., transitive attributes that are set only by the origin and not modified by (benign) exporting ASes.

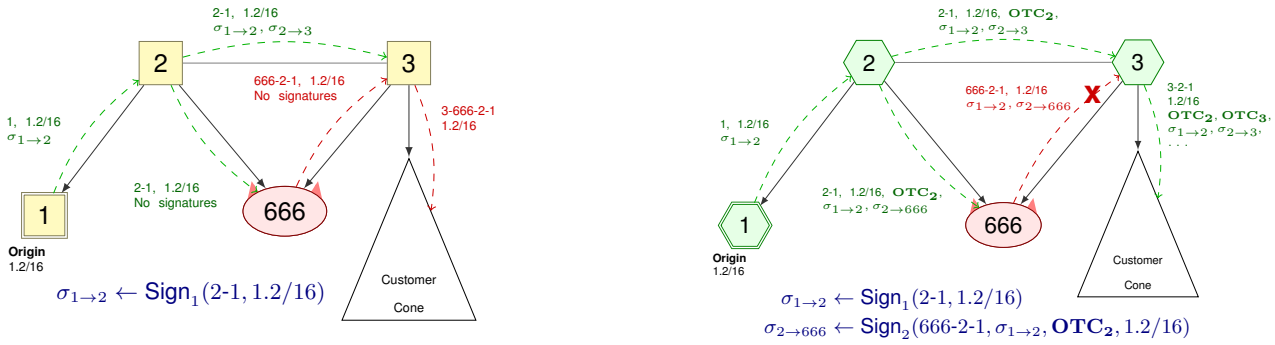
When exporting an announcement A, a BGP-iSec-adopting AS X adds to A a transitive attribute containing its signature σ_X, using its (private) signing key, over a string denoted $TBSby_X(A)$, which represents the string *to-be-signed (TBS) by* AS X when exporting A . $TBSby_X(A)$ encodes all pairs $(\theta, A[\theta])$, where θ is a integrity-protected attribute and $A[\theta]$ is the value of the θ attribute in announcement A, in alphabetic order of the attribute names. A BGP-iSec-adopting AS Y that receives announcement A computes $TBSby_X(A)$ for every BGP-iSec-adopting AS X in A's AS-path, and validates that A contains a signature σ_X by AS X over $TBSby_X(A)$. AS Y computes $TBSby_X(A)$ as the (alphabetically-ordered) sequence of pairs $\{(\theta, revert^Y_\theta(X, A))\}$, for all integrity-protected attributes θ in A.

Intuitively, $revert^Y_\theta(X, A)$ outputs the value of attribute θ as sent by AS X, or $\perp$ if θ was only added by an AS after X, or if θ is not integrity-protected. We define $revert^Y_\theta(X, A)$ in Appendix §D for AS-path, OTC, UP and fixed attributes. It can be extended for other integrity-protected attributes.

In Theorem 1 we show that BGP-iSec ensures announcement integrity, which we define as the integrity of integrity-protected attributes.

Deployment considerations. When an AS is deploying BGP-iSec, it may add signatures to announcements gradually, to ease deployment and/or identify problems. The AS would publish its public keys to the RPKI and signal that it is adopting only after it has completed deployment of BGP-iSec-signing and confirmed that the signatures propagate correctly, avoiding dropping of announcements before deployment is complete.

Protecting communities? It may also be desirable to protect security-sensitive communities, which will defend against the attacks of [20], [21]. However, compared to transitive path attributes, communities are more likely to be benignly dropped by intermediate ASes, since communities are often implemented by network operators in their operator-policy code, while transitive attributes are implemented by router vendors in the operating system [70]. A protected community that is dropped will invalidate the signature (and hence the announcement). We may protect a community by copying it into a transitive attribute, allowing recovery of the community when the corresponding attribute is intact. The attacker can still remove or corrupt the attribute, but this will invalidate the announcement (like for other integrity-protected attributes).



(a) BGPsec.

(b) BGP-iSec, Protected-OTC.

Fig. 3: Example route leaks, BGPsec vs BGP-iSec. (a) BGPsec. AS 666 does not adopt BGPsec, and relays an announcement that it receives from its provider AS 2 to another provider, AS 3, which is not detected by BGPsec. Even if AS 666 did adopt BGPsec, it could still leak a signed and valid path to AS 3, which would not be stopped by BGPsec. (b) BGP-iSec. AS 666 leaks an announcement it received from its provider AS 2 to its other provider AS 3, violating valley-free routing. In the figure, AS 666 relays the announcement without the protected-OTC attribute ($\mathbf{OTC_2}$); hence, the signature is invalid, and AS 3 discards the announcement. If, instead, AS 666 retained the OTC attribute, then AS 3 would discard the announcement due to receiving an OTC announcement from a customer (AS 666). If AS 666 sent the announcement without the signature, then AS 3 would discard it since it knows AS 2 is adopting and detects the missing signature.

B. Defenses Against Route Leaks

Preventing route leaks is not one of the design goals of BGPsec [47]. For example, Fig. 3a shows AS 666 leaking to its provider, AS 3, the announcement that AS 666 received from its (other) provider, AS 2. AS 3 prefers this route-leak announcement over the announcement it receives directly from AS 2, since AS 666 is a customer ('relationship first, length second'). As a result, AS 3's traffic is hijacked by AS 666.

To prevent route leaks, BGP-iSec extends a recent method from the IETF, *Only-To-Customer (OTC)* transitive attribute [18], which signals that an announcement should only be sent 'down', i.e., to customers[3]. The value of the OTC attribute is an ASN X in the AS-path, which is a provider or peer of the following AS in the AS-path, and therefore, following X, the announcement should be exported *only to customers* (see Appendix §B). The OTC attribute does not provide authentication, so does not suffice to protect against a malicious AS intentionally leaking a route—an attacker can simply remove the OTC and the leak will go undetected.

We designed and evaluated three complementing defenses against route-leakage for BGP-iSec: *Protected-OTC*, hash-based *UP attributes*, and *ProConID*. Protected-OTC is simply an integrity-protected variant of OTC; the other two defenses are novel. As we shall see, the UP attributes significantly enhances the security of Protected-OTC in the Global Attacker model, while ProConID further improves the security of the other two mechanisms in the Full Attacker model, at the cost of additional complexity and overhead.

1) Protected-OTC: The Protected-OTC mechanism simply signs the OTC attribute to protect it (see Appendix B). Fig. 3b shows one example, where since AS 2 signs $\mathbf{OTC_2}$ and AS 2 is the next-hop AS of AS 1, AS 666 cannot remove AS 2 or $\mathbf{OTC_2}$. As a result, AS 3 discards the announcement with $\mathbf{OTC_2}$ from AS 666, preventing the route leak.

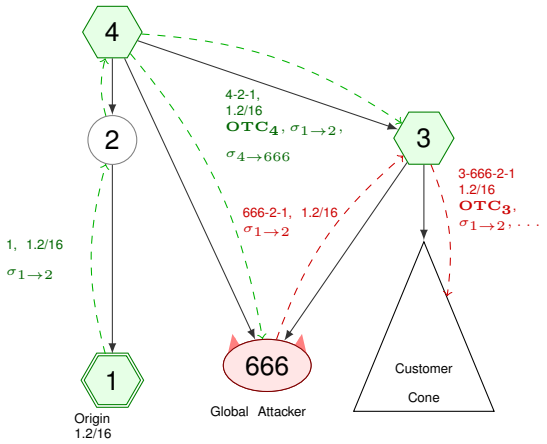
The above mechanism is similar to those proposed in [42], [43], which use a protected flag to mark an announcement being sent to a peer or customer, thus allowing the subsequent receivers of the announcement to check if it follows a valid path. However, these two studies did not quantify the effectiveness of the above in partial deployment as we do. We also design additional route leak defenses below, and our results show that these additional defenses significantly improve the defense against route leaks.

2) The UP Attributes: In partial deployment, the above Protected-OTC mechanism will not always suffice. Consider the scenario in Fig. 4a where AS 666 receives from AS 4 an announcement with two signatures, $\sigma_{1\to2}$ and $\sigma_{4\to666}$, the latter covering a signed OTC attribute. AS 666 can 'remove' AS 4 from the AS-path, allowing it to remove the OTC attribute, and to claim to have received the announcement directly from the non-adopting AS 2. Note this attack works even though AS 4 adopts BGP-iSec as described so far.

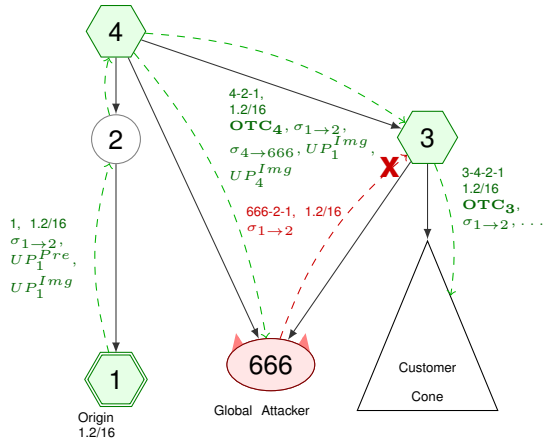
To reduce the attacker's ability to remove the OTC attribute, BGP-iSec includes another defense: the *Up-Permitted (UP) attribute*. This defense relies on the difficulty of guessing preimages of one-way hash functions. It also assumes that the BGP-iSec attributes (in particular, the UP attribute) are not exposed to the attacker, which holds in the Global Attacker model, but not in the MitM and Full Attacker models. Indeed, our evaluations (§IV) show that the UP attributes only contribute to security under the Global Attacker model, not the Full Attacker model. In §III-B3, we present the *ProConID* mechanism, which provides defense against the Full Attacker.

The UP attribute works as follows. Consider an adopting AS X that exports an announcement A to a provider; namely, A may be sent 'up', or is 'Up-Permitted' (UP). Then X adds to A two transitive attributes: *UP-image* and *UP-preimage*. The UP-preimage attribute contains a randomly chosen preimage UP_X^{Pre} for the one-way hash function h, and the UP-

[3]A method similar to OTC is *Down-Only (DO)* community [70]. As explained in [70], OTC attribute and DO community present different tradeoffs: OTC attributes are less likely to be dropped but require a software upgrade in the router OS, which may delay adoption; communities do not require a router upgrade, but are more likely to be dropped. We only consider OTC attribute in this paper since DO community is more likely to be dropped.



(a) Route leak prevention, without the hash-based UP attributes.



(b) With the hash-based UP attributes.

Fig. 4: Illustration of route leak prevention with Protected-OTC, without and with hash-based UP attributes in (a) and (b), respectively. (a) Without UP attributes. Since AS 2 is non-adopting, this allows AS 666 to announce the path 666-2-1, removing the signature and attributes added by AS 4, including the OTC attribute OTC_4, and evading detection by AS 3. For clarity, path shortening (PS) attributes are omitted. (b) With the (signed) UP^{Img} attributes. AS 666 can remove the signature and attributes added by AS 4, including the OTC attribute OTC_4, but cannot remove UP_1^{Img} or find UP_1^{Pre}, so AS 3 detects the route leak. Note this only defends against the Global Attacker; the Full Attacker will know the value of the UP_1^{Pre} attribute.

image attribute UP_X^{Img} contains $h(UP_X^{Pre})$. The UP-image attribute is included in the contents signed by the adopting ASes when adding or relaying it. The UP-preimage is not signed, and is *removed* whenever an adopting AS exports the announcement to a customer or a peer. Therefore, an adopting AS that receives an announcement containing a signed UP-image attribute UP_X^{Img} but without the correct UP-preimage UP_X^{Pre} will consider the announcement as 'down-only'. If this AS exports this announcement, it will add the OTC attribute to signal that it can only be forwarded to customers subsequently.

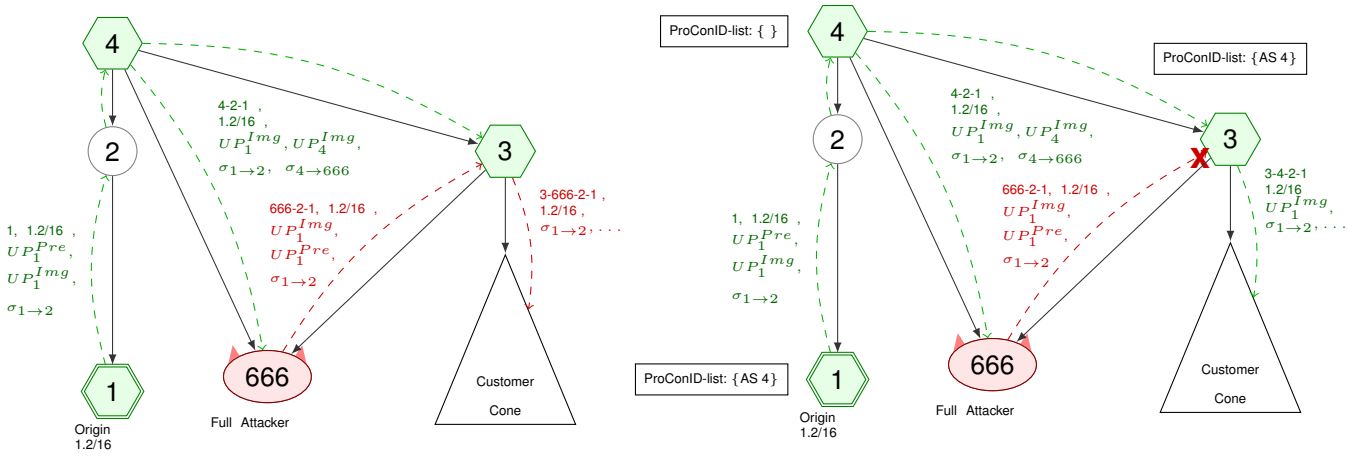
Fig. 4b illustrates the UP attributes defense. Recall that Fig. 4a shows that the attacker succeeds when only Protected-OTC is used. In Fig. 4b, we show how the same attack is foiled when ASes 1 and 4 deploy the UP attributes. ASes 1 and 4 generate random preimages UP_1^{Pre}, UP_4^{Pre}, compute $UP_1^{Img} = h(UP_1^{Pre})$, $UP_4^{Img} = h(UP_4^{Pre})$, and add UP_1^{Img}, UP_4^{Img} as transitive attribute to their announcements, respectively. Since AS 666 is a customer of AS 4, AS 4 does not send to AS 666 the UP_1^{Pre} and UP_4^{Pre} attributes. Instead, it adds and signs the OTC attribute, $\mathbf{OTC_4}$. AS 666 can shorten the AS-path and remove AS 4, and then remove $\mathbf{OTC_4}$ and UP_4^{Img} from the announcement. However, AS 666 cannot remove UP_1^{Img} (since this attribute is contained in $\sigma_{1\to2}$) or find the correct value of UP_1^{Pre}. Hence the route-leak announcement from AS 666 to AS 3 is detected and dropped.

Note that the scenario in Fig. 4b does not make use of UP_4^{Pre}. Indeed, in this and many other scenarios, it suffices for the origin (e.g., AS 1) to include the UP attribute. However, in some scenarios, the attacker may be able to obtain the origin's UP-preimage, e.g., when a rogue AS is one of the providers of the origin, but not the UP-image from another adopting AS. The UP-image attributes from non-origin ASes reduce this risk with an additional small overhead.

3) Providers-Cone Identification (ProConID): This mechanism protects against the Full Attacker, in contrast to the Protected-OTC and UP attributes defenses, which only protect against the Global Attacker. Fig. 5 shows the same setting as that in Fig. 4, except that it is for the Full Attacker model, instead of the Global Attacker model in Fig. 4. In Fig. 5a, the attacker, AS 666, knows the UP_1^{pre} attribute based on the Full Attacker model, and hence can construct a fake announcement with the legitimate UP_1^{pre} and UP_1^{img}, fooling AS 3 into choosing the fake announcement and falling victim to the route leak from AS 666. Fig. 5b illustrates the ProConID mechanism. As we shall see, it allows AS 3 to detect the route leak from AS 666.

The main element of ProConID is *ProConID-list*, a new RPKI object, signed and distributed by every BGP-iSec-adopting AS. For a BGP-iSec-adopting AS X, its ProConID-list can be obtained from the *provider cone* of X, i.e., the cone containing all the ASes that can be reached following provider-customer relationship upward from X (following convention, we place providers above customers in AS-topology). As an example, Fig. 6 shows the provider cone of an adopting AS (AS 1). Given the provider cone of AS X, the ProConID-list of X includes the *first* BGP-iSec-adopting AS along any upward path in the cone. In Fig. 6, the ProConID-list of AS 1 includes $\{2, 7, 8, 10\}$.

ProConID validation process. Consider an announcement A that contains an AS-path with a BGP-iSec-adopting AS, X_0, as the origin, followed by a sequence of ASes, $X_1, \ldots, X_\ell$, some being adopters and others not. We next only consider



(a) Without ProConID: vulnerable under a Full Attacker.

(b) With ProConID: secure under a Full Attacker.

Fig. 5: (a) Illustration of route leak by a Full Attacker, circumventing the UP-attribute and Protected-OTC defenses, and (b) how this attack is foiled by the ProConID defense. In (a), the attacker, AS 666, eavesdrops on the announcement from AS 2 to AS 4, and exports it as if it was sent to it. This announcement avoids the OTC (added by AS 4) and retains the UP preimage, hence, circumventing these defenses. In (b), adopting ASes (1, 3 and 4) deploy also ProConID. AS 3 detects the leak, since AS 1 is the last (and only) adopter in the AS-path of the announcement, and AS 3 is not included in AS 1's ProConID-list. AS 3 would similarly detect the leak if AS 666 would be adopting (not shown), where AS 3 detects the leak since AS 666 is not in the ProConID-list of AS 1.

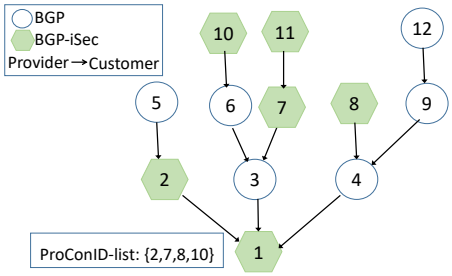


Fig. 6: Illustration of provider cone and ProConID-list of a BGP-iSec-adopting AS (AS 1).

BGP-iSec-adopting ASes along the path, and index them as $i_0 < \cdots < i_p \leq \ell$, where $i_0 = 0$ since X_0 is adopting, and X_ℓ can be adopting or not adopting, corresponding to $i_p = \ell$ and $i_p < \ell$, respectively. Assume another BGP-iSec-adopting AS, AS Y, receives announcement A from X_ℓ. Then AS Y discards A if it is not *ProConID-valid*, which we define as follows. (i) If X_ℓ is a provider of Y, then A is ProConID-valid. (ii) If X_ℓ is a peer of Y, then the condition that A is ProConID-valid if and only if for every BGP-iSec-adopting AS X_{i_j}, where $i_0 \leq i_j < i_p$, it holds that $X_{i_{j+1}}$ is in the ProConID-list of X_{i_j}. (iii) If X_ℓ is a customer of Y, then the condition that A is ProConID-valid only differs from case (ii) in that it further needs that Y is in the ProConID-list of X_{i_p}.

We use Fig. 5b to illustrate the above validation process. The ProConID-list of each BGP-iSec-adopting AS is marked in the figure. The validation fails for the route-leakage sent by the attacker (AS 666) to AS 3, since AS 3 is not listed in the ProConID-list of AS 1, while it should be if AS 666 is not adopting; and AS 666 does not appear in the ProConID-list of AS 1, while it should be if AS 666 is adopting.

Creating and maintaining ProConID-list. Network administrators may often know the identities and relationships of ASes in their provider cone, allowing them to directly and correctly define their ProConID-list. This is because provider cone is often small, and hence can be relatively easy to manage. For instance, using the CAIDA topology [44], we find that the median provider cone size of an AS is only 30 and the 90th percentile is 160. We can also assist network administrators to define and update the ProConID-list as follows. An adopting AS that detects that an incoming announcement may not pass ProConID validation can alert the origin and/or other relevant adopting ASes, allowing them to check and, if necessary, update their ProConID-list. The alert can use an out-of-band protocol, e.g., automated email. We expect the number of such events to be be relatively small since the sizes of the ProConID-list and provider cone are typically small, and the ProConID-list only requires the first adopter in each upstream path. In §IV-F, we quantify the operational overhead of ProConID using simulation.

When adopting ProConID in practice, as with other Internet validation mechanisms, e.g., Sender Policy Framework (SPF) [71], ASes may allow a grace period from the time of adoption of BGP-iSec (by origin and by receiving AS), during which they will only provide alerts but not yet drop ProConID-invalid announcements. Great care must also be taken when adding customer-provider relationships. When an adopting AS C joins the customer cone of another adopting AS, P, then ProConID filtering at P may drop announcements from C until C adds P to its ProConID-list. To prevent such transient dropping of announcements, we recommend that P will apply ProConID filtering only if P requires its direct customer ASes to report any new AS in P's customer cone. When P is informed of such new adopting AS C, then P should drop announcements that do not pass validation only after a grace period, allowing ProConID-lists to be updated.

The ProConID mechanism can be seen as a more secure variant of *Autonomous System Provider Authorization (ASPA)* [19], a recent IETF proposal. In ASPA, each participating AS publishes a signed list of its provider ASes, which are used to detect route leaks that violates valley-

free routing and some path manipulations (similar to Path-end validation [28]). If sufficiently deployed, ASPA can be effective against unintentional route leaks; however, even with high adoption, ASPA may fail against intentional route leaks. For example, consider topology of Fig. 5. Suppose that ASes 1, 4 and 3 adopt ASPA, while AS 2 does not, then AS 666 can still leak, using AS-path 666-2-1, *without the need of any eavesdropping capabilities*.

IV. Experimental Evaluation

A. Evaluation Settings

We evaluate the effectiveness of BGP-iSec using custom extensions[4] to the BGPy simulator [72]. This simulator propagates BGP announcements following the CAIDA serial 2 AS graph from September 2022 [44]. As described in §II-B, our simulations assume the widely used valley-free routing and export-to-all policy. For path selection, we consider both security-third and security-never policies. Since security-third has been used in existing works on BGP security (e.g., [26], [73]), we report the results for security-third to be consistent with the literature. We however recommend security-never in practice due to its ease of implementation and similar performance as security-third; see Appendix C-A.

We focus on post-ROV attacks. Prefix/sub-prefix hijacks are very effective attacks until ROV is widely adopted. For instance, as shown in [16], even when ROV adoption rate increases to 75%, a sub-prefix hijacker can intercept traffic from 25% of the ASes, which is higher than the rate obtained by path manipulation and route leaks (e.g., those shown in Fig. 7). Therefore, we assume wide adoption of ROV, and the attacker makes *ROV-valid* announcements (i.e., with legitimate origins and prefix lengths—no prefix or sub-prefix hijacks).

As in most evaluations of BGP security, we focus on the case of a single rogue AS; multiple collaborating rogue ASes will be more damaging. We consider two attack models, Global Attacker and Full Attacker (see §II-D). In both models, the attacker receives the announcements from *all* the ASes; in practice, we expect attackers to be able to collect announcements only from some ASes (e.g., from public BGP collectors). The Full Attacker is stronger since it has access to all BGP attributes, while Global Attacker is given access only to non-BGP-iSec attributes. In the following, unless otherwise specified, the results are for the Full Attacker model. We focus on *interception attacks*, where the attacker aims to attract the traffic destined to a victim AS (i.e., the origin of a prefix announcement), except in §IV-E, where we consider *DoS attacks*, i.e., the attacker aims to disconnect the victim AS.

We compare the security of BGP-iSec, BGPsec, and Path-end validation [28], which only protects the first hop from the origin and has been shown to outperform BGPsec. For each security policy, we assume a certain percentage of ASes adopts the policy, while the others run plain BGP. The percentage of adoption varies from 1%-99% for a given policy. At each percent adoption, 7,000 independent trials were performed with a uniformly randomly chosen attacker, origin, and set of adopting ASes. Unless otherwise stated, the origin is assumed to be adopting (i.e., adopting BGPsec, BGP-iSec, or Path-end validation), since otherwise, the attacker can simply use Aggressive strategy (see below), which is very effective for all the defenses we consider. As an example, in Fig. 2b, if the origin (AS 1) is not adopting BGP-iSec, then the attacker (AS 666) can simply announce 666-1, which will not be dropped by any BGP-iSec-adopting AS in the network. In Appendix C-C, we report the results when the origin may not be adopting. For each evaluation metric, we present the average and the 95% confidence intervals.

[4]Code available at https://github.com/c-morris/bgpy_pathsec

Attack strategies. The attacker may use path shortening and/or route leaks. Attacks can be *aggressive* or *timid*, i.e., without trying to or trying to avoid detection by the adopting ASes. For BGP-iSec, when the adoption rate is low, the aggressive approach is more likely to succeed, since there are less adopting ASes to drop hijack announcements. When the adoption rate increases, the timid approach becomes more effective for the attacker. Finding the optimal attack strategy to attract the most traffic is NP-hard [50]. Henceforth, we consider the following heuristic attack strategies that are intuitively effective:

• **Aggressive strategy:** With this strategy, the attacker uses the most aggressive path manipulation and export policy, not trying to avoid detection. Specifically, it uses a *1-hop* path manipulation, i.e., setting the AS-path to be the origin followed by itself (hence only the first position in AS-path is correct), and exports the hijack announcement to *all* its neighbors. Note that 1-hop path manipulation minimizes the AS-path length to make the announcement more likely to be selected as the best route, without violating ROV because the origin is legitimate. As we shall see, this Aggressive strategy is very effective for BGPsec; for BGP-iSec, it is only effective for low adoption.

• **Shortest-Path Export-All (SP-EA) strategy:** This strategy, first defined and used in [50], is similar to the Aggressive strategy, except being timid in terms of path manipulation: the attacker shortens the AS path as much as possible, but without being detected by the defense mechanisms. After attempting to shorten the path, the attacker exports (leaks) it to all neighbors.

• **Timid strategy:** This strategy is even more timid than SP-EA: the attacker is timid in both path manipulations *and* route leaks. Specifically, if an announcement has a Protected-OTC that cannot be removed or is missing an UP preimage unavailable to the attacker, the attacker will not attempt to leak the announcement at all. We found that in most scenarios, this Timid strategy is less effective than SP-EA. We therefore omit the results for this strategy.

Choice of attacker and origin ASes. We select both the attacker and legitimate origin (i.e., victim) uniformly at random from the set of multi-homed ASes, i.e., those that have more than one peer or provider. The reason for making the attacker multi-homed is two-fold. First, an attacker performing a route leak without shortening the path needs to have more than one provider to launch route leaks. Second, a stub AS with only one provider is more likely to be subject to prefix filtering by its provider, which makes the stub AS less likely to be able to perform any kind of attack at all. We choose the legitimate origin as a random multi-homed AS, since multi-home ASes are often large content providers or organizations with end users, and therefore more likely targets.

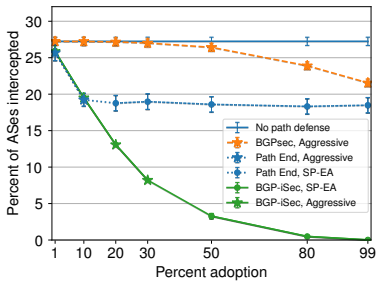


Fig. 7: The interception rates of a Full Attacker averaged over all ASes (adopting and non-adopting), comparing BGP-iSec, Path-End validation, BGPsec, and no path defense. All results assume full adoption of ROV. Percent adoption on the x-axis refers to the percentage of ASes adopting a security policy. The results for each security policy are for the best attacks: stars and circles on the curves represent the Aggressive and SP-EA strategies, respectively.

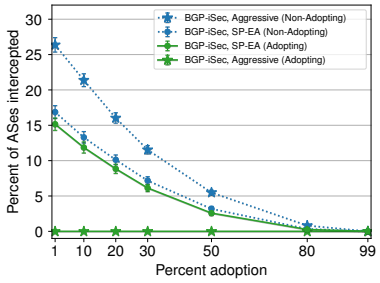


Fig. 8: The interception rates of a Full Attacker for adopting and non-adopting ASes separately.

B. BGP-iSec: Security Benefits

Fig. 7 compares the attacker interception rate, i.e., the percentage of ASes whose traffic to the victim is intercepted by the attacker. The results for each security policy (i.e., BGPsec, Path-end validation, or BGP-iSec) are obtained using the best attacker strategy at each adoption rate, i.e., the strategy that maximizes the attacker interception rate. Compared to the baseline (i.e., all ASes use BGP and ROV, but no path defenses against path manipulation and route leaks), BGPsec reduces the percentage of ASes hijacked by less than one percent until 30% of ASes have deployed it, and only roughly 5% more at 99% adoption. In contrast, BGP-iSec reduces the interception rate consistently, from about 27% down to essentially zero. In addition, BGP-iSec is already effective at low adoption rate: the attacker interception rate of BGP-iSec with only 10% adoption is already lower than that of BGPsec with 99% adoption. The performance of Path-end validation is between BGPsec and BGP-iSec: it leads to similar performance as BGP-iSec at low adoption rates, while significantly underperforms BGP-iSec at higher adoption rates.

Fig. 7 differentiates the results from the Aggressive and SP-EA strategies using stars and circles, respectively. For BGPsec, the Aggressive strategy is more effective than SP-EA for all adoption rates, since the attacker can always remove the signature and does not need to evade detection. For BGP-iSec, at low adoption, the Aggressive strategy is much stronger, but quickly becomes less effective than SP-EA at around 30% adoption. Similarly, for Path-end validation, the Aggressive strategy is initially more effective and then becomes less effective as the adoption rate increases.

Fig. 8 plots the attacker interception rate for BGP-iSec, separating between adopting and non-adopting ASes. For the adopting ASes (solid green lines), the attacker inception rate is zero under Aggressive strategy since the adopting ASes can detect the attack directly and will drop the hijack announcements; under SP-EA strategy, the attacker inception rate is non-zero, but drops steadily as the adoption rate increases. For the non-adopting ASes, the relative effectiveness of the two attack strategies is the opposite: the Aggressive strategy leads to higher interception rate than SP-EA for all adoption rates. Therefore, summarizing the interception rate across both adoption and non-adopting ASes, the best attack strategy changes from the Aggressive strategy to SP-EA as the adoption rate increases (see Fig. 7). We also see from Fig. 8 that, for a given adoption rate and attack strategy, the adopting ASes have lower interception rate than the non-adopting ASes, demonstrating the benefits of adopting BGP-iSec.

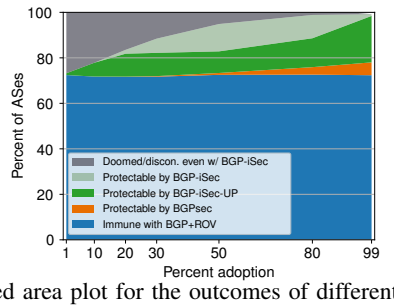
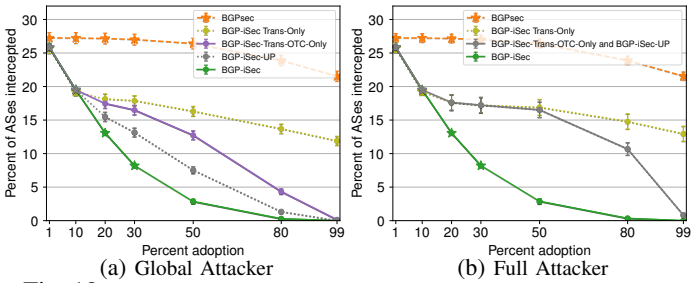


Fig. 9: Stacked area plot for the outcomes of different categories of ASes (adopting and non-adopting) against a Full Attacker.

C. BGP-iSec: Results Breakdown

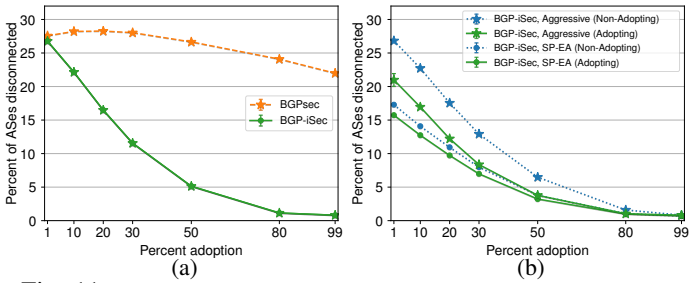
Following [26], we classify the ASes into the following five categories: (i) **Immune:** ASes that will route to the legitimate origin even in the baseline case, i.e., even without the use of BGPsec or BGP-iSec. (ii) **Protectable by BGPsec:** ASes whose traffic will be routed to the legitimate origin if they deploy either BGPsec or BGP-iSec, (iii) **Protectable by BGP-iSec:** ASes whose traffic will be routed to the legitimate origin if using BGP-iSec (but would be intercepted by attacker if using BGPsec), (iv) **Disconnected by BGP-iSec:** ASes whose traffic would be disconnected (to avoid interception) by BGP-iSec. (v) **Doomed (even with BGP-iSec):** ASes that will be intercepted by the attacker even if deploying BGP-iSec.

Fig. 9 is a stacked area plot on the percentages of the above categories of ASes versus adoption rate, considering all ASes. The rectangular blue region shows that the percentage of immune ASes among all the ASes is 72%. This value is independent of the adoption rate since it is for the baseline case with no adoption of BGPsec or BGP-iSec. The orange region shows the percentage of protectable ASes by BGPsec over the various adoption rates, which, consistent with the results in Fig. 7, increases very slowly with the adoption rate of BGPsec: even at 99% adoption, only 5.2% of the ASes are protectable by BGPsec. The green region shows the *additional* percentage of protectable ASes by BGP-iSec-UP over BGPsec, where BGP-iSec-UP is a variant of BGP-iSec, which adopts transitive signature, Protected-OTC, and UP attribute but not the ProConID mechanism. In contrast to the orange region, we already notice protectable ASes by BGP-iSec-UP even for low adoption rate of 10%; when the adoption rate reaches 99%, nearly all the ASes that are not already immune to attacks are protectable by BGP-iSec-UP. The light green region shows the



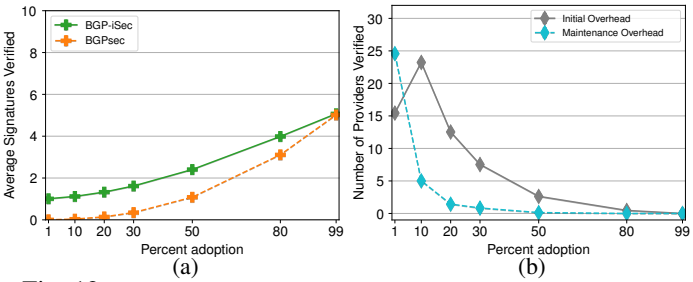
(a) Global Attacker (b) Full Attacker

Fig. 10: Impact of BGP-iSec mechanisms. In (b), the results for BGP-iSec-Trans-OTC-Only and BGP-iSec-UP overlap. Stars and circles mark Aggressive and SP-EA strategies, respectively.



(a) (b)

Fig. 11: Results for a DoS attacker under Full Attacker model: (a) results for all ASes, and (b) results for adopting and non-adopting ASes separately.



(a) (b)

Fig. 12: (a) Average number of signatures verified by each adopting AS per prefix: BGPsec vs. BGP-iSec. and (b) The operational overhead of ProConID mechanism.

additional percentage of ASes that are protected by the full-fledged BGP-iSec over BGP-iSec-UP, i.e., the benefits from the ProConID mechanism. We see that the benefits are clearer when the adoption rate is from 30% to 80%. Last, the gray region shows the percentage of doomed or disconnected ASes even with BGP-iSec (we present the percentage for these two categories together since the percentage of disconnection is close to zero), which decreases with the adoption rate and approaches zero when the adoption rate is 99%.

D. Impact of BGP-iSec Mechanisms

We consider the following variants of BGP-iSec: (i) **BGP-iSec-Trans-Only**: it only includes the transitive signature mechanism, the most basic mechanism in BGP-iSec, without the other mechanisms, (ii) **BGP-iSec-Trans-OTC-Only**: it includes the transitive signature and Protected-OTC but not UP attributes. Comparing this variant with BGP-iSec-Trans-Only shows the additional benefits from Protected-OTC. (iii) **BGP-iSec-UP**: it includes the transitive signature, Protected-OTC, and the UP attributes. Comparing this variant with BGP-iSec-Trans-OTC-Only shows the additional benefits from UP attributes, while comparing the full-fledged BGP-iSec with it shows the additional benefits from the ProConID mechanism.

Fig. 10a plots the attacker interception rate against all ASes under the Global Attacker model, again showing the best attack strategy for each adoption rate. The results for BGP-iSec, the above three variants, and BGPsec are plotted in the figure. We see that BGP-iSec-Trans-Only already significantly outperforms BGPsec, i.e., even just using the transitive signatures alone already leads to significant benefits over BGPsec. Comparing BGP-iSec-Trans-OTC-Only and BGP-iSec-Trans-Only shows the importance of Protected-OTC: it contributes to up to 15% reduction in attacker interception. The gap between BGP-iSec-UP and BGP-iSec-Trans-OTC-Only further shows the additional benefits of the UP attributes in defending against route leaks. Last, the ProConID mechanism leads to more benefits over BGP-iSec-UP (particularly for 20% to 80% adoption), at the cost of higher complexity.

Fig. 10b shows the results under the Full Attacker model. We again see that all variants of BGP-iSec have lower interception rate than BGPsec. For all variants of BGP-iSec, the interception rates under the Global and Full Attacker models are identical for the Aggressive strategy. For the SP-EA strategy, the Full Attacker model leads to higher interception rates than the Global Attacker model, for all variants of BGP-iSec except for the full-fledged BGP-iSec. Consider BGP-iSec-Trans-Only as an example. As long as there exists one non-adopting provider Y of the origin X, then the attacker R can shorten the path to R-Y-X and attach the signature from X to Y, and its attack will not be detected. The same observation holds for BGP-iSec-Trans-OTC-only and BGP-iSec-UP (in fact, the results of these two variants overlap in Fig. 10b since UP attribute is not effective in the Full Attacker model, as illustrated in Fig. 5a). Under the full-fledged BGP-iSec, the above attacks can be detected since the attacker is a multi-homed edge AS, and once it leaks to an adopting provider, the ProConID mechanism will detect the route leak and drop the leaked announcement (see one example in Fig. 5b).

E. Disconnection DoS Attack

So far, we have focused on interception attacks. We now consider an attacker whose goal is to *disconnect* the ASes, i.e., a *DoS* attacker. The disconnections can be due to two reasons: (i) *control-plane disconnections*, i.e., the attacker propagates a fake announcement with a short invalid path, which is later detected as invalid by an adopting AS and then dropped, causing some other ASes to have no route to the destination, and (ii) *data-plane disconnections*, i.e., the attacker was able to intercept data packets from other ASes, and then simply drop the traffic, instead of forwarding them to the destination. Fig. 11a shows the percentage of ASes (adopting and non-adopting) that are disconnected from the destination under a Full Attacker. It again shows the results from the best attacking strategy for each adoption rate. We see that BGP-iSec is significantly more effective in reducing disconnections than BGPsec. For both policies, the best attack strategy is the Aggressive strategy (marked by stars) for all adoption rates.

Fig. 11b shows the results for the adopting and non-adopting ASes separately with BGP-iSec. Under the same attack strategy, at a given adoption rate, adopting ASes have lower percentage of disconnections than non-adopting ASes, again demonstrating the benefits of adopting BGP-iSec.

F. Computational and Operational Overhead

The computational overhead of BGP-iSec and BGPsec differs in that BGPsec is mostly inactive until very high adoption rates, while BGP-iSec is active even at low adoption rates. In addition, BGP-iSec has overhead in hashing for the UP attributes, but this overhead is much lower than signature operations. Fig. 12a plots the average number of signatures verified per prefix for these two protocols. The results are obtained using the Best Path Only (BPO) optimization from [25]. That is, an adopting AS will only verify signatures on the *best* path it receives, as opposed to all of them. As expected, the gap in signatures verification overhead between BGP-iSec and BGPsec decreases with adoption rate, and becomes similar at high adoption rate. The results when using a naive, unoptimized implementation (i.e., an adopting AS verifies the signatures of all the announcements that it receives) show similar trends (see full version [74]).

For the ProConID mechanism, the operational overhead for an adopting AS comes primarily from creating and maintaining its ProConID-list. Specifically, we evaluate (i) initialization overhead when an AS first builds its ProConID-list, and (ii) maintenance overhead, i.e., the additional work for an AS to maintain its ProConID-list as other ASes adopt BGP-iSec. In both cases, we assume that an AS knows its immediate providers and needs to verify the providers that are two or more hops away (this is a conservative estimate since many times an AS knows its two-hop providers as well), and the overhead is quantified by the number of verifications that is needed for these unknown providers (which may need to be done manually by network administrators). For an AS, to determine its initialization overhead, we apply breadth-first search to its provider cone, and find the first adopting AS along each upstream branch, which forms a sub-tree, and then follow the sub-tree to determine the number of verifications needed. For example, in Fig. 6, AS 1 determines the sub-tree to ends at {2,7,8,10}. The cost for AS 1 to add AS 2 into its ProConID-list is 0 since AS 2 is a direct provider of AS 1; to add ASes 7, 8 and 10, the cost is 3 since AS 4 needs to verify AS 8, and AS 3 needs to verify ASes 7 and 6 (there is no cost for AS 6 to verify AS 10 since AS 10 is a direct provider of AS 6). During the verification for initialization, AS 1 further stores the provider information, which can be used to reduce maintenance cost later on. Specifically, suppose AS 6 adopts BGP-iSec at a later time, then AS 1 needs to add AS 6 into its ProConID-list, replacing AS 10, which will not incur any cost since AS 1 has already saved the information that AS 6 is a two-hop provider, closer than AS 10 during initialization.

Fig. 12b plots average overhead per AS, obtained by assuming a random order of adoption and calculating the overhead as ASes adopt following the order. As expected, initialization overhead first increases and then decreases with adoption rate. This is because for an AS X, for low adoption rate, very few ASes are in X's ProConID-list, while for high adoption rate, it is more likely that the ASes in X's ProConID-list are closer to X. We see maintenance overhead decreases with adoption rate since the provider information of AS X has been stored earlier and less update in ProConID-list is needed for high adoption rate.

BGP-iSec also increases somewhat the sizes of BGP announcements (due to additional attributes) and the amount of data that the ASes need to store (e.g., for downgrade tickets). However, the increase is modest, and unlikely to cause concern in terms of bandwidth or storage, especially with the out-of-band mechanism we describe in §III-A. In addition, BGP Extended Message Support [75] allows message size up to 64 KB, which can easily accommodate the BGP-iSec attributes.

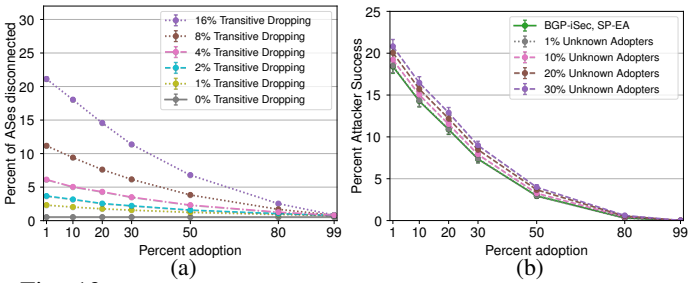


Fig. 13: (a) Percentage of BGP-iSec adopting ASes that are disconnected with various dropping rates; for non-adopting ASes, the disconnection rates are close to 0 for all the cases. (b) The interception rates of a Full Attacker when the KAPK assumption is violated.

G. Impact of Dropping Attributes

So far, we have assumed that benign ASes (i.e., the ASes that are not the attacker) do not drop transitive signatures in BGP-iSec. We next consider the scenarios where some non-adopting ASes drop transitive signatures. Specifically, a non-adopting AS can (i) discard the transitive signatures, and then forward the announcement, or (ii) drop an entire announcement that includes transitive signatures. The experiments in a recent study [69] on PEERING platform [76], [77] show that the above two cases happen to less than 2% and 1% of the ASes that they investigated, respectively. We evaluated BGP-iSec under both cases. In the following, we only present the results under case (i); the results under case (ii) are similar.

To combat the above dropping behavior, an AS that wants to adopt BGP-iSec can do one of the following: if it knows that all its providers drop transitive signatures, then it will not adopt BGP-iSec (since if it adopts, all its announcements will be dropped by some adopting ASes later on), or if it knows that at least one of its providers forwards unrecognized transitive attributes in compliance with the BGP specification, then it only uses such providers.

Following the above approach, BGP-iSec will avoid paths which include any ASes that drop transitive signatures. Therefore, the impact of these 'attribute dropping ASes' is mainly in causing disconnections. Fig. 13a plots the percentage of the BGP-iSec adopting ASes that are disconnected for a wide range of dropping rate, from 1% to 16%, i.e., 0.5 to 8 times of the dropping rate observed in [69]. The baseline result is when the dropping rate is zero. In this case, the disconnection is close-to-zero (it is not exactly zero due to the specific AS graph that we use). The disconnection rate increases with the dropping rate because a BGP-iSec AS must drop announcements with unexpected missing attributes, even though it is not due to attack and the AS path is actually legitimate. For all the dropping rates, the disconnection rate decreases as the BGP-iSec adoption rate increases. Even when the dropping rate is 4%, i.e., twice as that in [69], the disconnection rate is low (less than 5%) for all adoption rates. When the dropping rate increases to 8% and 16%, as expected, the disconnection

rate increases substantially, particularly for low adoption rate. On the other hand, as mentioned earlier, we expect that the dropping rate to be low, since path attributes (such as transitive signatures in BGP-iSec), once adopted, are supported by the router's BGP software [70].

H. Impact of Violating KAPK Assumption

The KAPK assumption (see §II-C) can be violated due to various reasons, e.g., publication delays in RPKI [78], failure conditions [79], or delayed fetching by the adopting ASes [80]. We next explore the impact of violating KAPK assumption, i.e., some BGP-iSec adopters are unknown by other adopters. In this scenario, the unknown adopters still verify signatures and enforce route leak prevention, but other adopters cannot verify their signatures. The attacker is also aware of which adopters are unknown, and therefore, will remove signatures from these unknown adopters, and then manipulate unprotected fields (e.g., remove OTC or shorten the AS-path), since such manipulation will not be detected.

We vary the percentage of unknown adopters in a wide range, 1% to 30%, to accommodate both normal and failure conditions. Fig. 13b plots the results. It only shows the results under SP-EA since the Aggressive strategy is not affected by the unknown adopters. We see that even when 30% of the adopters are unknown, which may only happen under extreme failure conditions, the impact is still small.

V. Security analysis

We next analyze BGP-iSec and prove several security properties under the MitM and Full Attacker models.

Announcement integrity. We next define *integrity-valid* announcements and the *announcement integrity* property;

Definition 1 (Integrity-valid announcements and announcement integrity)**.** *We say announcement A is* integrity-valid *if for every benign BGP-iSec-adopting AS X in the AS-path of A, it holds that X has previously sent an announcement A' whose integrity-protected attributes were* identical *to these in $TBSby_X(A)$, i.e., for every reversible attribute θ holds $A'[\theta] = A[\theta]$. We say that BGP-iSec ensures* announcement integrity *if benign BGP-iSec-adopting ASes drop every announcement which is not integrity-valid. See Definition 4 for the definitions of* integrity-protected attributes*, $A[\theta]$ and $TBSby_X(A)$.*

No false positives. There are situations where a benign BGP-iSec-adopting AS will discard an incoming integrity-valid announcement, e.g., when this is a route leak. However, such 'rogue announcements' can only be due to a rogue AS on the path, or to an announcement corrupted by a MitM attacker; BGP-iSec should not discard an announcement that was forwarded only by benign ASes. Let us define this *no false positives requirement*.

Definition 2 (No false positives)**.** *We say that BGP-iSec ensures* no false positives *if whenever a BGP-iSec-deploying benign AS Y flags an incoming announcement A as invalid (and discards it), then the AS-path of A contains at least one rogue AS, or A was corrupted by the attacker, i.e., is not a message sent by the last AS on the AS-path.*

Prevention of route leaks is a challenging goal; preventing *all* leaks appears to be impossible under partial adoption. Therefore, we also define a weaker notion, *visible-leak prevention*, which may suffice in practice. In both notions, we consider route leaks as routes which violate valley-free routing [48].

Definition 3 (Prevention of route leaks)**.** *We say that BGP-iSec ('completely') prevents leaks if a benign BGP-iSec-adopting AS Z discards every incoming announcement A received from a customer or peer, if the AS-path of A contains an AS X followed by an AS Y, where Y is a customer or peer of X, and X, Y or both are benign.*

We say that BGP-iSec prevents visible-leaks *if a benign BGP-iSec-adopting AS Z discards every incoming announcement A received from a customer or peer, if the AS-path of A contains an AS X followed by an AS Y, where Y is a customer or peer of X, and either (1) X is benign and BGP-iSec-adopting, or (2) the part of the AS-path from X to Z contains a benign BGP-iSec-adopting AS Y' before any non-benign AS X'.*

Theorem 1 shows that BGP-iSec *prevents route leaks* under full deployment and *prevents visible-leaks* under partial deployment, both against MitM adversary. We note that prevention of visible-leaks may suffice in practice, since the route leaks it may fail to prevent are from one rogue AS X to another rogue AS X' in the customer-cone of X; in such scenario, we may argue that X could have 'tunneled' the announcement to X', who would behave as if X is a customer, which seems an alternative, unpreventable and at least as effective route-leak attack. An example is shown in found in full version [74].

We now state the properties of BGP-iSec; the proof is found in Appendix E.

Theorem 1. *Assume (1) the transitive signatures and the integrity-protected attributes are correctly forwarded by all benign ASes, (2) the KAPK assumption (known adopting ASes and their public keys), and (3) valley-free routing. Then, BGP-iSec ensures against a MitM adversary (1)* announcement integrity*, (2)* no false positives *under full deployment and (3)* prevention of route leaks *under full deployment. Furthermore, against Full Attacker, BGP-iSec ensures (4)* no false positives *and (5)* prevention of visible leaks*.*

VI. Related Work

Improving performance of BGPsec. Several studies [22]–[25] aim to improve BGPsec. Their focus is on improving the computational cost of BGPsec, while BGP-iSec focuses on improving the security benefits of BGPsec.

Designs against path manipulations. Many proposals are for protecting BGP against path manipulations (see surveys [81]–[85]). In addition to RPKI/ROV [6] (and other origin authentication protocols [16], [86]), S-BGP [67] and Path-end validation [28] that were mentioned earlier, there are numerous other protocols such as soBGP [87], psBGP [88], pgBGP [89], IRV [90], SPV [91], and Listen and Whisper [92]; some of them are compared in [49], [50], [93]. Most of these protocols predate BGPsec. We design BGP-iSec to reuse, where possible, elements from BGPsec, and adopt transitive signatures in S-BGP, while make significant contributions in designing route leak defenses, conducting extensive evaluation, and analyzing

the security of BGP-iSec. In our evaluations, we showed that BGP-iSec significantly outperforms Path-end validation.

Designs against route leaks. Currently, the main defense against route-leaks is using filtering rules at routers (e.g., [94]); however, this is a slow, manual and error-prone process. There are several proposals for improved defense. Peerlock and Peerlock-lite [95], [96] are based on agreements between two transit ASes to protect their networks (specifically, one AS detects and filters route leaks for the other), which requires manual negotiation among pairs of ASes. The Down-Only (DO) community [70] and Only-to-Customer (OTC) attribute [18] are recent IETF proposals; both are not protected by signatures, and hence cannot defend against malicious attackers. Our Protected-OTC extends OTC to deal with malicious route leaks. As mentioned earlier, the approaches in [42], [43] are similar to Protected-OTC, which is not sufficient as we have shown in our evaluation results (see Fig. 10); our newly proposed UP attribute and ProConID provide significantly stronger protection against route leaks. ASPA [19] is another recent IETF proposal, which we have discussed briefly in §III-B3. BGP-iSec, and the above proposals, focus on *prevention* of route leaks; other works inspect route information logs to *detect* route leaks [97]–[100].

Alternative designs to BGP. Several works present *alternatives to BGP*, including SCION [35], MIRO [36], the seminal (but impractical) work of routing with Byzantine robustness [37] and more [38], [39]. In contrast, BGP-iSec does not require changes to BGP, and preserves much of the BGPsec design, which we expect to have a more likely contribution to the standardization and deployment.

VII. Conclusion and Future Work

We present BGP-iSec, a set of modifications and extensions to BGPsec, to provide better security in partial deployment and against additional threats, including route leaks and announcement manipulations. Using analysis and extensive simulations, we show that BGP-iSec provides significantly improved security over BGPsec, especially for partial adoption.

The design of BGP-iSec addresses both path manipulations and route leaks. It may be more convenient to address these two issues separately. Notice, however, that the Protected-OTC and UP attributes should be authenticated. BGP-iSec should not be viewed as a complete proposal, but as a basis to build upon for further designs. In addition, our design and simulations were limited to inter-AS operation of BGP; additional design and evaluation are needed for intra-AS aspects, including multiple routers connecting a pair of ASes.

More research is required to identify which of the BGP-iSec mechanisms, or other designs, would be best refined, standardized and deployed to improve the security of inter-domain routing. We next point out several directions. The first is *efficiency*; BGP-iSec focuses on improving the *security* of BGPsec in partial adoption but does not improve *efficiency*. While other works seek to improve the performance of BGP-sec, these optimizations may not be sufficient [25]. A second direction is *encouraging adoption* of path-security mechanisms such as BGP-iSec. A third direction is design, analysis and evaluation of other defenses against route leakage, e.g., evaluation of *ASPA* [19], [101] and comparison to the BGP-iSec anti-leakage defenses. Finally, several advanced aspects are not covered by BGP-iSec, e.g., *withdraw suppression* [47], support for private internal ASNs, and removal of prepending.

Acknowledgements

We thank the reviewers for their insightful and constructive feedback and the anonymous shepherd for guiding us through the revision process. We also thank Mark Ambrefe, Tom Beecher, Steven Bellovin, Justin Furuness, Joel Halpern, Jared Mauch, Lancheng Qin, Nicholas Scaglione, Job Snijders, Haya Shulman, John Scudder, Kotikalapudi Sriram, and Russ White for their comments and feedback.

This work is partially supported by the National Science Foundation under Grants No. 2247810, 2149765, and by the Comcast Corporation. The opinions expressed in this paper are those of the researchers and not of their university or funding sources. Cameron Morris' affiliation with the MITRE corporation is provided for identification purposes only, and is not intended to convey or imply MITRE's concurrence with, or support for, the positions, opinions, or viewpoints expressed by the author. Approved for Public release, Case #22-1487. Distribution unlimited.

References

[1] Y. Rekhter (Ed.), T. Li (Ed.), and S. Hares (Ed.), "A Border Gateway Protocol 4 (BGP-4)." RFC 4271 (Draft Standard), Jan. 2006. Updated by RFCs 6286, 6608, 6793, 7606, 7607, 7705, 8212, 8654, 9072.

[2] S. Murphy, "BGP Security Vulnerabilities Analysis." RFC 4272 (Informational), Jan. 2006.

[3] S. M. Bellovin, "Security problems in the TCP/IP protocol suite," *ACM SIGCOMM Computer Communication Review*, vol. 19, no. 2, pp. 32–48, 1989.

[4] M. Lad, D. Massey, D. Pei, Y. Wu, B. Zhang, and L. Zhang, "PHAS: a prefix hijack alert system," in *Proc. of USENIX Security Symposium*, 2006.

[5] K. Sriram, D. Montgomery, D. McPherson, E. Osterweil, and B. Dickson, "Problem Definition and Classification of BGP Route Leaks." RFC 7908 (Informational), June 2016.

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

[7] National Institute of Standards and Technology (NIST), "NIST RPKI Monitor, version 2.0." https://rpki-monitor.antd.nist.gov/. Accessed in November 2023.

[8] T. Chung, E. Aben, T. Bruijnzeels, B. Chandrasekaran, D. Choffnes, D. Levin, B. Maggs, A. Mislove, R. van Rijswijk-Deij, J. Rula, and N. Sullivan, "RPKI is Coming of Age: A Longitudinal Study of RPKI Deployment and Invalid Route Origins," in *Proc. of IMC*, ACM, 2019.

[9] T. Hlavacek, H. Schulmann, N. Vogel, and M. Waidner, "Keep your friends close, but your routeservers closer: Insights into RPKI validation in the internet," in *32nd USENIX Security Symposium, USENIX Security 2023, Anaheim, CA, USA, August 9-11, 2023*, pp. 4841–4858, USENIX Association, 2023.

[10] Y. Gilad, A. Cohen, A. Herzberg, M. Schapira, and H. Shulman, "Are We There Yet? On RPKI's Deployment and Security," in *NDSS*, The Internet Society, 2017.

[11] T. Hlavacek, A. Herzberg, H. Shulman, and M. Waidner, "Practical Experience: Methodologies for Measuring Route Origin Validation," in *IEEE/IFIP International Conference on Dependable Systems and Networks - DSN*, June 2018.

[12] N. Rodday, Ítalo S. Cunha, R. Bush, E. Katz-Bassett, G. D. Rodosek, T. C. Schmidt, and M. Wählisch, "Revisiting rpki route origin validation on the data plane," in *5th Network Traffic Measurement and Analysis Conference, TMA 2021, Virtual Event, September 14-15, 2021* (V. Bajpai, H. Haddadi, and O. Hohlfeld, eds.), IFIP, 2021.

[13] A. Reuter, R. Bush, I. Cunha, E. Katz-Bassett, T. C. Schmidt, and M. Wählisch, "Towards a rigorous methodology for measuring adoption of RPKI route validation and filtering," *ACM SIGCOMM Computer Communication Review*, vol. 48, no. 1, pp. 19–27, 2018. Online service: https://rov.rpki.net/.

[14] C. Testart, P. Richter, A. King, A. Dainotti, and D. Clark, "To Filter or Not to Filter: Measuring the Benefits of Registering in the RPKI Today," in *Proc. of Passive and Active Measurement Conference (PAM)*, pp. 71–87, Springer, Jan. 2020.

[15] D. Mirdita, H. Shulman, N. Vogel, and M. Waidner, "The CURE to vulnerabilities in RPKI validation," in *Proceedings of the 2024 Network and Distributed System Security (NDSS) Symposium*, 2024.

[16] R. Morillo, J. Furuness, C. Morris, J. Breslin, A. Herzberg, and B. Wang, "ROV++: Improved deployable defense against BGP hijacking," in *USENIX Network and Distributed System Security (NDSS) Symposium*, 2021.

[17] M. Lepinski (Ed.) and K. Sriram (Ed.), "BGPsec Protocol Specification." RFC 8205 (Proposed Standard), Sept. 2017. Updated by RFC 8206.

[18] A. Azimov, E. Bogomazov, R. Bush, K. Patel, and K. Sriram, "Route Leak Prevention and Detection Using Roles in UPDATE and OPEN Messages." RFC 9234 (Proposed Standard), May 2022.

[19] A. Azimov, E. Uskov, R. Bush, K. Patel, J. Snijders, and R. Housley, "A Profile for Autonomous System Provider Authorization," Internet-Draft draft-ietf-sidrops-aspa-profile-07, Internet Engineering Task Force, Jan. 2022. Work in Progress.

[20] F. Streibelt, F. Lichtblau, R. Beverly, A. Feldmann, C. Pelsser, G. Smaragdakis, and R. Bush, "BGP communities: Even more worms in the routing can," in *ACM IMC*, 2018.

[21] H. Birge-Lee, L. Wang, J. Rexford, and P. Mittal, "SICO: Surgical interception attacks by manipulating BGP communities," in *Proceedings of the 2019 ACM SIGSAC Conference on Computer and Communications Security*, pp. 431–448, 2019.

[22] D. M. Nicol, S. W. Smith, and M. Zhao, "Evaluation of efficient security for BGP route announcements using parallel simulation," *Simulation Modelling Practice and Theory*, vol. 12, no. 3, pp. 187 – 216, 2004. Modeling and Simulation of Distributed Systems and Networks.

[23] K. Butler, P. McDaniel, and W. Aiello, "Optimizing BGP security by exploiting path stability," in *Proceedings of ACM Conference on Computer and Communications Security (CCS)*, (New York, NY, USA), pp. 298–310, ACM, 2006.

[24] M. Zhao, S. W. Smith, and D. M. Nicol, "Aggregated path authentication for efficient BGP security," in *Proc. of CCS*, 2005.

[25] V. K. Sriram and D. Montgomery, "Design and analysis of optimization algorithms to minimize cryptographic processing in BGP security protocols," *Computer communications*, vol. 106, pp. 75–85, 2017.

[26] R. Lychev, S. Goldberg, and M. Schapira, "BGP security in partial deployment: Is the juice worth the squeeze?," *ACM SIGCOMM Computer Communication Review*, vol. 43, no. 4, pp. 171–182, 2013.

[27] S. Goldberg, "Why is it taking so long to secure Internet routing?," *Queue*, vol. 12, no. 8, p. 20, 2014.

[28] A. Cohen, Y. Gilad, A. Herzberg, and M. Schapira, "Jumpstarting BGP Security with Path-End Validation," in *Proceedings of the 2016 ACM SIGCOMM Conference*, SIGCOMM '16, (New York, NY, USA), pp. 342–355, ACM, 2016.

[29] P. Gill, M. Schapira, and S. Goldberg, "Let the market drive deployment: A strategy for transitioning to BGP security," *ACM SIGCOMM Computer Communication Review*, vol. 41, no. 4, pp. 14–25, 2011.

[30] T. Federal Communications Commission, "FCC inquiry into internet routing vulnerabilities." DA/FCC #: FCC-22-18, Docket/RM: 22-90, online at: https://www.fcc.gov/document/fcc-launches-inquiry-internet-routing-vulnerabilities, Feb. 2022.

[31] Cisco Systems, "Comments of CISCO systems, inc. to FCC PS docket 22-90, in the matter of secure internet routing," April 2022.

[32] Juniper Networks, "Comments of Juniper Networks re: secure internet routing (FCC PS docket 22-90)," April 2022.

[33] M. Ermert, "Missing link: How the Internet will be kept running (also in the future)." https://www.kiratas.com/missing-link-how-to-keep-the-internet-running-also-in-the-future/, April 2022.

[34] G. Huston, "Ietf 102: An update on securing bgp." https://blog.apnic.net/2018/07/25/ietf-102-an-update-on-securing-bgp/, July 2018.

[35] D. Barrera, L. Chuat, A. Perrig, R. M. Reischuk, and P. Szalachowski, "The SCION internet architecture," *Communications of the ACM*, vol. 60, no. 6, pp. 56–65, 2017.

[36] W. Xu and J. Rexford, "MIRO: multi-path interdomain routing," in *ACM SIGCOMM*, pp. 171–182, 2006.

[37] R. Perlman and C. Kaufman, "Hierarchical networks with byzantine robustness," in *2011 Third International Conference on Communication Systems and Networks (COMSNETS 2011)*, pp. 1–11, IEEE, 2011.

[38] D. Gupta, A. Segal, A. Panda, G. Segev, M. Schapira, J. Feigenbaum, J. Rexford, and S. Shenker, "A new approach to interdomain routing based on secure multi-party computation," in *Proceedings of the 11th ACM Workshop on Hot Topics in Networks*, pp. 37–42, 2012.

[39] Y. Liu, S. Zhang, H. Zhu, P.-J. Wan, L. Gao, and Y. Zhang, "An enhanced verifiable inter-domain routing protocol based on blockchain," in *International Conference on Security and Privacy in Communication Systems*, pp. 63–82, Springer, 2019.

[40] National Institute of Standards and Technology (NIST), "BGP Secure Routing Extensions Software Suite." https://tinyurl.com/e6t6672f, April 2022.

[41] S. Kent, C. Lynn, and K. Seo, "Secure Border Gateway Protocol (S-BGP)," *IEEE Journal on Selected areas in Communications*, vol. 18, no. 4, pp. 582–592, 2000.

[42] S. Sundaresan, R. Lychev, and V. Valancius, "Preventing attacks on BGP policies: One bit is enough," Tech. Rep. GT-CS-11-07, Georgia Institute of Technology, 2011.

[43] K. Sriram and D. Montgomery, "Enhancement to BGPsec for protection against route leaks," 2014. draft-sriram-route-leak-protection-00.

[44] "CAIDA Serial 2 Data Set," Apr. 2022.

[45] K. Sriram (Ed.), "BGPsec Design Choices and Summary of Supporting Discussions." RFC 8374 (Informational), Apr. 2018.

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

[47] S. Kent and A. Chi, "Threat Model for BGP Path Security." RFC 7132 (Informational), Feb. 2014.

[48] L. Gao and J. Rexford, "Stable Internet Routing without Global Coordination," *IEEE/ACM Trans. Netw.*, vol. 9, no. 6, pp. 681–692, 2001.

[49] H. Chan, D. Dash, A. Perrig, and H. Zhang, "Modeling Adoptability of Secure BGP Protocols," in *Proc. of SIGCOMM*, ACM, 2006.

[50] S. Goldberg, M. Schapira, P. Hummon, and J. Rexford, "How Secure are Secure Interdomain Routing Protocols?," in *Proc. of SIGCOMM*, ACM, 2010.

[51] W. Mühlbauer, A. Feldmann, O. Maennel, M. Roughan, and S. Uhlig, "Building an AS-topology model that captures route diversity," in *Proc. of SIGCOMM*, 2006.

[52] H. Madhyastha, E. Katz-Bassett, T. Anderson, A. Krishnamurthy, and A. Venkataramani, "iPlane Nano: Path Prediction for Peer-to-Peer Applications," in *Proce of NSDI*, 2009.

[53] R. Mazloum, M. Buob, J. Auge, B. Baynat, D. Rossi, and T. Friedman, "Violation of Interdomain Routing Assumptions," in *Proc. of Passive and Active Measurement Conference (PAM)*, March 2014.

[54] R. Anwar, H. Niaz, D. Choffnes, I. Cunha, P. Gill, and E. Katz-Bassett, "Investigating Interdomain Routing Policies in the Wild," in *Proc. of ACM IMC*, Oct. 2015.

[55] P. Mohapatra, J. Scudder, D. Ward, R. Bush, and R. Austein, "BGP Prefix Origin Validation." RFC 6811 (Proposed Standard), Jan. 2013. Updated by RFCs 8481, 8893.

[56] M. Reynolds, S. Turner, and S. Kent, "A Profile for BGPsec Router Certificates, Certificate Revocation Lists, and Certification Requests." RFC 8209 (Proposed Standard), Sept. 2017.

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

[58] T. Hlavacek, P. Jeitner, D. Mirdita, H. Schulmann, and M. Waidner, "Stalloris: RPKI downgrade attack," in *31st USENIX Security Symposium (USENIX Security 22)*, (Boston, MA), USENIX Association, Aug. 2022.

[59] D. Cooper, E. Heilman, K. Brogle, L. Reyzin, and S. Goldberg, "On the Risk of Misbehaving RPKI Authorities," in *Proceedings of the Twelfth ACM Workshop on Hot Topics in Networks*, HotNets-XII, (New York, NY, USA), pp. 16:1–16:7, ACM, 2013.

[60] T. Dai, P. Jeitner, H. Shulman, and M. Waidner, "The hijackers guide to the galaxy:{Off-Path} taking over internet resources," in *30th USENIX Security Symposium (USENIX Security 21)*, pp. 3147–3164, 2021.

[61] K. Shrishak and H. Shulman, "Limiting the power of rpki authorities," in *Proceedings of the Applied Networking Research Workshop*, pp. 12–18, 2020.

[62] T. Hlavacek, P. Jeitner, D. Mirdita, H. Schulmann, and M. Waidner, "Beyond limits: How to disable validators in secure networks," in *Proceedings of the ACM SIGCOMM 2023 Conference, ACM SIGCOMM 2023, New York, NY, USA, 10-14 September 2023*, pp. 950–966, ACM, 2023.

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

[64] J. Touch, A. Mankin, and R. Bonica, "The TCP Authentication Option." RFC 5925 (Proposed Standard), June 2010.

[65] RouteViews, "University of Oregon Route Views Project." http://www.routeviews.org/routeviews/, 2018.

[66] "RIPE NCC. Routing Information Service (RIS)." https://www.ripe.net/analyse/internet-measurements/routing-information-service-ris.

[67] S. T. Kent, C. Lynn, and K. Seo, "Secure Border Gateway Protocol (S-BGP)," *IEEE Journal on Selected Areas in Communications*, vol. 18, no. 4, pp. 582–592, 2000.

[68] K. Lougheed and Y. Rekhter, "Border Gateway Protocol 3 (BGP-3)." RFC 1267 (Historic), Oct. 1991.

[69] T. Hlavacek, I. Cunha, Y. Gilad, A. Herzberg, E. Katz-Bassett, M. Schapira, and H. Shulman, "DISCO: Sidestepping RPKI's deployment barriers," in *Proceedings of the 2020 Network and Distributed System Security (NDSS) Symposium*, February 2020.

[70] K. Sriram, A. Azimov, "Methods for Detection and Mitigation of BGP Route Leaks," Apr. 2022.

[71] S. Kitterman, "Sender Policy Framework (SPF) for Authorizing Use of Domains in Email, Version 1." RFC 7208 (Proposed Standard), Apr. 2014. Updated by RFCs 7372, 8553, 8616.

[72] J. Furuness, C. Morris, R. Morillo, A. Herzberg, and B. Wang, "Bgpy: The bgp python security simulator," in *Proceedings of the 16th Cyber Security Experimentation and Test Workshop*, CSET '23, (New York, NY, USA), p. 41–56, Association for Computing Machinery, 2023.

[73] P. Gill, M. Schapira, and S. Goldberg, "A survey of interdomain routing policies," *ACM SIGCOMM Computer Communication Review*, vol. 44, no. 1, pp. 28–34, 2013.

[74] C. Morris, A. Herzberg, B. Wang, and S. Secondo, "BGP-iSec: improved security against Post-ROV routing attacks (full version)." https://www.researchgate.net/publication/375553362_BGP-iSec_Improved_Security_of_Internet_Routing_Against_Post-ROV_Attacks, 2023.

[75] R. Bush, K. Patel, and D. Ward, "Extended Message Support for BGP." RFC 8654 (Proposed Standard), Oct. 2019.

[76] B. Schlinker, T. Arnold, I. Cunha, and E. Katz-Bassett, "PEERING: virtualizing BGP at the edge for research," in *Proc. of CoNEXT*, Dec. 2019.

[77] E. Katz-Bassett, B. Schlinker, I. Cunha, and N. Feamster, "PEERING: an AS for us," in *Proceedings of the HotNets-XIII*, 2014.

[78] R. Fontugne, A. Phokeer, C. Pelsser, K. Vermeulen, and R. Bush, "RPKI Time-of-Flight: Tracking Delays in the Management, Control, and Data Planes," in *PAM*, 2023.

[79] "Failure and recovery scenarios." https://krill.docs.nlnetlabs.nl/en/stable/failure-scenarios.html.

[80] J. Kristoff, R. Bush, C. Kanich, G. Michaelson, A. Phokeer, T. C. Schmidt, and M. Wählisch, "On measuring RPKI relying parties," in *IMC*, 2020.

[81] K. Butler, T. R. Farley, P. McDaniel, and J. Rexford, "A survey of BGP security issues and solutions," *Proceedings of the IEEE*, vol. 98, no. 1, pp. 100–122, 2010.

[82] G. Huston, M. Rossi, and G. Armitage, "Securing BGP: A literature survey," *IEEE Communications Surveys & Tutorials*, vol. 13, no. 2, pp. 199–222, 2011.

[83] A. Herzberg, M. Hollick, and A. Perrig, "Secure Routing for Future Communication Networks (Dagstuhl Seminar 15102)," *Dagstuhl Reports*, vol. 5, no. 3, pp. 28–40, 2015.

[84] A. Mitseva, A. Panchenko, and T. Engel, "The state of affairs in BGP security: A survey of attacks and defenses," *Computer Communications*, vol. 124, pp. 45–60, June 2018.

[85] M. S. Siddiqui, D. Montero, R. Serral-Gracia, X. Masip-Bruin, and M. Yannuzzi, "A survey on the recent efforts of the Internet Standardization Body for securing inter-domain routing," *Computer Networks*, vol. 80, pp. 1–26, April 2015.

[86] W. Aiello, J. Ioannidis, and P. McDaniel, "Origin authentication in interdomain routing," in *Proc of CCS*, 2003.

[87] R. White, "Securing BGP through secure origin BGP (sobgp)," *Business Communications Review*, vol. 33, no. 5, pp. 47–47, 2003.

[88] P. C. v. Oorschot, T. Wan, and E. Kranakis, "On interdomain routing security and pretty secure BGP (psbgp)," *ACM Transactions on Information and System Security (TISSEC)*, vol. 10, no. 3, pp. 1–41, 2007.

[89] J. Karlin, S. Forrest, and J. Rexford, "Autonomous security for autonomous systems," *Computer Networks*, Oct. 2008.

[90] G. Goodell, W. Aiello, T. Griffin, J. Ioannidis, P. D. McDaniel, and A. D. Rubin, "Working around BGP: An Incremental Approach to Improving Security and Accuracy in Interdomain Routing," in *NDSS*, The Internet Society, 2003.

[91] Y.-C. Hu, A. Perrig, and M. A. Sirbu, "SPV: secure path vector routing for securing BGP," in *SIGCOMM*, 2004.

[92] L. Subramanian, V. Roth, I. Stoica, S. Shenker, and R. Katz, "Listen and whisper: Security mechanisms for BGP," in *Proc. of NSDI*, 2004.

[93] R. Lychev, M. Schapira, and S. Goldberg, "Rethinking security for Internet routing," *Communication of the ACM*, vol. 59, no. 10, pp. 48–57, 2016.

[94] K. Sriram and D. Montgomery, "Resilient Interdomain Traffic Exchange: BGP Security and DDoS Mitigation." https://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-189.pdf, 2019.

[95] J. Snijders, "NTT peer locking." http://instituut.net/~job/peerlock_manual.pdf, 2016.

[96] T. McDaniel, J. M. Smith, and M. Schuchard, "Peerlock: Flexsealing BGP." https://arxiv.org/abs/2006.06576, June 2020.

[97] M. Siddiqui, D. Montero, S.-G. R., and M. Yannuzzi, "Self-reliant detection of route leaks in inter-domain routing," *Computer Networks*, vol. 82, pp. 135–155, 2015.

[98] A. Haeberlen, I. Avramopoulos, J. Rexford, and P. Druschel, "Netreview: detecting when interdomain routing goes wrong," in *USENIX Symposium on Networked Systems Design and Implementation (NDSI)*, 2009.

[99] A. Gurney, A. Haeberlen, W. Zhou, M. Sherr, and B. Loo, "Having your cake and eating it too: routing security with privacy protections," in *ACM Workshop on Hot Topics in Networks(HotNets)*, 2011.

[100] M. Zhao, W. Zhou, A. Gurney, A. Haeberlen, M. Sherr, and B. Loo, "Private and verifiable interdomain routing decisions," in *ACM SIGCOMM*, 2012.

[101] N. Rodday, G. D. Rodosek, A. Pras, and R. van Rijswik-Deij, "Exploring the benefit of path plausibility algorithms in BGP," in *Arxiv preprint*, 2023.

[102] A. Herzberg, H. Leibowitz, E. Syta, and S. Wrótniak, "MoSS: Modular Security Specifications framework," in *CRYPTO' 2021*, 2020. https://eprint.iacr.org/2020/1040.

APPENDIX A
DEFENSES AGAINST PATH SHORTENING

A. Limiting Path Shortening

Validation of announcement integrity prevents many path manipulations. However, in partial adoption, a rogue AS R may yet be able to send an *integrity-valid yet manipulated*

announcement A_R. Specifically, assume that the attacker obtains an announcement A whose AS-path contains some non-adopting AS N. Let A_R be an announcement whose AS-path is identical to that of A up to N, and then followed by the rogue AS R (and optionally other non-adopting and/or rogue ASes). Announcement A_R can be integrity-valid, since benign adopting ASes appear in its AS-path only *before* N, i.e., in the part that was not manipulated. We refer to such a manipulation, which results in *integrity-valid yet manipulated (shortened)* announcement A_R, as *path shortening*.

Why would the rogue AS R prefer to relay the manipulated announcement A_R, rather than the actual announcement A? One reason is that A_R may have a shorter AS-path, and therefore, more likely to be selected by receiving ASes. Another reason is that this may allow R to remove or manipulate immutable transitive attributes in A, e.g., the OTC attribute, which may prevent export to providers and peers. By replacing A with A_R, R can relay the manipulated A_R to peers or providers, and hence may attract more traffic to itself.

We expected the path-shortening attack to have considerable value for the attacker. In the following, we present a technique against path shortening.

B. Hash Chain Against Path Shortening

This path-shortening prevention mechanism is motivated by the limitations in BGPsec. While BGPsec already includes a defense against path shortening, namely, the signed next-AS field, this defense is effective only if all ASes on the path are signing. The attacker can easily foil this mechanism by exporting an announcement with a shorter path ending at a rogue AS, e.g., removing the signature of a previous AS on the path. It typically can shorten the path so that only the origin is left, as shown in the 1-hop hijack in Fig. 2.

By discarding announcements that do not contain all required signatures, i.e., prohibiting downgrades, BGP-iSec prevents many path-shortening attacks. However, since we allow partial adoption, this mechanism, by itself, is insufficient for a number of feasible path-shortening attacks. This is because although the next-AS field is signed, the first non-BGP-iSec AS on the path, say N, leaves a gap. An attacker could take advantage of this gap by truncating the path in an incoming announcement A right after the first non-adopting AS (N), and then exporting the announcement as if the attacker received it directly from N. In early adoption, the first non-adopting AS on the path may be very close to the origin, which seem to imply that such attacks can be quite effective. For example, in the topology shown in Fig. 14, AS 666 may send the announcement with AS-path of 1-2-666, which would be shorter than the path 1-2-7-5 sent by AS 5, thereby preferred by AS 3.

Fig. 14 illustrates the *hash-chain defense against path shortening*, which helps to mitigate this risk. Let us explain this defense, although, we do not include it in BGP-iSec, since our simulations results show that its benefits may not justify its overhead in complexity and cost. We believe the limited value of this defense is since in low adoption, there are often not enough adopting ASes along the path to make a difference; and in high adoption, the signed next-AS field prevents path shortening on its own.

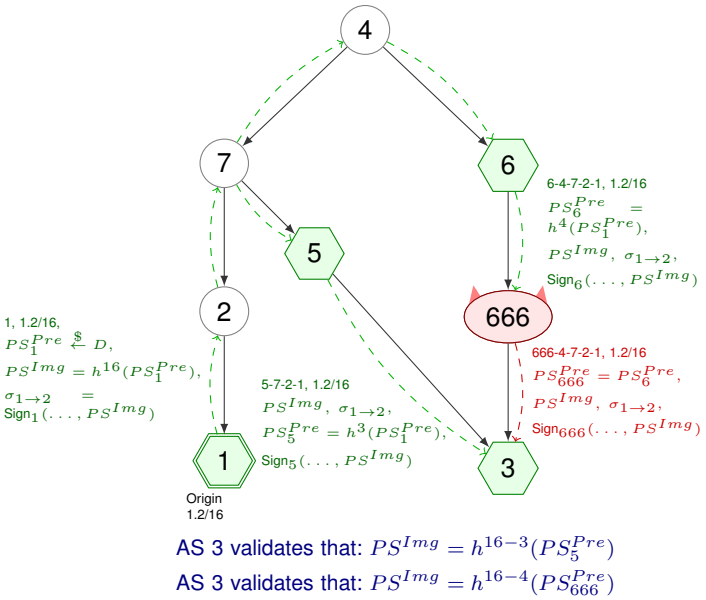


Fig. 14: Example hash chain against path shortening where $l = 16$. The value of $PS^{Img} = h^{16}(PS_1^{Pre})$ is included with the other attributes in BGP-iSec signature. Even with the PS attributes, an attacker can still *replace* the last adopting AS on the path in scenarios like this, however, they cannot shorten it beyond the last adopting AS. AS 3 does not detect anything invalid about the announcement from AS 666, but it still chooses the correct route from AS 5 because it has a shorter path. For clarity, this figure omits the OTC and UP-attributes which are both used to prevent route leakage (§III-B), and has other simplifications, e.g., showing only messages from BGP-iSec-ASes.

The hash-chain defense adds two new transitive attributes to an exported announcement, which we refer to as PS^{Pre} and PS^{Img}. The PS term stands for *Path Shortening*, and the superscripts refer to *Preimage* and *Image*, respectively, of a *hash chain* of sufficient length[5] l, e.g., $l = 16$.

The origin selects $PS^{Pre} \xleftarrow{\$} D$ randomly from some domain D, typically binary strings of sufficient length, and computes $PS^{Img} \leftarrow h^l(PS^{Pre})$, where $h : D \to D$ is a one-way hash[6] function. The value of the PS^{Img} attribute never changes, and the origin signs it (like other attributes). However, the value of the PS^{Pre} is not signed, as it changes whenever the announcement is exported. To distinguish between the different values of PS^{Pre} in Fig. 14, we use a subscript to identify the sending AS, i.e., PS_X^{Pre} is the preimage attribute as sent by AS X.

Let us explain what happens when an adopting AS Y receives an announcement A, where the AS-path contains (another) adopting AS X; assume that X is the last adopting-AS in the AS-path. First, Y validates A; in addition to the validation of the signature and of the OTC indicators, Y also validates the hash-chain, as follows.

Let $l' \geq 0$ be the number of hops from the origin till X (including non-adopters), and PS^{Img}, PS_X^{Pre} be the relevant attributes in A. Then Y validates that $PS^{Img} = h^{l-l'}(PS_X^{Pre})$.

[5] AS-paths can contain up to 255 hops, so $l = 255$ suffices. However, a much smaller value, e.g., $l = 16$, would probably suffice, since the average AS-path length is about four.

[6] To be more precise, h would be a one-way permutation.

If the announcement A is valid and Y exports it, then Y computes: $PS_Y^{Pre} \leftarrow h^k(PS_X^{Pre})$, where k is the number of hops from X to Y (including non-adopting ASes). In particular, if Y received the announcement directly from adopting AS X, then $PS_Y^{Pre} \leftarrow h(PS_X^{Pre})$. AS Y exports the announcement with only PS_Y^{Pre}; the 'previous' PS^{Pre} value, i.e., PS_X^{Pre}, is removed. Also note that because of the one-way property, it is easy to compute PS_Y^{Pre} and PS^{Img} given PS_X^{Pre}, but infeasible to compute PS_X^{Pre} given PS_Y^{Pre}.

The path shortening defense has the following limitation: if Y is a rogue AS, it can remove X and all subsequent non-adopting ASes in the AS-path received, by reusing the value of PS_X^{Pre}. For example, this is done by AS 666 in Fig. 14. In addition, as discussed in §II, the path-shortening defense fails against a Full Attacker, since such adversary can know the preimage as sent by the origin. To summarize, path-shortening is effective against the *Global Attacker* model (see §II-D). Note, however, that some routers make public the BGP announcements they receive, e.g., in the RouteViews and RIPE RIS services. For the hash-chain mechanism to work, it is desirable for such services to remove the PS^{Pre} attributes; luckily, often these services only provide specific, well-known attributes. Note also that multiple colluding ASes could send each other hash chains to avoid this mechanism.

We evaluated the path shortening mechanism using extensive simulation. While intuitively it appears to be beneficial, we found that it only leads very little reduction in attacker success in partial deployment (no more than 2-3% in Global Attacker model). In retrospect, the limited benefits are not surprising: under high adoption rate, the next-AS in the transitive signatures already prevents path shortening attacks and the hash chain does not provide much additional benefit; under low adoption rate, the benefits are also limited, since it is not common that there exists an adopting AS that precedes the attacker to prevent the attacker from shortening the path.

Appendix B
Only-to-customers (OTC) Attribute

The *only-to-customers (OTC)* attribute [18] is designed to prevent (unintentional) route leaks. When protected by BGP-iSec, the OTC attribute prevents some malicious route leaks. We discuss it, and additional BGP-iSec defenses against route leaks, in §III-B. The value of the OTC attribute is an ASN X in the AS-path, which is a provider or peer of the following AS in the AS-path, and therefore, following X, the announcement should be exported *only to customers* (hence, OTC). An adopting AS adds the OTC attribute containing its own identity if it exports to a peer or customer, and the identity of the previous AS if that AS is a peer or provider.

We briefly review this method below. For simplicity, we only consider customer, provider and peer relationships, excluding Router Servers and Router Clients in IXPs, which are described in [18]. We first describe how AS Y uses OTC to filter received announcements. Assume AS Y *receives* an announcement with the OTC attribute, with value x, from neighbor X. Then AS Y drops the announcement if *either (1)* X is a customer of Y, *or (2)* X is a peer of Y and $x \neq X$. The second rule allows the peer X to add the OTC attribute, with its own ASN (X) as value, i.e., $x = X$.

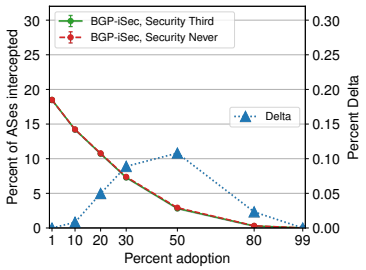


Fig. 15: The interception rates for BGP-iSec with security-third vs. security-never policies in Full Attacker model against a Shortest-Path-Export-All attacker. The Delta line (on the right-hand of y-axis) is the interception rate of security-never minus that of security-third.

We now describe how and when AS Y will *add* the OTC attribute; notice that the *value* of this attribute differs in the two cases when AS Y adds it: *(1)* when *exporting* an announcement to a peer or a customer AS Z, and *(2)* when *receiving* the announcement from a provider or peer AS X. In both cases, the value of the attribute added is the ASN of the *exporting* AS, i.e., in case (1), the value is the ASN of Y itself, and in case (2) the value of the attribute is the ASN of X. The two cases offer redundancy, which is beneficial when not all ASes support the attribute, or if some ASes corrupt the attribute.

Appendix C
Additional Evaluation Results

A. Security-never versus Security-third

In our simulation, we do not observe visible benefits from security-third over security-never. Fig. 15 plots the interception rates under interception attack and Full Attacker model. The results are for SP-EA strategy, since the results are identical for the two policies under Aggressive strategy. We see that the results under these two policies almost overlap with each other. Similar results hold for the disconnection DoS attack.

We next briefly discuss implementation of security-never and security-third in practice. Security-never is easy to implement—a router simply needs to discard invalid announcements and then follow the standard path-selection policy that is already implemented. Implementing security-third incurs more effort, since router software would need to be modified to prefer paths where all ASes adopt BGP-iSec as part of the path-selection process. Security-never may also have somewhat lower computational overhead compared to security-third, especially if not implementing the BPO optimization from [25]. Given the nearly identical interception rates of security-never and security-third as well as the additional efforts in implementing security-third, we believe that it is not justified to implement security-third in practice.

B. Overhead for Unoptimized Implementation

In §IV, we present overhead results with the Best Path Only (BPO) optimization. For completeness, here in Fig. 16 shows results for an un-optimized BGPsec implementation that verifies every signature on every incoming announcement. Although the overhead is higher, BGPsec and BGP-iSec still converge as adoption approaches 100%.

C. Non-adopting Origins

We next consider the case where origins may not be adopting. For a non-adopting origin, the attacker can simply

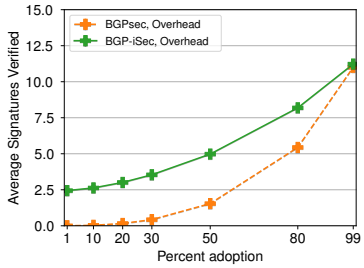


Fig. 16: Average number of signatures verified by each adopting AS per prefix using no optimization methods. As adoption approaches 100%, BGPsec and BGP-iSec converge to the same amount of overhead.

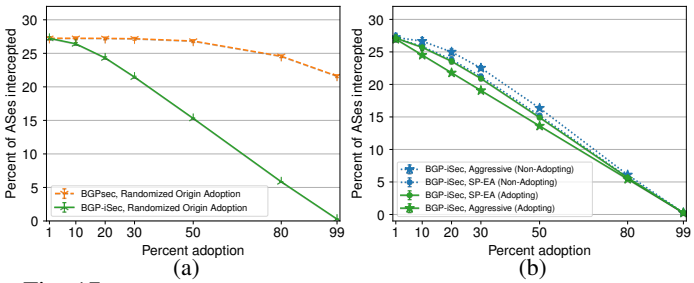


Fig. 17: Results for random origin (i.e., original can be adopting or not adopting based on percentage of adopting) for interception attack under Full Attacker model: (a) results for all ASes, and (b) results for adopting and non-adopting ASes separately.

use the Aggressive strategy and not be detected, causing the interception rate to be identical to the "No path defense" line in Fig. 7, i.e., the results are independent of the percent of adoption. We next consider the scenario of random origins, i.e., for adoption rate of r, the origin has probability r adopting BGP-iSec and probability $(1-r)$ not adopting BGP-iSec. In this case, the interception rate is simply a weighted average of these two cases.

Fig. 17 shows the results under interception attack. We see that, while as expected, compared to Fig. 7, the benefits of both BGPsec and BGP-iSec are reduced due to the cases when the origin is not adopting. On the other hand, BGP-iSec still significantly outperforms BGPsec. Compared to Fig. 8 (where origins always adopt BGP-iSec), Fig. 17b shows that the gap between adopting and non-adopting ASes is smaller. Fig. 18 plots the results under disconnection DoS attack. We again see that BGP-iSec still significantly outperforms BGPsec, and the gap between adopting and non-adopting ASes is smaller than that when origins always adopt BGP-iSec.

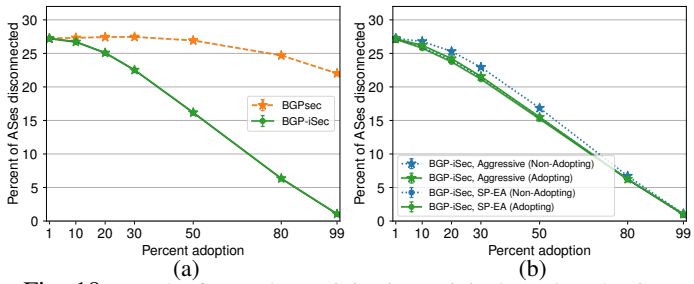


Fig. 18: Results for random origin (i.e., original can be adopting or not adopting based on percentage of adopting) for a disconnection DoS attacker under Full Attacker model: (a) results for all ASes, and (b) results for adopting and non-adopting ASes separately.

Appendix D
Definition of $TBSby_X(A)$ and the $revert$ function

Definition 4 ($TBSby_X(A)$)**.** *Let A be an announcement received by AS Y and let X by an AS in the AS-path of A. Let $TBSby_X(A)$ be a string encoding the set of pairs $\{(\theta, revert^Y_\theta(X,A))\}$, in alphabetic order, for all integrity-protected attributes θ in A, where $revert^Y_\theta(X,A)$ is the function defined in Equation 1. Integrity-protected attributes include (1) the AS-path attribute, (2) the OTC attribute, (3) the UP-image attribute and (4) all fixed attributes.*

To define $revert^Y_\theta(X,A)$ we use the notation $A[\theta]$ for the value of the θ attribute in announcement A, where $A[\theta] = \perp$ if there is no θ attribute in A, and use $+\!\!+$ to denote concatenation. We expect the definition can be extended as necessary to support many other transitive attributes.

$$revert^Y_\theta(X,A) \equiv \begin{cases} \perp \text{ if } (A[\theta] = \perp \vee X \notin A[\text{'AS-path'}]) \\ \perp \text{ if } \left(\begin{array}{c} \left(\begin{array}{c} \theta = \text{'OTC'} \vee \\ (\exists x)\theta = (\text{'UP-image'}, x) \end{array} \right) \wedge \\ X \text{ precedes } A[\text{'OTC'}] \text{ in AS-path} \end{array} \right) \\ \text{else, if } \left(\begin{array}{c} \theta = \text{'AS-path'} \wedge X \\ \text{is last AS in path} \end{array} \right) : \\ \quad : \text{'Y-'} +\!\!+ A[\text{'AS-path'}] \\ \text{else, if } \theta = \text{'AS-path'} : \\ \quad : \left(\begin{array}{c} \text{The AS-path up to} \\ \text{the AS following } X \end{array} \right) \\ \text{else, if } \left(\begin{array}{c} (\exists x)\theta \in \{\text{'OTC'}, (\text{'UP-image'}, x)\} \\ \vee\ \theta \text{ is fixed} \end{array} \right) : \\ \quad : A[\theta] \\ \text{otherwise: } \perp \end{cases} \tag{1}$$

Let $TBSby_X(A) = \perp$ when X is not present in the AS-path of A.

Appendix E
Proof of Theorem 1

Note that we present proof-sketches. We believe that it is possible to convert these into rigorous, reduction-based proofs, using an appropriate specifications framework [102].

Proof of part 1 (announcement integrity against MitM). Consider an announcement A received, and not dropped, by a benign BGP-iSec-deploying AS, Y. Let X be a benign BGP-iSec-adopting AS on the AS-path of A. From the KAPK assumption, Y knows that X deploys BGP-iSec, and Y has the correct public key of X. Hence, Y validates the signature of X on $TBSby_X(A)$; and since A was not dropped, then the signature must be valid. From the security of the signature scheme and since X is benign, it follows that X has signed $TBSby_X(A)$, upon exporting an announcement A' whose integrity-protected attributes were *identical* to these in $TBSby_X(A)$, i.e., for every reversible attribute θ holds $A'[\theta] = A[\theta]$. Namely, announcement A is *integrity-valid* and hence BGP-iSec ensures announcement integrity (against MitM, since we did not restrict the adversary's manipulation of communication). □

Proof of part 2 (no false positives under full deployment). We prove by induction on the location of the first benign adopting AS Y that drops imported announcement A, where all ASes in the AS-path of A are benign. Assume, to the contrary, that Y drops announcement A received from the benign BGP-iSec-adopting AS X, where A was received as

sent by X (not corrupted by the MitM). However, this means that A contains valid signatures by all previous ASes, and therefore Y will not discard it due to the signature validation.

It remains to show that Y will not discard A due to the route-leakage defenses. This easily follows. First, since both X and Y deploy the OTC attribute correctly, then surely Y will not discard A due to the OTC being set on a message from a peer/customer. Second, since X validated the UP attributes from previous ASes along the path, then these would also be valid when checked by Y. Third, X may add its own UP attribute - if it is a customer of Y - but then it would also include the UP preimage, and therefore Y will not discard A due to the validation of the UP attribute. Fourth, X also performed the ProConID validation (subsubsection III-B3) for all ASes before X along the path, therefore, the same validation would also succeed for Y. Finally, since X and Y are benign and BGP-iSec-deploying, if X is a customer of Y then Y appears in the ProConID-list of X, therefore, A would not be discarded by the ProConID validation by Y. □

PROOF OF PART 3 (PREVENTION OF ROUTE LEAKS UNDER FULL DEPLOYMENT). We prove by induction on the location of the first benign BGP-iSec-adopting AS Z that does not drop an imported announcement A received from a customer or a peer, in spite of A being a route-leak, namely, the AS-path of A contains an AS X followed by AS Y, where Y is a customer or peer of X, and X, Y or both are benign. If X is benign, and as assumed it is also BGP-iSec-adopting, and Y is a customer or peer of X, then X would add the OTC attribute, and from the announcement integrity property, Z would discard A (since it receives A from a customer/peer in spite of the OTC); and similarly if Y is benign. □

PROOF OF PART 4 (NO FALSE POSITIVES AGAINST THE FULL ADVERSARY). We prove by induction on the location of the first benign adopting AS Y that drops imported announcement A, where all ASes in the AS-path of A are benign. Assume, to the contrary, that Y drops announcement A received from the benign AS X. Note that here we know that all announcements including A, are received as sent, since the adversary is full, not MitM. However, since all ASes along the path are benign, and benign ASes forward all transitive attributes as received, it follows that Y receives all transitive attributes sent by the last BGP-iSec-adopting AS along the path, which we denote as X'; and recall that all ASes along the path are benign, i.e., X' is also benign. The proof follows as in part (2). □

PROOF OF PART 5 (PREVENTION OF VISIBLE ROUTE LEAKS AGAINST THE FULL ADVERSARY). We prove by induction on the location of the first benign BGP-iSec-adopting AS Z that does not drop an imported announcement A received from a customer or a peer, in spite of A being a visible route-leak. Namely, the AS-path of A contains an AS X followed by AS Y, where Y is a customer or peer of X, and either (1) X is benign and BGP-iSec adopting or (2) the path from X to Z contains a benign BGP-iSec-adopting AS Y' before any non-benign AS X'.

Case (1) is already covered by part 3; it remains to consider case (2). However, case (2) contradicts the induction hypothesis. □

APPENDIX F
EXAMPLE OF UNPREVENTABLE ROUTE LEAK

Fig. 19 shows an example of unpreventable route leak.

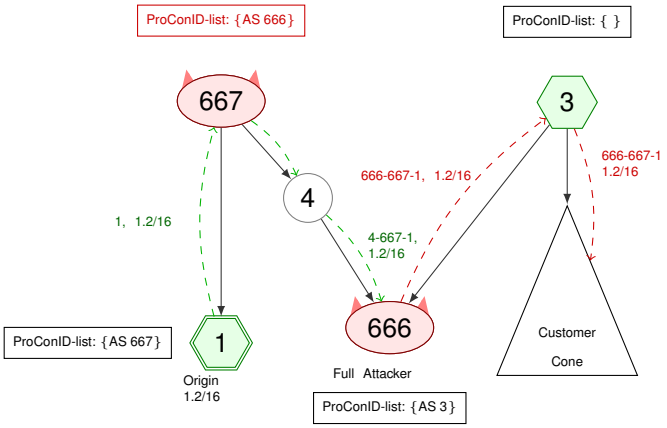


Fig. 19: Here, AS 667 includes 666 in its ProConID-list so the announcement can travel downward to 666 via non-adopting AS 4 and then back upward to AS 3 without being detected. Since 4 is non-adopting, it can be removed from the AS path, however, colluding ASes controlled by the attacker can always skip over any ASes between them by directly connecting through a VPN tunnel.