Pretty Secure BGP (psBGP)*

Tao Wan Evangelos Kranakis P.C. van Oorschot

{twan, kranakis, paulv}@scs.carleton.ca

School of Computer Science, Carleton University, Ottawa, Canada.

Abstract

The Border Gateway Protocol (BGP) is an IETF standard inter-domain routing protocol on the Internet. However, it is well known that BGP is vulnerable to a variety of attacks, and that a single misconfigured or malicious BGP speaker could result in large scale service disruption. We first summarize a set of security goals for BGP, and then propose Pretty Secure BGP (psBGP) as a new security protocol achieving these goals. psBGP makes use of a centralized trust model for authenticating Autonomous System (AS) numbers, and a decentralized trust model for verifying the propriety of IP prefix origination. We compare psBGP with S-BGP and soBGP, the two leading security proposals for BGP. We believe psBGP trades off the strong security guarantees of S-BGP for presumed-simpler operations, while requiring a different endorsement model: each AS must select a small number (e.g., one or two) of its peers from which to obtain endorsement of its prefix ownership assertions. This work contributes to the ongoing exploration of tradeoffs and balance between security guarantee, operational simplicity, and policies acceptable to the operator community.

1. Introduction and Motivation

The Internet consists of a number of Autonomous Systems (ASes), each of which consists of a number of routers under a single technical administration (e.g., sharing the same routing policy). The Border Gateway Protocol (BGP) [35] is an IETF standard inter-domain routing protocol for exchanging routing information between ASes on the Internet. It is well-known that BGP has many security vulnerabilities [24, 30], for example: AS numbers and BGP speakers (routers running BGP) can be spoofed; BGP update messages can be tampered with; and false BGP update messages can be spread. One serious problem is that a single misconfigured or malicious BGP speaker may poison the routing tables of many other well-behaved BGP speakers by advertising false routing information (e.g., see [10]). Examples of consequences include denial of service (i.e., legitimate user traffic cannot get to its ultimate destinations) and man-in-the-middle attacks (i.e., legitimate user traffic is forwarded through a router under the control of an adversary).

Many solutions [38, 24, 26, 15, 41, 2, 20] have been proposed for securing BGP. S-BGP [23, 24] is one of the earliest security proposals, and probably the most concrete one. S-BGP makes use of strict hierarchical public key infrastructures (PKIs) for both AS number authentication and IP prefix ownership verification (i.e., verifying which blocks of IP addresses are assigned or delegated to an AS). Besides computational costs, many people consider S-BGP to be impractical because of the viewpoint that requiring strict hierarchical PKIs makes it difficult to deploy across the Internet (e.g., [3]). It has been suggested that the centralized PKI model of S-BGP counters the distributed trust model adopted by inter-domain routing where each AS is free to choose which other ASes to trust. Our viewpoint is that the matters on which trust is required of S-BGP PKIs differ from those for inter-domain routing, and in fact, the purpose for which a PKI is used in S-BGP is indeed appropriate, at least in theory. In S-BGP, the roots of the PKIs are trusted for their authority of AS numbers and the IP address space. On the other hand, regarding trust in inter-domain routing, one AS might trust another AS for forwarding its traffic but not for its authority of AS numbers and the IP address space. Therefore, the centralized PKI model in S-BGP appears to match its purpose well. However, further analysis suggests that while it might be practical to build a centralized PKI for authenticating AS numbers, it is difficult to build such an

*This paper appears in the *Proceedings of the 12th Annual Network and Distributed System Security Symposium*, San Diego, USA. February 3-4, 2005. ©ISOC.

infrastructure for tracing how IP addresses are allocated and delegated, as explained below.

Agreeing in part with an important design decision made in S-BGP, we suggest that it is practical to build a centralized PKI for AS number authentication because: 1) the roots of the PKI are the natural trusted authorities for AS numbers, i.e., the Internet Assigned Number Authority (IANA) or the Internet Corporation of Assigned Numbers and Names (ICANN) and the Regional Internet Registries (RIRs), hereafter IANA; and 2) the number of ASes on the Internet and its growth rate are relatively manageable, making PKI certificate management feasible. For example, based on the BGP data collected by the RouteViews project [29], there are in total about 17 884 ASes on the Internet as of August 1, 2004. This number has grown by an average of 190 (157 removed and 347 added) per month since January 1, 2004.

However, it would appear to be difficult to build a centralized PKI for verifying IP prefix ownership given the complexity, if not impossibility, of tracing how existing IP address space is allocated and delegated, and tracing all changes of IP address ownership. This is in part due to the large number of prefixes in use and frequent organization changes (e.g., corporations splitting, merging, bankruptcy, etc.). As pointed by Aiello et al. [2], it is exceptionally difficult to even approximate an IP address delegation graph for the Internet. Therefore, it may well be impossible to build a centralized PKI mirroring such a complex and unknown delegation structure. To quote from a study by Atkinson and Floyd [3] on behalf of the Internet Architecture Board (IAB): "*a recurring challenge with any form of inter-domain routing authentication is that there is no single completely accurate source of truth about which organizations have the authority to advertise which address blocks*".

In contrast, soBGP [41] proposes use of a web-of-trust model for authenticating AS public keys and a hierarchical structure for verifying IP prefix ownership. While a web-of-trust model has strong proponents for authenticating user public keys within the technical PGP community [42], it is not clear if it is suitable for authenticating public keys of ASes which are identified by AS numbers strictly controlled by IANA; thus it is questionable if any entity other than IANA should be trusted for signing AS public key certificates. With respect to IP prefix ownership verification, soBGP makes use of a strictly hierarchical structure similar to that of S-BGP. Prefix delegation structures might be simplified in soBGP by using ASes instead of organizations, however, it is not clear if it is practical to do so since IP addresses are usually delegated to organizations not to ASes [2]. We suggest that soBGP, like S-BGP, also faces difficulty in tracing changes of IP address ownership in a strict hierarchical way. Thus, both S-BGP and soBGP have made architectural design choices which arguably lead to practical difficulties.

1.1. Our Contributions

In this paper, we present a new proposal for securing BGP, namely Pretty Secure BGP (psBGP), based on our analysis of the security and practicality of S-BGP and soBGP, and in essence, combining their best features. Our objective is to explore alternative policies and tradeoffs to provide a reasonable balance between security and practicality. psBGP makes use of a centralized trust model for authenticating AS numbers, and a decentralized trust model for verifying IP prefix ownership. One advantage of psBGP is that apparently it can successfully defend against threats from uncoordinated, misconfigured or malicious BGP speakers in a practical way. The major architectural highlights of psBGP are as follows (see §3 for other details and Table 2 in §5 for a summary comparison).

1) psBGP makes use of a *centralized trust model* for AS number authentication. Each AS obtains a public key certificate from one of a number of the trusted certificate authorities, e.g., RIRs, binding an AS number to a public key. We suggest that such a trust model provides best possible authorization of AS number allocation and best possible authenticity of AS public keys. Without such a guarantee, an attacker may be able to impersonate another AS to cause service disruption.

2) psBGP makes use of a *decentralized trust model* for verifying the propriety of IP prefix ownership. Each AS creates a *prefix assertion list* consisting of a number of bindings of an AS number and prefixes, one for itself and one for each of its peering ASes. A prefix ownership assertion made by an AS is *proper* if it is consistent with the assertion made by one of its asserting peers. In this way, we distribute the difficult task of tracing IP address ownership across all ASes on the Internet. On the other hand, psBGP requires that each AS must select a small number of peers (e.g., one or two) from which to obtain endorsement of its prefix ownership assertions. This new endorsement model might require a new communication path between two peers if such path does not already exist. Assuming reasonable due diligence in tracking IP address ownership of direct peers, and assuming no two ASes in collusion (see discussion in §3.4.1), a single misbehaving AS originating improper prefixes will be detected because they will cause inconsistency with prefix assertions made by its

$\mathbb{S}, s_i$	$\mathbb{S}$ is the complete AS number space; currently $\mathbb{S} = \{1, \ldots, 2^{16}\}$. s_i is an AS number; $s_i \in \mathbb{S}$.
$\mathbb{P}, f_i$	$\mathbb{P}$ is the complete IP address space. f_i is an IP prefix which contains a range of IP addresses; $f_i \subset \mathbb{P}$.
T	an authority with respect to $\mathbb{S}$ and $\mathbb{P}$, e.g., $T \in RIRs$.
p_k	$p_k = [s_1, s_2, \ldots, s_k]$ is an AS_PATH; s_1 is the first AS inserted onto p_k.
m	$m = (f_1, p_k)$ is a BGP route (a selected part of a BGP UPDATE message).
$peer(s_i)$	a set of ASes with which s_i establishes a BGP session on a regular basis. More specifically, a given AS s_i may have many BGP speakers, each of which may establish BGP sessions with speakers from many other ASes. $peer(s_i)$ is the set of all other such ASes.
$k_A, \overline{k_A}$	one of A's public and private key pairs.
$\{m\}_A$	digital signature on message m generated with A's private key $\overline{k_A}$.
$(k_A, A)_{k_B}$	a public key certificate binding k_A to A, signed by B using $\overline{k_B}$.
$(k_A, A)_B$	equivalent to $(k_A, A)_{k_B}$ when the signing key is not the main focus.
$(f_i, s_i)_A$	a prefix assertion made by A that s_i owns f_i.
f_i^A, f_i^B	possible different prefixes asserted by A and B related to a given AS.

Table 1. Notation

asserting peers.

The rest of the paper is organized as follows. Section 2 defines notation, overviews BGP, discusses BGP threats, and summarizes BGP security goals. psBGP is presented and analyzed in Sections 3 and 4 respectively. Comparison of S-BGP, soBGP, and psBGP is given in Section 5. Preliminary performance analysis of psBGP is presented in Section 6. A brief review of related work is given in Section 7. We conclude in Section 8.

2. BGP Security Threats and Goals

Here we define notation, give a brief overview of BGP, discuss BGP security threats, and summarize a number of security goals for BGP.

2.1. Notation

A and B denote entities (e.g., an organization, an AS, or a BGP speaker). X or Y denotes an assertion which is any statement. An assertion may be *proper* or *improper*. We avoid use of the term *true* or *false* since in BGP, it is not always clear that a statement is 100% factual or not. An assertion is proper if it conforms to the rules governing the related entity making that assertion. Table 1 defines notation used in this paper.

2.2. Overview of BGP

Conceptually, a routing network can be abstracted as a graph, where a vertex is a router and an edge is a network link. If a network consists of a small (e.g., several) or medium (e.g., tens or hundreds) number of routers, a single routing protocol is probably capable of exchanging and maintaining routing information in that network. Since there are a large number of routers (e.g., hundreds of thousands or more) on the Internet, any single routing protocol currently available probably cannot scale to that size. As a result, a hierarchical routing approach has been used for the Internet. Internet routing protocols can be classified as *intra-domain* (used within an AS) or *inter-domain* (used between ASes).

BGP is an inter-domain routing protocol based on a distance vector approach. A BGP speaker establishes a session over TCP with each of its direct neighbors, exchanges routes with them, and builds routing tables based on the routing information received from them. Unlike a simple distance vector routing protocol (e.g., RIP [17]) where a route has a simple metric (e.g., number of hops), a BGP route is associated with a number of attributes and routes are selected based on local routing policy. One notable route attribute is AS_PATH, which consists of a sequence of ASes traversed by this route. BGP is often considered as a path vector routing protocol.

ASes on the Internet can be roughly classified into three categories: a *stub-AS* has only one connection to other ASes; a *multihomed-AS* has more than one connection to other ASes, but is not designed to carry traffic for other ASes (e.g., for the purpose of load balance or redundancy); and a *transit-AS* has more than one connection to other ASes, and is designed to carry traffic for others.

While a stub-AS may have only one BGP speaker, a multihomed or a transit-AS often has more. A BGP session between two BGP speakers located within two different ASes is often referred to as external-BGP (eBGP), and a BGP session between two BGP speakers within a common AS is often referred to as internal-BGP (iBGP). An eBGP speaker actively exchanges routing information with an external peer by importing and exporting BGP routes. An iBGP speaker only helps propagate routing updates to other BGP speakers within a common AS, and it does not make any changes to a routing update.

A BGP session between two different ASes usually implies one of the following four types of business relationship [13]: *customer-to-provider*, *provider-to-customer*, *peer-to-peer*, and *sibling-to-sibling*. A customer AS usually pays a provider AS for accessing the rest of the Internet. Two peer ASes usually find that it is mutually beneficial to allow each other to have access to their customers. Two sibling ASes are usually owned by a common organization and allow each other to have access to the rest of the Internet.

2.3. BGP Security Threats

BGP faces threats from both BGP speakers and BGP sessions. A misbehaving BGP speaker may be misconfigured (mistakenly or intentionally), compromised (e.g., by exploiting software flaws), or unauthorized (e.g., by exploiting a BGP peer authentication vulnerability). A BGP session may be compromised or unauthorized. We focus on threats against BGP control messages without considering those against data traffic (e.g., malicious packet dropping). Attacks against BGP control messages include, for example, modification, insertion, deletion, exposure, and replaying of messages. In this paper, we focus on modification and insertion (hereafter *falsification* [4]) of BGP control messages; deletion, exposure and replaying are beyond the scope of this paper. Deletion appears indistinguishable from legitimate route filtering. Exposure might compromise confidentiality of BGP control messages, which may or may not be a major concern [4]. Replaying is a serious threat, which can be handled by setting expiration time for a message; however it seems challenging to find an appropriate value for an expiration time.

There are four types of BGP control messages: OPEN, KEEPALIVE, NOTIFICATION, and UPDATE. The first three are used for establishing and maintaining BGP sessions with peers, and falsification of them will very likely result in session disruption. As mentioned by Hu et al. [20], they can be protected by a point-to-point authentication protocol, e.g., IPsec [21]. We concentrate on falsification of BGP UPDATE messages (hereafter, we refrain from capitalizing update as UPDATE) which carry inter-domain routing information and are used for building up routing tables.

A BGP update message consists of three parts: withdrawn routes, network layer reachability information (NLRI), and path attributes (e.g., AS_PATH, LOCAL_PREF, etc.). A route should only be withdrawn by a party which had previously announced that route. Otherwise, a malicious entity could cause service disruption by withdrawing a route which is actually in service. Digitally signing BGP update messages will allow to verify if a party has the right to withdraw a route. Further discussion is beyond the scope of the present paper.

NLRI consists of a set of IP prefixes sharing the same characteristics as described by the path attributes. NLRI is falsified if an AS originates a prefix not owned by that AS, or aggregated improperly from other routes. Examples of consequences include denial of service and man-in-the-middle attacks. There are two types of AS_PATH: AS_SEQUENCE or AS_SET. An AS_PATH of type AS_SEQUENCE consists of an ordered list of ASes traversed by this route. An AS_PATH of type AS_SET consists of an unordered list of ASes, sometimes created when multiple routes are aggregated. Due to space limitations, we focus on the security of AS_SEQUENCE. (Note: AS_SET is less widely used on the Internet. For example, as of August 1, 2004, only 23 of 17 884 ASes originated 47 of 161 796 prefixes with AS_SET.) An AS_PATH is falsified if an AS or any other entity illegally operates on an AS_PATH, e.g., inserting a wrong AS number, deleting or modifying an AS number on the path, etc. Since AS_PATH is used for detecting routing loops and used by route selection processes, falsification of AS_PATH can result in routing loops or selecting routes not selected otherwise. We are interested in countering falsification of NLRI and AS_PATH. We assume there are multiple non-colluding misbehaving ASes and BGP speakers in the network, which may have legitimate cryptographic keying materials. This non-colluding assumption is also made by S-BGP and soBGP, explicitly or implicitly.

2.4. BGP Security Goals

We seek to design secure protocol extensions to BGP which can resist the threats as discussed above. As with most other secure communication protocols, BGP security goals must include data origin authentication and data integrity. In addition, verification of the propriety of BGP messages is required to resist falsification attacks. Specifically, the propriety of NLRI and AS_PATH should be verified. All verification will be performed most likely by a BGP speaker online, but possibly by an operator offline . We summarize five security goals for BGP (cf. [23, 24]). G1 and G2 relate to data origin authentication, G3 to data integrity, and G4 and G5 to the propriety of BGP messages.

G1. *(AS Number Authentication)* It must be verifiable that an entity using an AS number s_i as its own is in fact an authorized representative of the AS to which a recognized AS number authority assigned s_i.

G2. *(BGP Speaker Authentication)* It must be verifiable that a BGP speaker, which asserts an association with an AS number s_i, has been authorized by the AS to which s_i was assigned by a recognized AS number authority.

G3. *(Data Integrity)* It must be verifiable that a BGP message has not been illegally modified en route.

G4. *(Prefix Origination Verification)* It must be verifiable that it is proper for an AS to originate an IP prefix. More specifically, it is proper for AS s_1 to originate prefix f_1 if 1) f_1 is owned by s_1; or 2) f_1 is aggregated from a set F of prefixes such that $f_1 \subseteq F$, i.e., $\forall f_x \subseteq f_1, f_x \subseteq F$[1].

G5. *(AS Path Verification)* It must be verifiable that an AS_PATH ($p_k = [s_1, s_2, \ldots, s_k]$) of a BGP route m consists of a sequence of ASes actually traversed by m in the specified order, i.e., m originates from s_1, and has traversed through $s_2, \ldots, s_k$ in order.

3. Pretty Secure BGP (psBGP)

psBGP makes use of a centralized trust model for authenticating AS numbers and AS public keys. RIRs are the root trusted certificate authorities. Each AS s is issued a public key certificate (ASNumCert), signed by one of the RIRs, denoted by $(k_s, s)_T$. An AS with an ASNumCert $(k_s, s)_T$ creates and signs two data structures: a SpeakerCert $(k'_s, s)_{k_s}$ binding a public key k'_s to s; and a *prefix assertion list* (PAL), listing prefix assertions made by s about the prefix ownership of s and s's peers. PAL_s is an ordered list: the first assertion is for s itself and the rest are for each of s's peers ordered by AS number. Figure 1 illustrates the certificate structure used in psBGP (see also §3.4.1 re: MultiASCert). We next describe psBGP with respect to five security goals, corresponding to G1-G5 above.

[1]If s_1 does not own f_1 and $\exists f_x \subseteq f_1$ such that $f_x \nsubseteq F$, then s_1 *overclaims* IP prefixes, which is considered to be a type of falsification.

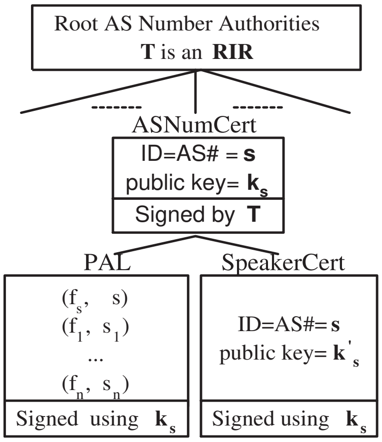

Figure 1. psBGP Certificate Structure

3.1. AS Number Authentication in psBGP

Following S-BGP, we make use of a centralized PKI [37] for AS number authentication, with four root Certificate Authorities (CAs), corresponding to the four existing RIRs. When an organization B applies for an AS number, besides supplying documents currently required (e.g., routing policy, peering ASes, etc.), B additionally supplies a public key, and should be required to prove the possession of the corresponding private key [37, 1]. When an AS number is granted to B by an RIR, a public key certificate (ASNumCert) is also issued, signed by the issuing RIR, binding the public key supplied by B to the granted AS number. An AS number s is called *certified* if there is a valid ASNumCert $(k_s, s)_T$, binding s to a public key k_s signed by one of the RIRs.

The proposed PKI for authenticating AS numbers is practical for the following reasons. 1) The roots of the proposed PKI are the existing trusted authorities of the AS number space, removing a major trust issue which is probably one of the most difficult parts of a PKI. The root of a PKI must have control over the name space involved in that PKI. Thus, RIRs are the natural and logical AS number certificate authorities, though admittedly non-trivial (but feasible) effort might be required for implementing such a PKI. 2) The number of ASes on the Internet and its growth rate are relatively manageable (see §6 - Table 3). Considering there are four RIRs, the overhead of managing ASNumCerts should certainly be feasible as large PKIs are currently commercially operational [16].

To verify the authenticity of an ASNumCert, an AS must have the trusted public key (or certificate) of the signing RIR. These few root trusted public key certificates can be distributed using *out-of-band* mechanisms. ASNumCerts can be distributed with BGP update messages. An ASNumCert is revoked when the corresponding AS number is not used or reassigned to another organization. Issues of revocation, though extremely important, are beyond the scope of the present paper; we restrict comment to the observation that revocation is a well-studied issue, if albeit still challenging (e.g., see [1]). So far, we assume that every AS has the public key certificates of RIRs and can obtain the ASNumCerts of any other ASes if and when necessary.

There is much debate on the architecture for authenticating the public keys of ASes in the BGP security community, particularly on the pros and cons of using a strict hierarchical trust model vs. a distributed trust model, e.g., a web-of-trust model. We make use of a strict hierarchical trust model (with depth of one) for authenticating AS numbers and their public keys to provide a strong guarantee of security. Therefore, it will be difficult for an attacker to spoof an AS as long as it cannot compromise or steal the private key corresponding to the public key of an ASNumCert signed by an RIR or the signing key of an RIR. In contrast, a web-of-trust model does not provide such a guarantee. Some other issues that arise with a web-of-trust model might include: trust bootstrapping, trust transitivity, and vulnerability to a single misbehaving party [28, 36].

3.2. BGP Speaker Authentication in psBGP

An AS may have one or more BGP speakers. A BGP speaker must be authorized by an AS to represent that AS to establish a peer relationship with another AS. In psBGP, an AS with a certified ASNumCert issues an operational public key certificate shared by all BGP speakers within the AS, namely SpeakerCert. A SpeakerCert is signed using the private key of the issuing AS, corresponding to the public key in the AS's ASNumCert (see Figure 1). A SpeakerCert is an assertion made by an AS that a BGP speaker with the corresponding private key is authorized to represent that AS. SpeakerCerts can be distributed with BGP update messages.

We consider three design choices for BGP speaker authentication: 1) each BGP speaker is issued a unique public key certificate; 2) group signatures (e.g., see [8]) are used, i.e., each BGP speaker has a unique private key but shares a common public key certificate with other speakers in the same AS; or 3) all BGP speakers in a given AS share a common public-private key pair. We propose the latter primarily for its operational simplicity. Choice 1) provides stronger security but requires more certificates, and discloses BGP speaker identities. Such disclosure may or may not introduce competitive security concerns [40]. Choice 2) provides stronger security, requires the same number of certificates, and does not disclose BGP identities, but involves a more complex system.

The private key corresponding to the public key of a SpeakerCert is used for establishing secure connections with peers (§3.3), and for signing BGP messages. Therefore, it must be stored in the communication device associated with a BGP speaker. In contrast, since the private key corresponding to the public key of an ASNumCert is only used for signing a SpeakerCert and a PAL, it need not be stored in a BGP speaker. Thus, compromising a BGP speaker only discloses the private key of a SpeakerCert, requiring revocation and reissuing of a SpeakerCert, without impact on an ASNumCert. This separation of ASNumCerts from SpeakerCerts provides a more conservative design (from a security viewpoint), and distributes from RIRs to ASes the workload of certificate revocation and reissuing resulting from BGP speaker compromises. In summary, an ASNumCert must be revoked if the corresponding AS number is re-assigned or the corresponding key is compromised. A SpeakerCert must be revoked if a BGP speaker in that AS is compromised, or for other reasons (e.g., if the private key is lost).

3.3. Data Integrity in psBGP

To protect data integrity, BGP sessions between peers must be protected. Following S-BGP and soBGP, psBGP uses IPsec Encapsulating Security Payload (ESP) [22] with null encryption for protecting BGP sessions. Since many existing BGP speakers implement TCP MD5 [18] with manual key configurations for protecting BGP sessions, it must be supported by psBGP as well. In psBGP, automatic key management techniques can be implemented to improve the security of TCP MD5 as each BGP speaker has a public-private key pair (common to all speakers in that AS).

3.4. Verification of Prefix Origin in psBGP

When an AS s_i originates a BGP update message $m = (f, [s_i, \dots])$, another AS needs to verify if it is proper for s_i to originate a route for a prefix f. As stated in §2.4 (G4), it is proper for s_i to originate a route for prefix f if: 1) s_i owns f; or 2) s_i aggregates f properly from a set F of prefixes carried by a set of routes s_i has received, possibly combined with some prefixes owned by s_i.

3.4.1. Verification of Prefix Ownership in psBGP

Facing the difficulty of building an IP address delegation infrastructure (recall §1), we propose a *decentralized* approach for verifying the propriety of IP address ownership, and more specifically by using *consistency checks*. Our approach is inspired by the way humans acquire their trust in the absence of a trusted authority: by corroborating information from multiple sources (hopefully independent).

In psBGP, each AS s_i creates and signs a *prefix assertion list* (PAL_{s_i}), consisting of a number of tuples of the form (IP prefix list, AS number), i.e., $PAL_{s_i} = [(f_i^{s_i}, s_i), (f_1^{s_i}, s_1), \dots, (f_n^{s_i}, s_n)]$, where for $1 \leq j \neq i \leq n, s_j \in peer(s_i)$ and $s_j < s_{j+1}$. The first tuple $(f_i^{s_i}, s_i)$ asserts that s_i owns $f_i^{s_i}$; the rest are sorted by AS number, and assert the prefix ownership of s_i's peers. $(f_j^{s_i}, s_j)$ $(s_j \neq s_i)$ asserts by s_i that s_j is a peer of s_i and s_j owns prefix $f_j^{s_i}$ if $f_j^{s_i} \neq \phi$. Otherwise, it simply asserts that s_j is a peer of s_i.

As a new requirement in psBGP, each AS is responsible for carrying out some level of due diligence offline: for the safety of that AS and of the whole Internet, to determine what IP prefixes are delegated to each of its peers. We suggest the effort required for this is both justifiable and practical, since two peering ASes usually have a business relationship (e.g., a traffic agreement) with each other, allowing offline direct interactions. For example, s_i may ask each of its peer s_j to show the proof that f_j is in fact owned by s_j. Publicly available information about IP address delegation may also be helpful.

Two assertions $(f_i, s_i), (f'_i, s'_i)$ made by two ASes are *comparable* if they assert the prefix ownership of a given AS, i.e., $s_i = s'_i$ and the asserted prefixes are non-empty, i.e., $f_i, f'_i \neq \phi$; and are *incomparable* otherwise, i.e., they assert the prefix ownership of different ASes or one of the asserted prefixes is an empty set. Two comparable assertions (f_i, s_i) and (f'_i, s_i) are *consistent* if $f_i = f'_i$; and are *inconsistent* if $f_i \neq f'_i$.

Let n be the number of s_i's peers. (f_i, s_i) is *k-proper* if there exist some fixed number k $(2 \leq k \leq n+1)$ of consistent assertions of (f_i, s_i) made by s_i or s_i's peers. Requiring $k = n + 1$ means that the assertion (f_i, s_i) made by s_i and all of its peers must be consistent for (f_i, s_i) to be k-proper; this provides maximum confidence in the correctness of (f_i, s_i) if the condition is met. However, it is subject to attacks by a single misbehaving AS. For example, if $\exists s_j \in peer(s_i)$, and s_j makes a false assertion $(f_i^{s_j}, s_i)$ inconsistent with $(f_i^{s_i}, s_i)$, then $(f_i^{s_i}, s_i)$ will not be verified as k-proper, although it might indeed be proper. From the perspective of assertion list management, the greater k is, the larger prefix assertion lists will grow, and the more updates of prefix assertion lists will be required since a change to an AS number s_i or a prefix f_i requires the update of all PALs making an assertion about s_i or f_i. Moreover, there are a large number of ASes which might have only one peer. For example, as of August 1, 2004, there were 6619 ASes which have only one peer based on one BGP routing table collected from the RouteViews project [29]. Requiring $k \geq 3$ will prevent these ASes from originating authorized prefixes.

To begin with, we suggest $k = 2$ in psBGP, i.e., $(f_i^{s_i}, s_i)$ is *proper* if there exists any single $s_j \in peer(s_i)$ such that s_j make an assertion $(f_i^{s_j}, s_i)$ which is consistent with $(f_i^{s_i}, s_i)$. When verifying $(f_i^{s_i}, s_i)$, an AS checks its consistency with the prefix assertion related to s_i made by each of s_i's peers until a consistent one is found, or no consistent assertion is found after all relevant assertions made by s_i's peers have been checked. In the former case, $(f_i^{s_i}, s_i)$ is verified as proper; in the latter case, it is verified as *improper*. For simplicity, the consistency among the prefix assertions related to s_i made by s_i's peers amongst themselves is not checked. A non-aggregated route $(f, [s_i, \dots])$ originated by s_i is verified as proper if $(f_i^{s_i}, s_i)$ is proper and $f \subseteq f_i^{s_i}$.

We now discuss how psBGP reacts to erroneous prefix assertions (e.g., resulting from human errors, lack of due diligence, or collusion). An AS s_i erroneously asserting the ownership of a prefix will not result in service disruption of the legitimate owner of that prefix as long as none of s_i's asserting peers endorses its assertion. s_i erroneously asserting the prefix ownership of a peer s_j will not result in service disruption of s_j if there exists another peer of s_j which correctly asserts s_j's prefix ownership. If s_i is the only asserting peer for s_j, or more generally, $\forall s_i \in peer(s_j)$, s_i issues $(f_j^{s_i}, s_j)$ inconsistent with $(f_j^{s_j}, s_j)$, $(f_j^{s_j}, s_j)$ will be verified as *improper* by other ASes, even if it might be actually proper. This is the case when misbehaving ASes form a network cut from s_j to any part of the network. It appears difficult, if not impossible, to counter such an attack; however, we note that even if such a denial of service attack could be prevented, many other techniques beyond the control of BGP can also be used to deny the routing service of s_j, e.g., link-cuts [6], filtering, or packet dropping. Note that a prefix assertion made by s_i about a remote AS s_k, i.e., $s_i \notin peer(s_k)$, will not be checked when s_k's own prefix assertion is verified. Thus, a misbehaving AS is unable to mislead other ASes about the prefix ownership of a non-peering AS.

psBGP assumes that no two ASes are in collusion.

Two ASes s_i and s_j are to be in collusion if they assert being a peer of each other, s_i erroneously asserts the ownership of a prefix, and s_j endorses s_i's erroneous prefix assertion. If s_i and s_j are owned and managed by two different organizations, it is very likely that uncoordinated erroneous assertions by s_i and s_j will be inconsistent. Here we discuss two cases where the assumption of no collusion may not hold: 1) s_i and s_j are owned by a common organization; and 2) s_i and s_j are owned by two different organizations which are controlled by the same attacker. In case 1), a multi-AS organization might use a single centralized database to generate router configurations for all of its owned ASes. Thus, it is possible that prefix assertion lists for two peering ASes owned by a common organization are also created from a single centralized database. If a prefix is erroneously entered into such a database, it might end up with two erroneous yet consistent prefix assertion lists. We recommend that an AS should obtain prefix assertion endorsement from another AS owned by a different organization. As a local policy, an AS might mandate to not trust a prefix assertion by AS s_i if it is not endorsed by an AS s_j where s_i and s_j are owned by different organizations. To facilitate the distribution of the knowledge of AS ownership by a multi-AS organization, psBGP makes use of a new certificate, namely MultiASCert, which binds a list of ASes owned by a common organization to the name of that organization, and is signed by an RIR. Prefix assertions by two ASes owned by a common organization (i.e., appearing on a MultiASCert) might not be accepted even if they are consistent. In this way, human errors by a multi-AS organization will not result in service disruption in psBGP. In case 2), if an attacker could set up two organizations and manage to obtain an AS number from an RIR for each of them, the psBGP security, even with MultiASCerts, can be defeated.

3.4.2. Verification of Aggregated Prefixes

Suppose s_i owns IP prefix f_i. When receiving a set of routes with a set of prefixes $F = \{f_j\}$, the BGP specification [35] allows s_i to aggregate F into a prefix f_g to reduce routing information to be stored and transmitted. We call f_j a *prefix to be aggregated*, and f_g an *aggregated prefix*. s_i can aggregate F into f_g if one of the following conditions holds: 1) $\forall f_j \subseteq f_g, f_j \subseteq f_i$; or 2) $\forall f_j \subseteq f_g, f_j \subseteq F \cup f_i$.

In case 1), s_i must own f_i which is a superset of the aggregated prefix f_g. Most likely, f_i will be the aggregated prefix, i.e., $f_g = f_i$. This type of aggregation is sometimes referred to as prefix *re-origination*. From a routing perspective, prefix re-origination does not have any effect since traffic destined to a more specific prefix will be forwarded to the re-originating AS and then be forwarded to the ultimate destination from there. From a policy enforcement perspective, prefix re-origination does have an effect since the AS_PATH of an aggregated route is different from any of the AS_PATHs of the routes to be aggregated. Since AS_PATH is used by the route selection process, changing AS_PATH has an impact on route selections. From a security perspective, prefix re-origination is no different than normal prefix origination since the aggregated prefix is either the same as, or a subset of, the prefix owned by the aggregating AS. Therefore, the aggregated route f_g can be verified by cross-checking the consistency of s_i's prefix assertion list with those of its peers (§3.4.1).

In case 2), s_i does not own the whole address space of the aggregated prefix f_g. Therefore, f_g cannot be verified in the same way as for prefix re-origination. To facilitate verification of the propriety of route aggregation by a receiving AS, psBGP requires that the routes to be aggregated be supplied by the aggregating AS along with the aggregated route. This approach is essentially similar to that taken by S-BGP. Transmission of routes to be aggregated incurs additional network overhead, which is something BGP tries to reduce. However, we view such additional overhead to be relatively insignificant given that modern communication networks generally have high bandwidth and BGP control messages account for only a small fraction of subscriber traffic. The main purpose of route aggregation is to reduce the size of routing tables, i.e., reducing storage requirements; note that this is preserved by psBGP.

3.5. Verification of AS_PATH in psBGP

There is no consensus on the definition of "AS_PATH security", and different security solutions of BGP define it differently. In S-BGP, the security of an AS_PATH is interpreted as follows: for every pair of ASes on the path, the first AS authorizes the second to further advertise the prefix associated with this path. In soBGP, AS_PATH security is defined as the plausibility of an AS_PATH, i.e., if an AS_PATH factually exists on the AS graph (whether or not that path was actually traversed by an update message in question is not considered).

Since AS_PATH is used by the BGP route selection process, great assurance of the integrity of an AS_PATH increases the probability that routes are selected based on proper information. While the BGP specification [35] does not explicitly state that AS_PATH is used for route selection, it commonly is in practice (e.g., by Cisco IOS). Without the guarantee of AS_PATH integrity, an attacker may be able to modify an AS_PATH is a such way that it is plausible in the AS graph and is also more favored (e.g., with a shorter length) by recipient ASes than the original path. In this way, a recipient AS may be misled to favor the falsified route over any correct routes. As a result, traffic flow might be influenced. Thus, we suggest that it might not be sufficient to verify only the existence/non-existence of an AS_PATH, and it is desirable to obtain greater assurance of the integrity of an AS_PATH; we acknowledge that the cost of any solution should be taken into account as well. While psBGP allows the verification of AS_PATH plausibility, in what follows, we define AS_PATH security according to the original definition of AS_PATH [35], as "an ordered set of ASes a route in the update message has traversed".

We choose the S-BGP approach with the improvement of the bit-vector method by Nicol et al. [32] (see next paragraph) for securing AS_PATH in psBGP, since it fits into the design of psBGP and provides greater assurance of AS_PATH integrity with reasonable overhead. Hu et al. [20] propose a secure path vector protocol (SPV) for protecting AS_PATH using authentication hash trees with less overhead than S-BGP. psBGP does not use the SPV approach since it has different assumptions than psBGP. For example, SPV uses different public key certificates than psBGP.

Let $n_i = |peers(s_i)|$ be the number of peers of s_i. Given $m_k = (f_1, [s_1, s_2, \ldots, s_k])$, a psBGP speaker s_i ($1 \leq i \leq k-1$) generates a digital signature $\{f_1, [s_1, \ldots, s_i], v_i[n_i]\}_{s_i}$ where $v_i[n_i]$ is a bit vector of bit-length n_i, with one bit corresponding to each peer in s_i's prefix assertion list (§3.4.1). If s_i intends to send a routing update to a peer s_j, it sets the bit in $v_i[\ ]$ corresponding to s_j. In this way, a message sent to multiple peers by a BGP speaker need be signed only once. For s_{k+1} to accept m_k, s_{k+1} must receive the following digital signatures: $\{f_1, [s_1], v_1[n_1]\}_{s_1}, \{f_1, [s_1, s_2], v_2[n_2]\}_{s_2}, \ldots$, and $\{f_1, [s_1, s_2, \ldots, s_k], v_k[n_k]\}_{s_k}$.

4. Security Analysis of psBGP

We analyze psBGP against the listed security goals from §2.4. The analysis below clarifies how our proposed mechanisms meet the specified goals, and by what line of reasoning and assumptions. While we believe that mathematical "proofs" of security may often be based on flawed assumptions that fail to guarantee "security" in any real-world sense, they are nevertheless very useful, e.g., for finding security flaws, for precisely capturing protocol goals, and for reducing ambiguity, all of which increase confidence. We thus encourage such formalized reasoning for lack of better alternatives.

Proposition 1 *psBGP provides AS number authentication (G1).*

Proof Outline: For an AS number s to be certified, psBGP requires an ASNumCert $(k_s, s)_T$. Since T controls s, and is the trusted guardian of AS numbers (by assumption), any assertion made by T about s is proper. Thus $(k_s, s)_T$ is proper. In other words, s is an AS number certified by T, and k_s is a public key associated with s certified by T. More formally[2], (T *controls* s) $\wedge$ $(k_s, s)_T \Rightarrow (k_s, s)$ is proper.

Proposition 2 *psBGP provides BGP speaker authentication (G2).*

Proof Outline: For a BGP speaker r to be accepted as an authorized representative of an AS s, psBGP requires an ASNumCert $(k_s, s)_T$, a SpeakerCert $(k'_s, s)_{k_s}$, and evidence that r possesses $\overline{k'_s}$. By Proposition 1, $(k_s, s)_T$ proves that s is an AS number certified by T and k_s is a public key associated with s certified by T. Similarly, $(k'_s, s)_{k_s}$ proves that k'_s is a public key associated with s certified by s. Evidence that r possesses $\overline{k'_s}$ establishes that r is authorized by s to represent s. Thus, the Proposition is proved. More formally, (T *controls* s) $\wedge$ $(k_s, s)_T \Rightarrow (k_s, s)$ is proper; (k_s, s) is proper $\wedge$ $(k'_s, s)_{k_s}$ $\Rightarrow (k'_s, s)$ is proper; (k'_s, s) is proper $\wedge$ r possesses $\overline{k'_s} \Rightarrow$ r is authorized by s.

Proposition 3 *psBGP provides data integrity (G3).*

Proof Outline: psBGP uses the IPsec Encapsulating Security Payload (ESP) [21, 22] with null encryption for protecting BGP sessions, and relies upon IPsec ESP for data integrity.

Before presenting Proposition 4, we establish two Lemmas.

Lemma 1 *Assume that* $\forall s_i \in \mathbb{S}, \exists s_j \in peer(s_i)$ *such that* s_j *carries out reasonable due diligence to create a proper prefix assertion* $(f_i^{s_j}, s_i)$ *(A1); and that no two ASes are in collusion (A2)*[3]*, then psBGP provides reasonable assurance of prefix ownership verification, i.e., a prefix assertion* $(f_i^{s_i}, s_i)$ *that is actually proper will be verified as such; otherwise not.*

Proof Outline: Suppose $(f_i^{s_i}, s_i)$ is proper. Since $\exists s_j \in peer(s_i)$ which makes a proper assertion $(f_i^{s_j}, s_i)$ (by assumption A1), then $(f_i^{s_i}, s_i)$ is consistent with

[2] Here we adapt BAN-like notation, modified for our purpose (cf. [9, 12, 14]).

[3] See §3.4.1 for discussion of examples where this collusion assumption may not hold.

$(f_i^{s_j}, s_i)$ since two proper assertions must be consistent. Thus, $(f_i^{s_i}, s_i)$ will be verified as proper because there exists a prefix assertion from s_i's peer s_j, $(f_i^{s_j}, s_i)$, which is consistent with $(f_i^{s_i}, si)$.

Suppose $(f_i^{s_i}, s_i)$ is improper. To show that $(f_i^{s_i}, s_i)$ will not be verified as proper, we need to show that there does not exist $(f_i^{s_j}, s_i)$, $s_j \in peer(s_i)$, such that $(f_i^{s_j}, s_i)$ is consistent with $(f_i^{s_i}, s_i)$. $\forall (f_i^{s_j}, s_i), s_j \in peer(s_i)$, if s_j carries out due diligence successfully, then $(f_i^{s_j}, s_i)$ is proper and will be inconsistent with the improper $(f_i^{s_i}, s_i)$. If s_j misbehaves or its due diligence fails to reflect actual IP ownership, then $(f_i^{s_j}, s_i)$ is improper. We consider it to be a collusion of s_j and s_i if $(f_i^{s_j}, s_i)$ and $(f_i^{s_i}, s_i)$ are improper but consistent. This case is ruled out by assumption A2. Thus, an improper prefix assertion $(f_i^{s_i}, s_i)$ will be verified as improper since there does not exist an improper assertion which is consistent with $(f_i^{s_i}, s_i)$ without collusion. This establishes Lemma 1.

Lemma 2 *psBGP provides reasonable assurance of IP prefix aggregation verification.*

Proof Outline: Let f_g be a prefix aggregated by AS s_x from a set of routes $\{m_i = (f_i, p_i) | p_i = [s_i, \dots]\}$ received by s_x. psBGP requires that for f_g originated by s_x to be verified as proper, s_x must either own a prefix f_x such that $f_g \subseteq f_x$ (verified by Lemma 1), or provide evidence that s_x has in fact received $\{m_i\}$ and $f_g \subseteq \cup\{f_i\}$. Valid digital signatures from each AS on p_i can serve as evidence that s_x has received $\{m\}$ (see Proposition 5). If $f_g \subseteq \cup\{f_i\}$, then s_x aggregates f_g properly. If s_x cannot provide required evidence, s_x's aggregation of f_g is verified as improper. This establishes Lemma 2.

Proposition 4 *psBGP provides reasonable assurance of IP prefix origination verification, i.e., an AS s_i's origination of a prefix f is verified as proper if f is owned by s_i or is aggregated properly by s_i from a set of routes received by s_i. Otherwise, s_i's origination of f is verified as improper.*

Proof Outline: Lemma 1 allows verification of the propriety of prefix ownership. Suppose $(f_i^{s_i}, s_i)$ is verified as proper, i.e., $f_i^{s_i}$ is verified to be owned by s_i. If s_i owns f, then $f \subseteq f_i^{s_i}$. In psBGP, s_i's origination of f is verified as proper if $f \subseteq f_i^{s_i}$. If $f \nsubseteq f_i^{s_i}$, psBGP requires that s_i provide proof that f is aggregated properly from a set of received routes (see Lemma 2). If s_i does not own f and s_i does not provide proof of the propriety of prefix aggregation, psBGP verifies s_i's origination of f as improper. This establishes Proposition 4.

Proposition 5 *psBGP provides assurance of AS_PATH verification (G5).*

Proof Outline: Let $m_k = (f_1, p_k)$ be a BGP route, where $p_k = [s_1, s_2, \dots, s_k]$). Let r_i $(1 \le i \le k-1)$ be a BGP speaker in s_i which has originated $(i = 1)$ or forwarded $(2 \le i \le k-1)$ m_i to s_{i+1}. In psBGP, the integrity of p_k implies that m_k has traversed the exact sequence of $s_1, s_2, \dots, s_k$. In other words, there does not exist i $(2 \le i \le k-1)$ such that s_{i-1} didn't send $(f_1, [s_1, \dots, s_{i-1}])$ to s_i.

By way of contradiction, assume that it is possible in psBGP that $(f_1, [s_1, \dots, s_k])$ is accepted by a BGP speaker r_{k+1} and there exists i $(2 \le i < k)$ such that s_{i-1} didn't send $(f_1, [s_1, \dots, s_{i-1}])$ to s_i. psBGP requires that for $[s_1, s_2, \dots, s_k]$ to be accepted by r_{k+1}, $\forall i$ $(1 \le i < k)$, r_{i+1} has received a valid digital signature $\{p_1, [s_1, \dots, s_i], v_i[\ ]\}_{s_i}$ where the bit in $v_i[\ ]$ corresponding to s_{i+1} is set. $\{p_1, [s_1, \dots, s_i], v_i[\ ]\}_{s_i}$ serves as a signed assertion that s_i does send that routing update to s_{i+1}. This contradicts the above assumption. Thus, Proposition 5 is established.

The above results establish the desired psBGP security properties, and are summarized by Theorem 1.

Theorem 1 (psBGP Security Property) *psBGP achieves the following five security goals: AS number authentication (G1), BGP speaker authentication (G2), data integrity (G3), IP prefix origination verification (G4), and AS_PATH verification (G5).*

5. S-BGP, soBGP, and psBGP Comparison

We compare the different approaches taken by S-BGP, soBGP, and psBGP for achieving the BGP security goals listed in §2.4. Table 2 provides a summary. We see that psBGP falls somewhere between S-BGP and soBGP in several of the security approaches and architectural design decisions, but makes distinct design choices in several others.

5.1. AS Number Authentication

Both S-BGP and psBGP use a centralized trust model for authenticating AS numbers, which is different from the web-of-trust model used by soBGP. The difference between the AS number authentication of psBGP and S-BGP is that S-BGP follows the existing structure of AS number assignment more strictly than psBGP. In S-BGP, an AS number is assigned by IANA to an organization and it is an organization that creates and signs a certificate binding an AS number to a public key (thus, a two-step chain). In psBGP, an ASNumCert is signed directly by IANA (depth=1), and is independent of the name of

Goal	S-BGP	soBGP	psBGP
G1: AS Number Authentication	centralized (multiple levels)	decentralized (with trust transitivity)	centralized (depth=1)
G2: BGP Speaker Authentication	one certificate per BGP speaker	one certificate per AS	one certificate per AS
G3: Data Integrity	IPsec or TCP MD5	IPsec or TCP MD5	IPsec or TCP MD5
G4: Prefix Origination Verification	centralized (multiple levels)	centralized (multiple levels)	decentralized (no trust transitivity)
G5: AS_PATH Verification	integrity	plausibility	integrity

Table 2. Comparison of S-BGP, soBGP, and psBGP approaches for achieving BGP security goals.

	Jan	Feb	Mar	Apr	May	Jun	Jul	Aug
Start of Month	16 554	16 708	16 879	17 156	17 350	17 538	17 699	17 884
Removed during Month	153	137	155	174	138	179	164	N/A
Added during Month	307	308	432	368	326	342	349	N/A

Table 3. AS Number Dynamics from January 1 to August 1, 2004

an organization. Thus, psBGP has less certificate management overhead than S-BGP, requiring fewer certificates. In addition, some changes in an organization X may not require revoking and reissuing the public key certificate of the AS controlled by X. For example, if X changes its name to Y but the AS number s associated with X does not change, psBGP does not need to revoke the ASNumCert $(k_s, s)_T$. However, in S-BGP, the public key certificates $(k_X, X)_T$, $(k_s, s)_{k_X}$ might be revoked, and new certificates $(k_Y, Y)_T$, $(k'_s, s)_{k_Y}$ might be issued.

5.2. BGP Speaker Authentication

In S-BGP, a public key certificate is issued to each BGP speaker, while both soBGP and psBGP use one common public key certificate for all speakers within one AS. Thus, soBGP and psBGP require fewer BGP speaker certificates (albeit requiring secure distribution of a common private key to all speakers in an AS).

5.3. Data Integrity

S-BGP uses IPsec for protecting BGP session and data integrity. Both soBGP and psBGP adopt this approach. TCP MD5 [18] is supported by all three proposals for backward compatibility. In addition, automatic key management mechanisms can be implemented for improving the security of TCP MD5.

5.4. Prefix Origination Verification

Both S-BGP and soBGP propose a hierarchical structure for authorization of the IP address space; however S-BGP traces how IP addresses are delegated among organizations, while soBGP only verifies IP address delegation among ASes. It appears that soBGP simplifies the delegation structure and requires fewer certificates for verification; however, it is not clear if it is feasible to do so in practice since IP addresses are usually delegated between organizations, not ASes. In psBGP, consistency checks of PALs of direct peers are performed to verify if it is proper for an AS to originate an IP prefix. Therefore, psBGP does not involve verification of chains of certificates (instead relying on offline due diligence). We note that while psBGP does not guarantee perfect security of the authorization of IP address allocation or delegation, as intended by S-BGP and soBGP, as discussed in §1 it is not clear if the design intent in the latter two can actually be met in practice.

5.5. AS_PATH Verification

Both S-BGP and psBGP verify the integrity of AS_PATH based on its definition in the BGP specification [35]. In contrast, soBGP verifies the plausibility of an AS_PATH. Thus, S-BGP and psBGP provide stronger security of AS_PATH than soBGP, at the cost of digital signature operations which might slow down network convergence.

	Jan	Feb	Mar	Apr	May	Jun	Jul	Aug
Start of Month	148 903	148 014	151 174	156 019	157 925	160 818	155 118	161 796
Stable During Month	143 200	144 422	146 139	151 481	153 171	148 280	151 436	N/A
Stable During Jan-Aug	119 968	119 968	119 968	119 968	119 968	119 968	119 968	N/A
Removed During Month	5 703	3 592	5 035	4 538	4 754	12 538	3 682	N/A
Added During Month	4 814	6 752	9 880	6 444	7 647	6 838	10 360	N/A

Table 4. IP Prefix Dynamics from January 1st to August 1st, 2004

# of PA Changes		1	2-4	5-10	11-30	31-60	61-100	101-200	201-300	301-1000	1001-5000	over 5000	Total
n=1	# of ASes	1 497	677	319	152	43	26	19	5	2	1	1	2 742
	(percentage)	(8.3%)	(3.8%)	(1.8%)	(0.8%)	(0.2%)	(0.1%)	(0.1%)	(0%)	(0%)	(0%)	(0%)	(15.2%)
n=2	**# of ASes**	**1 508**	**713**	**346**	**187**	**66**	**21**	**33**	**7**	**8**	**1**	**2**	**2 892**
	(percentage)	**(8.4%)**	**(4.0%)**	**(1.9%)**	**(1.0%)**	**(0.4%)**	**(0.1%)**	**(0.2%)**	**(0%)**	**(0%)**	**(0%)**	**(0%)**	**(16.0%)**
n=3	# of ASes	1 516	725	355	205	70	23	32	13	9		4	2 952
	(percentage)	(8.4%)	(4.0%)	(2.0%)	(1.1%)	(0.4%)	(0.1%)	(0.2%)	(0.1%)	(0%)	(0%)	(0%)	(16.4%)
n=all	# of ASes	1 424	784	387	233	78	34	27	12	14	2	28	3 023
	(percentage)	(7.9%)	(4.3%)	(2.1%)	(1.3%)	(0.4%)	(0.2%)	(0.1%)	(0.1%)	(0.1%)	(0%)	(0.2%)	(16.7%)

Table 5. Projected number of ASes in absolute number, and as percentage of all ASes, requiring the specified number of prefix assertion changes in psBGP, based on July 2004 Data. We recommend row $n = 2$.

6. Performance Analysis of psBGP

Here we present our preliminary esitimates of memory, bandwidth, and CPU overhead, and the analysis of certificate dynamics in psBGP. While rigorous study has been performed by Aiello et al. [2] on the prefix delegation stability on the Internet as a whole, it is desirable to study certificate dynamics of a secure system and to project certificate management overhead on a per AS level. We use BGP data collected by the RouteViews project [29], and retrieved one BGP routing table of the first day of each month from January to August 2004. Despite likely incompleteness of the RouteViews data set, it is one of the most complete data repositories publicly available, and has been widely used in the BGP community.

6.1. Memory Overhead

There are four types of certificates which require extra memory space to store for a BGP speaker to support psBGP. We estimate the memory overhead for each type and then give an estimate of the total. While a BGP update message may carry extra digitally signed data and signatures which need to be stored temporarily, they can be discarded after verfication. Thus, we do not consider their memory overhead here.

ASNumCerts and SpeakerCerts. We observed in total 17 884[4] ASes as of August 1, 2004. One ASNumCert is required per AS. In the worst case, an AS may need to store the ASNumCert of every AS on the Internet; in this case, 17 844 ASNumCerts would be stored. As with S-BGP and soBGP, psBGP makes use of the X.509v3 certificate structure which has wide industrial support. Assuming the average size of a certificate is 600 bytes [25], 10.479M bytes memory would be required for storing 17 844 ASNumCerts. The same holds for SpeakerCerts.

PALs and MultiASCerts. Each AS s_i issues a PAL, whose size is primarily determined by the number of prefixes delegated to s_i, the number of s_i's peers, and the number of prefixes delegated to each of s_i's asserted peers. While some ASes have many peers, and some are delegated many prefixes, many ASes have only a small number of peers and are delegated a small number of prefixes. On average, each AS has 4.2 peers and is delegated 9.1 prefixes. Assuming the average size of a PAL is 1 024 bytes, 17.844M bytes of memory would be required to store 17 844 PALs, one for each AS. For MultiASCerts, a BGP speaker needs to store one certificate for each organization which owns multiple ASes. Based

[4] AS numbers used by IANA itself for experimental purpose are not counted.

on the data from Aiello et al. [2], there are 385 multi-AS organizations which in total own $1\,259$ ASes. On average, each multi-AS organization owns 3.3 ASes. Assuming the average size of a MultiASCert is 600 bytes, 0.226M bytes of memory are required for storing all MultiASCerts.

In summary, a total of 38.028M bytes of memory are required for storing all certificates to support psBGP. However, more efficient certificate distribution mechanisms (e.g., see [1, 25]) may be used; further discussion is beyond the scope of the present paper.

6.2. Bandwidth Overhead

Except for a small number of public key certificates of trusted CAs which need to be distributed using out-of-band mechanisms, all other certificates in psBGP can be distributed with BGP update messages, which consumes extra network bandwidth. However, such overhead is not persistent since those certificites only need to be distributed periodically or upon changes. We suggest that such overhead is of little significance and will not discuss it here.

The primary bandwidth overhead is introduced by digitally signed data and signatures carried by each BGP update message for protecting the message. For a fully protected BGP route where every AS on the route digitally signs the update message, the overhead is mainly determined by the number of such ASes (the average number is 3.7 according to Kent [25]). psBGP also makes use of a bit-vector approach [32] to reduce the number of operations of digital signature generations, where the size of a bit-vector used by an AS is roughly equal to the number of peers of that AS. Thus, more overhead will be added if an AS digitally signing a route has a large number of peers. To compare with S-BGP which uses a 16-bit length AS number instead of a bit-vector, the bandwidth overhead for a given route might be higher in psBGP if some of the ASes on the route have more than 16 peers (a corresponding bit-vector will be larger than 16-bit), and will be lower if all of the ASes have less than 16 peers. Overall, there might not be significant difference between the bandwidth overhead of psBGP and S-BGP. As pointed out by Kent [25], BGP control messages only account for a small fraction of network bandwidth versus subscriber traffic. Thus, from our preliminary analysis, we expect that bandwidth overhead of psBGP will not create difficulty in the deployment of psBGP.

6.3. CPU overhead

A BGP speaker supporting psBGP needs to digitally sign each BGP update message sent to each different set of peers, and to verify each unique digital signature carried by each BGP update message it receives and chooses to use. As shown by Kent et al. [23] in their study of S-BGP performance, such CPU overhead is significant. While the bit-vector approach adopted by psBGP might reduce CPU overhead of digital signature generation to some degree if a BGP speaker usually sends an update message to multiple peers [32], it does not reduce overhead of digital signature verification. Overall, we expect that significant CPU overhead will be generated by psBGP if an AS chooses to maximamlly protect BGP update messages. To mitigate the problem, some approaches might be helpful, such as caching [23], delay of signature verification [23], using a digital signature algorithm with a faster verification operation (e.g., RSA) [32], etc. In addition, since many BGP speakers currently in use might not be capable of performing digital signature operations required to achieve maximum protection of BGP udpate messages, it might be desirable to provide them a less expensive option with less protection (e.g., verification of AS_PATH plausibility but not integirty).

6.4. Certificate Dynamics

ASNumCerts and SpeakerCerts. The monthly number of ASes has grown by an average of 190 since January 1, 2004, with an average of 347 ASes added and 157 ASes removed (see Table 3). When an AS number is added or removed, the corresponding ASNumCert must be issued or revoked by an RIR. Thus, four RIRs between them must issue an average of 347 new ASNumCerts and revoke an average of 157 existing ASNumCerts per month. This would certainly appear to be manageable in light of substantially larger PKIs existing in practice (e.g., see [16]). Note the issuing and revocation of a SpeakerCert is performed by an AS, not an RIR.

Prefix Assertion Lists (PALs). A prefix assertion list PAL_{s_i} must be changed (removed, added, or updated) if: 1) the AS number s_i changes (i.e., removed or added); 2) an IP prefix owned by s_i changes; 3) s_i's peer relationship changes, i.e., a peer is removed or added; or 4) an IP prefix changes which is asserted by s_i for one of its peers. Table 4 depicts the dynamics of prefixes,

We study the number of prefix assertion (PA) changes required for each AS based on the two routing tables of July 1 and August 1, 2004. Each prefix addition or removal is counted once (i.e., resulting in one PA addition or removal) if the AS number of the AS owning that prefix does not change. If an AS number is newly added (or removed) during the month, all additions (or removals) of the prefixes owned by that AS are counted once as a whole.

Table 5 depicts the projected PAL dynamics based on the data set of July 2004. The total number of ASes observed during July 2004 is 18 048, including 17 884 observed on August 1, 2004 and 164 removed during July 2004. We can see that the more asserting peers[5] an AS has, the more PA changes required. We recommend the scenario $n = 2$, where each AS has at most two asserting peers even if it has more than two peers. This provides a level of redundancy in the case that one of the two asserting peers fails to carry out its due diligence.

We see from Table 5 that in the recommended scenario $n = 2$, 16% of the ASes need to update their PALs during the month. 8.4% of the ASes need only one PA change in the month, 4% need 2 to 4 PA changes, 1.9% need 5 to 10 PA changes. However, a small number of ASes need more than 100 changes, and AS 701 (UUNET) and its two asserting peers need around $5\,000$ changes. While 5000 prefix assertion updates in a month require significant effort, we suggest that it is feasible for a large organization like UUNET (in this case).

6.5. Discussion

The timeliness of PAL updates is important to ensure service availability. PALs need to be updated and distributed in a timely manner so that prefix ownerships can be verified using currently correct information. To ensure that an asserting peer of a given AS updates its PALs for that AS in a timely manner, a service agreement between them would likely be required, e.g., an extension to their existing agreements. Since there is usually some time delay window before newly delegated prefixes are actually used on the Internet, an asserting peer should be required to update its PAL to include newly delegated prefixes of the asserted peer within that delay window. Updates of prefix removals can be done with lower priority since they would appear to have only relatively small security implications. PALs along with other certificates (e.g., ASNumCerts, SpeakerCerts, and corresponding Certificate Revocation Lists) can be distributed with BGP update messages in newly defined path attributes [25]; thus, they can be distributed as fast as announcements of prefixes and are accessible without any dependence on BGP routes. Those certificates might also be stored in centralized directories [25]. However, a "pull" model might make it challenging to decide how often centralized directories should be checked.

To the best of our knowledge, there is no similar study of projecting the number certificate updates per AS by S-BGP and soBGP. We are currently conducting such study for soBGP and will compare psBGP with soBGP on this aspect.

7. Related Work

Significant research has been published on securing routing protocols. Perlman [34] was among the first to recognize and study the problem of securing routing infrastructures. Bellovin [5] discussed security vulnerabilities of Internet routing protocols as early as 1989. More recently, Bellovin and Gansner [6] discussed potential link-cutting attacks against internet routing. Kumar [27] proposed the use of digital signatures and sequence numbers for protecting the integrity and freshness of routing updates. Smith et al. [38] proposed the use of digital signatures, sequence numbers, and a loop-free path finding algorithm for securing distance vector routing protocols including BGP. Thorough analysis of BGP vulnerabilities and protections was performed by Murphy [30, 31].

The most concrete security proposal to date for addressing BGP vulnerabilities is S-BGP [23, 24, 37], which proposes the use of centralized PKIs for authenticating AS numbers and IP prefix ownership. The S-BGP PKIs are rooted at RIRs, and parallel to the existing system of AS number assignment and IP address allocation. AS_PATH is protected using nested digital signatures, and the integrity of an AS_PATH is guaranteed.

soBGP [41] proposes the use of a web-of-trust model for AS public key authentication, and a centralized hierarchical model for IP prefix ownership verification. AS_PATH is verified for plausibility by checking against an AS topology graph. Each AS issues certificates listing all peering ASes. A global AS graph can be constructed from those certificates. Thus, the existence of an AS_PATH can be verified.

Goodell et al. [15] proposed a protocol, namely Interdomain Routing Validator (IRV), for improving the security and accuracy of BGP. Each AS builds an IRV server which is authoritative of the inter-domain routing information of that AS. An IRV can query another IRV to verify BGP update messages received by its hosting AS. Improper prefix origination and AS_PATH might be detected by uncovering the inconsistency among responses from other IRVs. One advantage of IRV is that it supports incremental deployment since it does not re-

[5] Here an asserting peer of an AS s_i is selected from those peers to which s_i exports its prefixes. We expect such a peer would have the knowledge of s_i's prefix ownership.

quire changes to the existing routing infrastructure.

Kruegel et al. [26] propose a model of AS topology augmented with physical Internet connectivity to detect and stop anomalous route announcements. Their approach passively monitors BGP control traffic, and does not require modification to the existing routing infrastructure. Therefore, it would appear to be easy to deploy.

In a rigorous study of prefix origination authentication, Aiello et al. [2] formalize the IP prefix delegation system, present a proof system, and propose efficient constructions for authenticating prefix origination. Real routing information is analyzed for restoring the IP delegation relationship over the Internet. They discover that the current prefix delegation on the Internet is relatively static and dense, however they also note that it is extremely difficult, if not impossible, to determine this delegation structure.

Listen and Whisper [39] are proposed mechanisms for protecting the BGP data plane and control plane respectively; they are best used together. The first approach (Listen) detects invalid data forwarding by detecting "incomplete" (as defined in [39]) TCP connections. Whisper uncovers invalid routing announcements by detecting inconsistency among *path signatures* of multiple update messages, originating from a common AS but traversing different paths.

Hu et al. [20] propose a Secure Path Vector (SPV) protocol for securing BGP. SPV makes use of efficient cryptographic primitives, e.g., authentication trees, one-way hash chains for protecting AS_PATH. It is shown that SPV is more efficient than S-BGP.

8. Concluding Remarks

Different approaches have been taken by S-BGP and soBGP for addressing security in BGP. In essence, psBGP combines their best features, while differing fundamentally in the approach taken to verify IP prefix ownership. As no centralized infrastructure for tracing changes in IP prefix ownership currently exists, and it would appear to be quite difficult to build such an infrastructure, we suggest that the decentralized approach taken by psBGP provides a more feasible means of increasing confidence in correct prefix origination. We also suggest that the certificate structure and trust model in psBGP has practical advantages. We hope that our comparison of S-BGP, soBGP and psBGP will help focus discussion of securing BGP on the technical merits of the various proposals. We also hope this paper will serve to stimulate discussion in the Internet community about alternate design choices and trust models for securing BGP.

Acknowledgements

We thank Steve Bellovin and anonymous reviewers for their constructive comments which significantly improve the quality of this paper. Specifically, we thank Steve Bellovin for pointing out the collusion problem of multi-AS organizations and for motivating the proposal as described in the last paragraph of §3.4.1. The first author is supported in part by Alcatel Canada, MITACS (Mathematics of Information Technology and Complex Systems), and the NCIT (National Capital Institute of Telecommunications). The second author is supported in part by MITACS and NSERC (Natural Sciences and Engineering Research Council of Canada). The third author is Canada Research Chair in Network and Software Security, and is supported in part by NCIT, MITACS, an NSERC Discovery Grant, and the Canada Research Chairs Program.

References

[1] C. Adams and S. Lloyd. Understanding Public-Key Infrastructure, 2^{nd} edition. Addison Wesley Professional, 2003.

[2] W. Aiello, J. Ioannidis, and P. McDaniel. Origin Authentication in Interdomain Routing. In *Proc. of the 10th ACM Conferences on Computer and Communication Security (CCS'03)*, Washington, D.C., USA. October 2003.

[3] R. Atkinson and S. Floyd. IAB Concerns & Recommendations Regarding Internet Research & Evolution. RFC 3869, August 2004.

[4] A. Barbir, S. Murphy, and Y. Yang. Generic Threats to Routing Protocols. Internet Draft, April 13, 2004.

[5] S.M. Bellovin. Security Problems in the TCP/IP Protocol Suite. *ACM Computer Communications Review*, 19(2): 32-48, April 1989.

[6] S.M. Bellovin and E.R. Gansner. Using Link Cuts to Attack Internet Routing. May 2003. http://www.research.att.com/ smb/papers/

[7] M. Blaze, J. Feigenbaum, J. Lacy. Decentralized Trust Management. In *Proceedings of IEEE Conference on Security and Privacy*, Oakland, USA. May 1996.

[8] D. Boneh, X. Boyen, and H. Shacham. Short Group Signatures. In *Proceedings of Crypto 2004*, LNCS vol 3152, pp. 41-55. Santa Barbara, USA. August 15-19, 2004.

[9] M. Burrows, M. Abadi, and R. Needham. A Logic of Authentication. *Research Report 39*, Digital Systems Research Center, Feburary 1989.

[10] V.J. Bono. 7007 Explanation and Apology. http://www.merit.edu/mail.archives/nanog/1997-04/msg00444.html

[11] P. Ferguson and D. Senie. Network Ingress Filtering: Defeating Denial of Service Attacks Which Employ IP Soure Address Spoofing. RFC 2827, May 2000.

[12] K. Gaarder and E. Snekkenes. Applying a Formal Analysis Technique to the CCIT X.509 Strong Two-Way Au-

thentication Protocol. In *Journal of Cryptology*, 3: 81-98, 1991.

[13] L. Gao. On Inferring Autonomous System Relationships in the Internet. In *Proceedings of IEEE Global Internet*, November 2000.

[14] V.D. Gligor, R. Kailar, S. Stubblebine, and L. Gong. Logics for Cryptographic Protocols - Virtues and Limitations. In *Proceedings of the Computer Security Foundations Workshop IV*, pp. 219-226. IEEE Computer Society Press, Los Alamitos, California, USA. 1991.

[15] G. Goodell, W. Aiello, T. Griffin, J. Ioannidis, P. McDaniel, and A. Rubin. Working around BGP: An Incremental Approach to Improving Security and Accuracy in Interdomain Routing. In *Proc. of 2003 Internet Society Symposium on Network and Distributed System Security (NDSS'03)*, San Diego, USA. February 2003.

[16] R. Guida, R. Stahl, T. Bunt, G. Secrest and J. Moorcones. Deploying and Using Public Key Technology: Lessons Learned in Real Life. *IEEE Security and Privacy*, July/August 2004. pp. 67-71.

[17] C. Hedrick. Routing Information Protocol. RFC 1058. June 1988.

[18] A. Heffernan. Protecting of BGP Sessions via the TCP MD5 signature option. RFC 2385 (Std Track), August 1998.

[19] Y.C. Hu, A. Perrig, and D.B. Johnson. Efficient Security Mechanisms for Routing Protocols. In *Proc. of NDSS'03*, San Diego, USA. Feb 2003.

[20] Y.C. Hu, A. Perrig, and M. Sirbu. SPV: Secure Path Vector Routing for Securing BGP. In *Proc. of SIGCOMM'04*, Portland, Oregon, USA. Aug.30 - Sep.3, 2004.

[21] S. Kent and R. Atkinson. Security Architecture for the Internet Protocol. RFC 2401 (Std Track), November 1998.

[22] S. Kent and R. Atkinson. IP Encapsulating Security Payload (ESP). RFC 2406 (Std Track), November 1998.

[23] S. Kent and C. Lynn, J. Mikkelson, and K. Seo. Secure Border Gateway Protocol (Secure-BGP) - Real World Performance and Deployment Issues. In *Proc. of 2000 Internet Society Symposium on Network and Distributed System Security (NDSS'00)*, San Diego, USA. February 2000.

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

[25] S. Kent. Secure Border Gateway Protocol: A Status Update. In *Proceedings of the* 7^{th} *IFIP TC-6 TC-11 Conference on Communications and Multimedia Security*, Italy, October 2-3, 2003.

[26] C. Kruegel, D. Mutz, W. Robertson, and F. Valeur. Topology-based Detection of Anomalous BGP Messages. In *Proceedings of the 6th Symposium on Recent Advances in Intrusion Detection (RAID'03)*, September 2003.

[27] B. Kumar. Integration of Security in Network Routing Protocols. In ACM SIGSAC Review, 11(2): 18-25, Spring 1993.

[28] U. Maurer. Modelling a Public-Key Infrastructure. In *Proc. of the* 4^{th} *European Symposium on Network and Distributed System Security (ESORICS'96)*, pp. 324-350, 1996.

[29] D. Meyer. The RouteViews Project. August 2004. http://www.routeviews.org/

[30] S. Murphy. Border Gateway Protocol Security Analysis. IETF Internet Draft, draft-murphy-bgp-vuln-00.txt. November 2001.

[31] S. Murphy. BGP Security Protection. IETF Internet Draft, draft-murphy-bgp-protect-02.txt. Feburary 2002.

[32] D.M. Nicol, S.W. Smith, and M.Y. Zhao. Evaluation of efficient security for BGP route announcements using parallel simulation. *Simulation Pratice and Theory Journal*, special issue on Modeling and Simulation of Distributed Systems and Networks. June 2004.

[33] University of Oregon - Looking Glass. http://antc.uoregon.edu/route-views/

[34] R. Perlman. *Network Layer Protocols with Byzantine Robustness*. PhD thesis, Massachusetts Institute of Technology, August 1988.

[35] Y. Rekhter and T. Li. A Border Gateway Protocol 4 (BGP-4), RFC 1771, March 1995.

[36] M. Reiter and S. Stubblebine. Toward Acceptable Metrics of Authentication. In *IEEE Symposium on Security and Privacy*, pp. 10-20, 1997.

[37] K. Seo, C. Lynn, and S. Kent. Public-Key Infrastructure for the Secure Border Gateway Protocol (S-BGP). *IEEE DARPA Information Survivability Conference and Exposition II*, 2001.

[38] B.R. Smith and J.J. Garcia-Luna-Aceves. Securing the Border Gateway Routing Protocol. In *Proceedings of Global Internet 1996.* London, UK. November 1996.

[39] L. Subramanian, V. Roth, I. Stoica, S. Shenker, and R. Katz. Listen and Whisper: Security Mechanisms for BGP. In *Proc. of the First Symposium on Networked Systems Design and Implementation (NSDI'04)*, San Francisco, CA, USA. March 2004.

[40] R.White, D. McPherson, and S. Sangli. *Practical BGP*. Addison-Wesley. June 2004.

[41] R. White. Securing BGP Through Secure Origin BGP (soBGP). In *The Internet Protocol Journal*, 6(3): 15-22, September 2003.

[42] P. Zimmermann. *The Official PGP User's Guide* (second printing). Cambridge, MA: MIT Press, 1995.