INTERNET-DRAFT
Intended Status: Proposed Standard
Expires: January 3, 2012

Yang Xiang
Tsinghua Univ.
Zhiliang Wang
Tsinghua Univ.
Jianping Wu
Tsinghua Univ.
Xingang Shi
Tsinghua Univ.
Xia Yin
Tsinghua Univ.
July 2, 2011

Efficient Secure BGP AS Path using FS-BGP
draft-xiang-sidr-fsbgp-00.txt

Abstract

This draft proposes Fast Secure BGP (FS-BGP), an efficient mechanism for securing AS paths and preventing prefix hijacking by signing critical AS path segments (i.e., adjacent AS triples). FS-BGP can achieve similar level of security as S-BGP, but with much higher efficiency.

Status of this Memo

Table of Contents

1. Introduction

In order to improve the security of BGP, several extensions have been proposed, which fall into two categories: anomaly detection and cryptographic based authentication. However, anomaly detection approaches [Whisper] [PGBGP] can not guarantee security and correctness. Cryptographic approaches, which are being pursued by the SIDR WG, use the Public Key Infrastructure (PKI) to authenticate routing announcements. There are a bunch of solutions including S-BGP [S-BGP] [I-D.lepinski-bgpsec-protocol] and many others. However, S-BGP may consume significant resources of computation and storage. The other solutions either compromise in the security [IRV] [I-D. ng-sobgp-bgp-extensions] [psBGP] [SPV], or bring in more complexity on certification distribution [SA].

Towards these unsolved issues, we propose an efficient approach, FS-BGP (Fast Secure BGP), to secure AS path. Through signing critical AS path segments (i.e., adjacent AS triples), FS-BGP can achieve similar level of security as S-BGP, but with much higher efficiency. Analysis, evaluations, and more discussions can be found in our recent technical report [TR-FSBGP].

2. Terminology

(i):	AS i
<n, ..., 0>:	AS path from AS n to the origin AS 0
<n, ..., 0>f:	AS path of prefix f
<i+1, i, i-1>:	critical AS path segment, adjacent AS triple in a path
<1, 0, f>:	origin critical AS path segment in a path of prefix f
{msg}i:	signature on msg generated by AS i

3. Background

In BGP, UPDATE messages can not be validated, so neither the origin AS nor the AS path is guaranteed to be correct. Secure BGP (S-BGP) [SBGP] is the dominant solution to this problem, and it uses a PKI to help authenticating involved parties and messages. Specifically, S-BGP uses Route Attestations (RAs) for path authentication.

As shown in Figure 1, a RA is all signatures signed by ASes along the path to authenticate the existence and position of ASes in the path. We define {msg}i as the signature on msg generated with AS i's private key. In Figure 1, each AS i equivalently signs the corresponding extended AS path <i+1, i, ..., 0> and the prefix f. The inclusion of the recipient AS i+1 in each signature is necessary to prevent cut-and-paste attack.

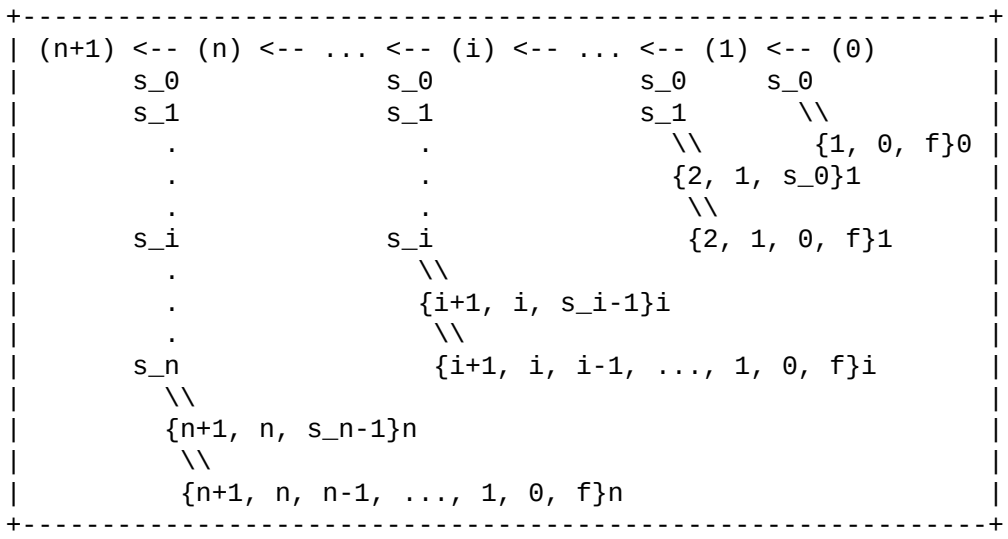

Figure 1. RAs in S-BGP.

The main concern about deploying S-BGP in practice is the huge computational cost for signing and verifying signatures. The dominating barrier for adopting S-BGP is the overhead of processing RAs, that is to authenticate paths. Toward this direction, there are a bunch of solutions for reducing the overhead of path authentication.

soBGP [I-D.ng-sobgp-bgp-extensions] maintains all authenticated AS edges in a database, but faces the problem of forged paths. IRV [IRV] builds an authentication server in each AS, but brings the problem of maintaining and inter-connecting these servers, and introduces query latencies. SPV [SPV] accelerates the signing process by pre-generated one-time signatures based on a single root value, but involves a significant amount of state information, and its security can only be guaranteed probabilistically. Signature Amortization (S-A) [SA] uses one bit vector for each neighbor of an AS to indicate the allowed recipients of a route, such that only one signing is needed for multiple recipients. However, each AS will need to pre-establish a neighbor list corresponding to the bit vector, and to distribute it to all other ASes.

As we can see, existing methods usually compromise security, and most of them only improve the performance of signing. However, verification happens more frequently than signing, since one signature often needs to be verified at multiple places.

According to the above analysis, it is important to design an efficient method to secure AS paths. Our solution, FS-BGP, builds on the assumption that a PKI is ready for use, and focuses on AS path

authentication.

4. Secure Feasible AS Paths

S-BGP can not prevent replay of outdated routes. It can only use expiration-date to roughly control the window of exposure to replay attack. As a result, though it only signs currently announcing path, it actually authenticates all announced feasible paths. Under a stable AS-level topology, we call a path feasible when the path satisfies the import and export policies of all ASes along the path.

Since failures often occur in the global routing system, many feasible paths can be easily announced and become authenticated. Thus, if a protocol can guarantee that all authenticated paths are feasible path, then it can achieve similar level of security as S-BGP. So we wonder that is it possible to efficiently secure feasible paths but not blindly sign every currently announcing path.

BGP is a policy-based routing protocol. An AS only exports a route to a neighbor if it is willing to forward traffic to the corresponding prefix from that neighbor. Although complex policies (i.e., route filters [RFC2622]) exist, AS usually does not differentiate between prefixes or nonadjacent ASes. For example, in Figure 2, when AS n decides whether routes learned from AS n-1 can be exported to AS n+1, it only considers its relation with the two neighbors, but does not consider other ASes along the path (<n-2, ..., 1, 0>). We call this the Neighbor Based Importing and Exporting (NBIE).

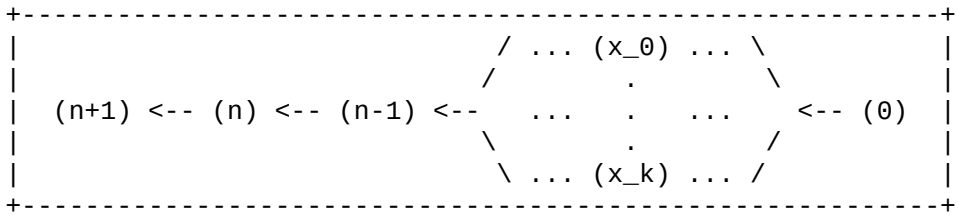

Figure 2. In S-BGP, AS n signs k paths which share a mutual AS path segment <n+1, n, n-1>.

NBIE abstracts the basic functionality of BGP. According to our measurement results in whois database, only a small portion of routing polices (route filters) violate the NBIE assumption. Nevertheless, the purpose of route filters is to protect the routing system against distribution of inaccurate routing information [RFC2622]. In other words, the use of route filters is mainly due to security considerations rather than policy requirements. We believe that under a security environment (i.e., FS-BGP or S-BGP), these filters are not needed any more. In deed, our schema can flexibly

support complicated routing polices [TR-FSBGP].

5. FS-BGP: Fast Secure BGP

5.1. Signing Critical AS Path Segments

Following our key observation above, we propose Fast Secure BGP (FS-BGP) to grantee the authentication of feasible paths. Given a feasible path p=<n+1, n, ..., 0>, we define its set of critical path segments as c_i, 0<i<=n, where

$$c_i = \begin{cases} <1,0,f> & \text{, for } i=0 \\ <i+1,i,i-1> & \text{, for } 0<i<=n \end{cases}$$

We call AS i the owner of c_i. Particularly, c_0 is called the originating critical path segment owned by AS 0. A critical path segment <i+1, i, i-1> actually describes an routing export policy of its owner AS i, and implies that AS i can export all routes imported from AS i-1 to AS i+1.

More specifically, FS-BGP uses Critical Segment Attestations (CSA) to authenticate paths. A CSA is simply the signature of the critical path segment signed by its owner. In a path p=<n+1, n, ..., 0>, the CSA s_i signed by AS i is defined as:

$$s_i = \begin{cases} \{1,0,f\}0 & \text{, for } i=0 \\ \{i+1,i,i-1\}i & \text{, for } 0<i<=n \end{cases}$$

The inclusion of the prefixes f in s_0 is necessary, because AS 0 might be multi-homing and only announces part of its prefixes to AS 1. Figure 3 and Figure 1 compare the signatures in FS-BGP and S-BGP. Obviously, the number of distinct critical path segments is far less than the number of distinct paths. As a result, the number of signing and verification operations in FS-BGP can be greatly reduced, after using a small cache. In Figure 2, AS n needs to sign each of the k paths individually in S-BGP. However, in FS-BGP, all the k different paths can reuse one signature of the common critical segment <n+1, n, n-1>.

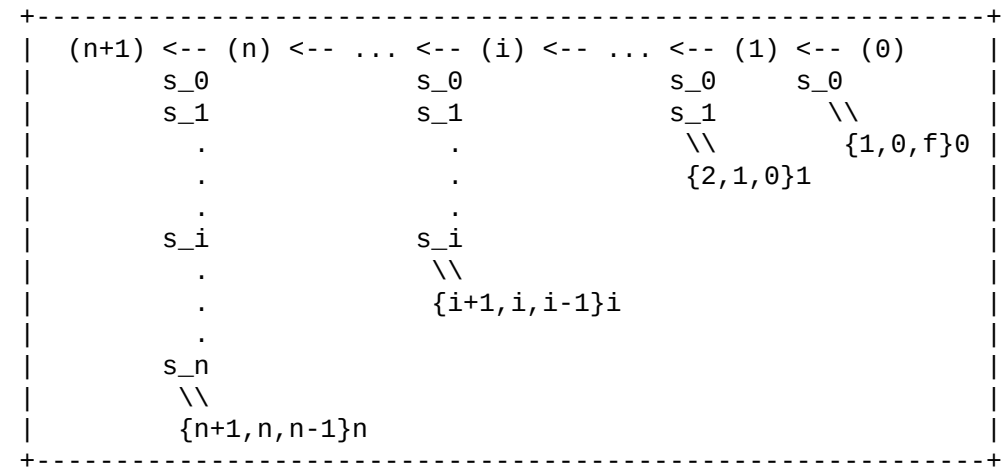

Figure 3. CSAs in FS-BGP.

We argue that, under the NBIE rule, if every AS along a path signs its critical path segment, then the path can be authenticated as a feasible path [TR-FSBGP]. However, since not all feasible paths are actually announced, it is possible to forge a path if the security mechanism relies on CSA only, as shown in Section 5.2. We will provide effective solution to this problem.

5.2. Prevent Effective Hijacking in FS-BGP

In FS-BGP, an AS using FS-BGP can forge paths that are not actually announced by others, but avoids CSA based detection. Forged paths can be constructed by concatenating critical segments, and used for prefix hijacking.

Although forging paths in FS-BGP is possible, there are still some restrictions on how paths can be forged. First, a path can only be forged by combining non-forged paths which share mutual segments. Second, some part of a forged path must be treated as sub-optimal or suppressed by some AS along the path. Third, forged paths are still feasible, and can only be used for prefixes associated with the originating critical segment in the path. Last, forged paths can not be very short [TR-FSBGP].

Although there are limitations on forged paths, prefix hijacking is still possible. In this section, we discuss solutions to prevent prefix hijacking. We only concern effective hijacking, in a sense that, the recipient of a forged route indeed changes its forwarding path.

Firstly, we divide all feasible paths into three categories:

Optimal path: the best path that passes all the decision steps in BGP.

Sub-optimal path: paths with the same Local Preference as the optimal path, but not chosen as the best one.

Suppressed path: paths with lower Local Preferences than the optimal and sub-optimal paths. For example, paths that are more expensive (i.e., through a provider), are often suppressed by a low preference.

We argue that, if a forged path is no shorter than the non-forged path BGP should announce, it can not be used for effective hijacking [TR-FSBGP]. Under a stable AS-level topology, a router will use its optimal path for every prefix. If BGP is purely a shortest path routing protocol (optimal path is always the shortest one), manipulator can not effectively hijack any prefix by forging paths. However, policy routing makes hijacking possible.

We know only suppressed path can be shorter than the optimal path (since a sub-optimal path has the same local preference as the optimal path, its length can not be shorter). Thus, if there is a mechanism to guarantee that all suppressed paths are no shorter than their corresponding optimal paths, manipulator can no longer effectively hijack a prefix either. This idea can be implemented by using AS Path Pre-pending (ASPP).

We call such a mechanism Suppressed Path Padding (SPP), and Figure 4 depicts the pseudo code for deciding how many times an AS i should pad itself in a path. If a path is imported from a neighbor AS i-1 with the highest local preference, AS i only appears once (line 1 and 2). Otherwise, the number of occurrences k_i must be large enough such that no suppressed path can be shorter than the corresponding optimal path. Given a path p, denote the optimal path to the same prefix as p by opt(p), then k_i is set as the largest Path Length difference between any suppressed path p imported from this neighbor and the corresponding opt(p) (line 4 to 7).

```
+-------------------------------------------------------------+
| Algorithm: Suppressed Path Padding                          |
| INPUT:  local AS i, neighbor AS i-1                         |
| OUTPUT: k_i: number of times that AS i needs to be padded   |
|         in the paths import from AS i-1                     |
| 1:  IF AS i-1 has the highest local preference THEN         |
| 2:      RETURN 1                                            |
| 3:  k_i <- 1                                                |
| 4:  FOR ALL path p imported from AS i-1 DO                  |
| 5:      opt(p) <- the optimal path corresponding to p       |
| 6:      IF length(p) - length(opt(p)) > k_i THEN            |
| 7:          k_i <- length(p) - length(opt(p))               |
| 8:  RETURN k_i                                              |
+-------------------------------------------------------------+
```

Figure 4. SPP (Suppressed Path Padding).

It is worth noting that, SPP is quite general. When necessary, it can and also should be used even in S-BGP. Consider the case when the optimal route fails. At this time, S-BGP will announce a previously sub-optimal or suppressed path temporarily, and this path can be used later by the manipulator to launch an effective attack, if it is short enough. S-BGP can not prevent this attack, while our SPP works effectively.

6. Security Considerations

The entire document is about security consideration. More theoretical analysis and experiment results can be found in our technical report [TR-FSBGP].

7. IANA Considerations

This document requires no IANA actions.

8. Conclusions

This draft proposes Fast Secure BGP (FS-BGP), an efficient mechanism for securing feasible AS paths and preventing prefix hijacking by signing critical AS path segments. We believe that FS-BGP can achieve similar level of security as S-BGP. Our experiment results show that, FS-BGP has a much higher efficiency.

9 References

9.1 Normative References

[RFC2622] Alaettinoglu, C., Villamizar, C., Gerich, E., Kessens, D., Meyer, D., Bates, T., Karrenberg, D., and M. Terpstra, "Routing Policy Specification Language (RPSL)", RFC 2622, June 1999.

[RFC4271] Rekhter, Y., Ed., Li, T., Ed., and S. Hares, Ed., "A Border Gateway Protocol 4 (BGP-4)", RFC 4271, January 2006.

[S-BGP] S. Kent, C. Lynn, J. Mikkelson, and K. Seo, "Secure Border Gateway Protocol (S-BGP)", IEEE Journal on Selected Areas in Communications, 18:103-116, 2000.

[I-D.lepinski-bgpsec-protocol] M. Lepinski, "BGPSEC Protocol Specification", draft-lepinski-bgpsec-protocol, work-in-progress, 2011.

[I-D.ng-sobgp-bgp-extensions] J. Ng, "Extensions to BGP to Support Secure Origin BGP (soBGP)", draft-ng-sobgp-bgp-extensions, 2004.

[IRV] 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, 2003.

[psBGP] P. C. van Oorschot, T. Wan, and E. Kranakis, "On interdomain routing security and pretty secure BGP (psBGP)", ACM Trans. Inf. Syst. Secur., 10(3), 2007.

[SPV] Y.-C. Hu, A. Perrig, and M. A. Sirbu, "SPV: secure path vector routing for securing BGP", In SIGCOMM, pages 179-192, 2004.

[SA] D. M. Nicol, S. W. Smith, and M. Zhao, "Evaluation of efficient security for BGP route announcements using parallel simulation", Simulation Modeling Practice and Theory, 12(3-4):187-216, 2004.

[TR-FSBGP] Yang Xiang, Zhiliang Wang, Xia Yin, Xingang Shi, and Jianping Wu, "FS-BGP: An Efficient Approach to Securing AS Paths", Tsinghua University, Technical Report, THUTR-2011-FSBGP, 2011.

9.2 Informative References

[Whisper] L. Subramanian, V. Roth, I. Stoica, S. Shenker, and R. H. Katz, "Listen and Whisper: Security Mechanisms for BGP", In NSDI, pages 127-140, 2004.

[PGBGP] J. Karlin, S. Forrest, and J. Rexford, "Pretty Good BGP: Improving BGP by Cautiously Adopting Routes", In ICNP, pages 290-299, 2006.

Authors' Addresses

Yang Xiang
Tsinghua University, Beijing, 100084 P.R. China

Email: xiangy08@csnet1.cs.tsinghua.edu.cn

Zhiliang Wang
Tsinghua University, Beijing, 100084 P.R. China

Email: wzl@csnet1.cs.tsinghua.edu.cn

Jianping Wu
Tsinghua University, Beijing, 100084 P.R. China

Email: jianping@csnet1.cs.tsinghua.edu.cn

Xingang Shi
Tsinghua University, Beijing, 100084 P.R. China

Email: shxg@csnet1.cs.tsinghua.edu.cn

Xia Yin
Tsinghua University, Beijing, 100084 P.R. China

Email: yxia@csnet1.cs.tsinghua.edu.cn