Network Working Group
Internet-Draft
Intended status: Standards Track
Expires: September 8, 2011

M. Lepinski, Ed.
BBN

March 7, 2011

BGPSEC Protocol Specification
draft-lepinski-bgpsec-protocol-00.txt

Abstract

This document describes BGPSEC, a mechanism for providing path security for BGP route advertisements. BGPSEC is implemented via a new optional non-transitive BGP path attribute.

Requirements Language

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in RFC 2119 [4].

Status of this Memo

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at http://datatracker.ietf.org/drafts/current/.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on September 8, 2011.

Table of Contents

1. Introduction

This document describes BGPSEC, a mechanism for providing path security for BGP route advertisements. That is, a BGP speaker who receives a valid BGPSEC update has cryptographic assurance that the advertised route has the following two properties:

1. The route was originated by an AS that has been explicitly authorized by the holder of the IP address prefix to originate route advertisements for that prefix.

2. Every AS listed in the AS_Path attribute of the update explicitly authorized the advertisement of the route to the subsequent AS in the AS_Path.

This document specifies a new optional (non-transitive) BGP path attribute, BGPSEC_Path_Signatures. It also describes how a BGPSEC-compliant BGP speaker (referred to hereafter as a BGPSEC speaker) can generate, propagate, and validate BGP update messages containing this attribute to obtain the above assurances.

BGPSEC relies on the Resource Public Key Infrastructure (RPKI) certificates that attest to the allocation of AS number and IP address resources. (For more information on the RPKI, see [7] and the documents referenced therein.) Any BGPSEC speaker who wishes to send BGP update messages to external peers (eBGP) containing the BGPSEC_Path_Signatures must have an RPKI end-entity certificate (as well as the associated private signing key) corresponding to the BGPSEC speaker's AS number. Note, however, that a BGPSEC speaker does not require such a certificate in order to validate update messages containing the BGPSEC_Path_Signatures attribute.

2. BGPSEC Negotiation

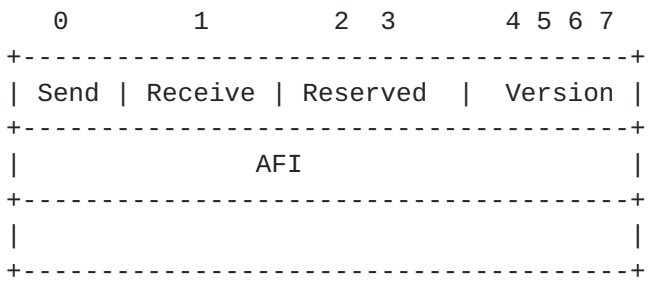
This document defines a new BGP capability [3]that allows a BGP speaker to advertise to its neighbors the ability to send and/or receive BGPSEC update messages (i.e., update messages containing the BGPSEC_Path_Signatures attribute).

This capability has capability code : TBD

The capability length for this capability MUST be set to 3.

The three octets of the capability value are specified as follows.

Capability Value:



The high order bit (bit 0) of the first octet is set to 1 to indicate that the sender is able to send BGPSEC update messages, and is set to zero otherwise. The next highest order bit (bit 1) of this octet is set to 1 to indicate that the sender is able to receive BGPSEC update messages, and is set to zero otherwise. The next two bits of the capability value (bits 2 and 3) are reserved for future use.

The four low order bits (4, 5, 6 and 7) of the first octet indicate the version of BGPSEC for which the BGP speaker is advertising support. This document defines only BGPSEC version 0 (all four bits set to zero). Other versions of BGPSEC may be defined in future documents. A BGPSEC speaker MAY advertise support for multiple versions of BGPSEC by including multiple versions of the BGPSEC capability in its BGP OPEN message.

If there does not exist at least one version of BGPSEC that is supported by both peers in a BGP session, then the use of BGPSEC has not been negotiated. (That is, in such a case, messages containing the BGPSEC_Path_Signatures MUST not be sent.)

If version 0 is the only version of BGPSEC for which both peers (in a BGP session) advertise support, then the use of BGPSEC has been negotiated and the BGPSEC peers MUST adhere to the specification of BGPSEC provided in this document. (If there are multiple versions of BGPSEC which are supported by both peer, then the behavior of those peers is outside the scope of this document.)

The second two octets contain the 16-bit Address Family Identifier (AFI) which indicates the address family for which the BGPSEC speaker is advertising support for BGPSEC. This document only specifies BGPSEC for use with two address families, IPv4 and IPv6. BGPSEC for use with other address families may be specified in future documents. Note that if the BGPSEC speaker wishes to use BGPSEC with two different address families (i.e., IPv4 and IPv6) over the same BGP session, then the speaker must include two instances of this capability (one for each address family) in the BGP OPEN message. Also note that a BGPSEC speaker SHOULD NOT advertise the capability

of BGPSEC support for IPv6 unless it has also advertised support for IPv6 [2].

By indicating support for receiving BGPSEC update messages, a BGP speaker is, in particular, indicating that the following are true:

- The BGP speaker understands the BGPSEC_Path_Signatures attribute (see Section 3).
- The BGP speaker supports 4-byte AS numbers (see RFC 4893).

Note that BGPSEC update messages can be quite large, therefore any BGPSEC speaker announcing the capability to receive BGPSEC messages SHOULD also announce support for the capability to receive BGP extended messages [5].

A BGP speaker MUST NOT send an update message containing the BGPSEC_Path_Signatures attribute within a given BGP session unless both of the following are true:

- The BGP speaker indicated support for sending BGPSEC update messages in its open message.
- The peer of the BGP speaker indicated support for receiving BGPSEC update messages in its open message.

3. The BGPSEC_Path_Signatures Attribute

The BGPSEC_Path_Signatures attribute is a new optional (non-transitive) BGP path attribute.

This document registers a new attribute type code for this attribute : TBD

The BGPSEC_Path_Signatures attribute has the following structure:

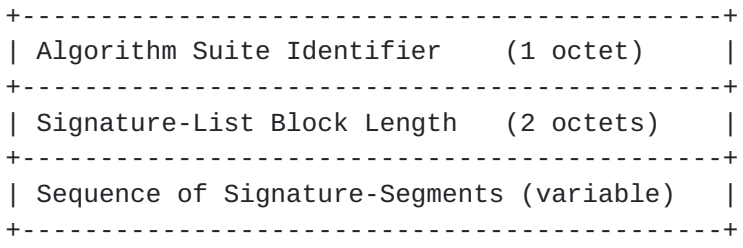
BGPSEC_Path_Signatures Attribute

Expire Time contains a binary representation of a time as an unsigned integer number of (non-leap) seconds that have elapsed since midnight

UTC January 1, 1970. The Expire Time indicates the latest point in time that the route advertised in the update message can possibly be considered valid (see Section 5 for details on validity of BGPSEC update messages).

The BGPSEC_Path_Signatures attribute will contain one or two Signature-List Blocks, each of which corresponds to a different algorithm suite. Each of the Signature-List Blocks will contain a signature segment for each AS in the AS Path attribute. In the most common case, the BGPSEC_Path_Signatures attribute will contain only a single Signature-List Block. However, in order to enable a transition from an old algorithm suite to a new algorithm suite, it will be necessary to include two Signature-List Blocks (one for the old algorithm suite and one for the new algorithm suite) during the transition period.

Signature-List Block



An algorithm suite consists of a digest algorithm and a signature algorithm. This version of BGPSEC only supports signature algorithms that produce a signatures of fixed length. This specification creates an IANA registry of one-octet BGPSEC algorithm suite identifiers. Additionally, this document registers a single algorithm suite which uses the digest algorithm SHA-256 and the signature algorithm RSA with 2048-bit keys [1]. The signatures produced by this algorithm suite have a length of 256 octets. Future registrations of algorithm suites for BGPSEC must specify the length of signatures produced by the algorithm suite.

BGPSEC Algorithm Suites

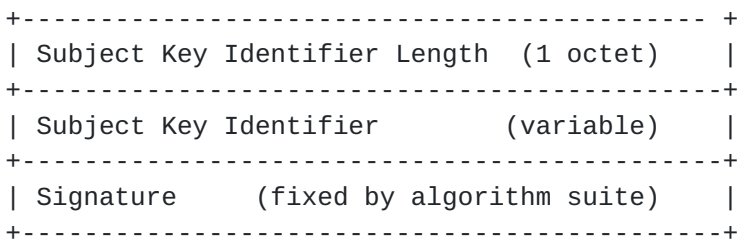
Algorithm Suite Identifier	Digest Algorithm	Signature Algorithm	Specification Pointer
TBA	SHA-256	RSA 2048	RFC 3447

The Signature-List Block Length is the total number of octets in all Signature-Segments (i.e., the total size of the variable-length

portion of the Signature-List block.)

A Signature-Segment has the following structure:

Signature Segments



The Subject Key Identifier Length contains the size (in octets) of the value in the Subject Key Identifier field of the Signature-Segment. The Subject Key Identifier contains the value in the Subject Key Identifier extension of the RPKI end-entity certificate that is used to verify the signature (see Section 5 for details on validity of BGPSEC update messages).

The Signature contains a digital signature that protects the NLRI, the AS_Path and the BGPSEC_Path_Signatures attribute (see Sections 4 and 5 for details on generating and verifying this signature, respectively). The length of the Signature field is a function of the algorithm suite for a given Signature-List Block. The specification for each BGPSEC algorithm suite must provide the length of signatures constructed using the given algorithm suite.

4. Generating a BGPSEC Update

Sections 4.1 and 4.2 cover two cases in which a BGPSEC speaker may generate an update message containing the BGPSEC_Path_Signatures attribute. The first case is that in which the BGPSEC speaker originates a new route advertisement (Section 4.1). That is, the BGPSEC speaker is constructing an update message in which the only AS to appear in the AS Path attribute is the speaker's own AS (normally appears once but may appear multiple times if AS prepending is applied). The second case is that in which the BGPSEC speaker receives a route advertisement from a peer and then decides to propagate the route advertisement to an external (eBGP) peer (Section 4.2). That is, the BGPSEC speaker has received a BGPSEC update message and is constructing a new update message for the same NLRI in which the AS Path attribute will contain AS number(s) other than the speaker's own AS.

In the remaining case where the BGPSEC speaker is sending the update message to an internal (iBGP) peer, the BGPSEC speaker populates the BGPSEC_Path_Signatures attribute by copying the BGPSEC_Path_Signatures attribute from the received update message. That is, the BGPSEC_Path_Signatures attribute is copied verbatim. Note that in the case that a BGPSEC speaker chooses to forward to an iBGP peer a BGPSEC update message that has not been successfully validated (see Section 5), the BGPSEC_Path_Signatures attribute SHOULD NOT be removed. (See Section 7 for the security ramifications of removing BGPSEC signatures.)

The information protected by the signature on a BGPSEC update message includes the AS number of the peer to whom the update message is being sent. Therefore, if a BGPSEC speaker wishes to send a BGPSEC update to multiple BGP peers, it MUST generate a separate BGPSEC update message for each unique peer AS to which the update message is sent.

A BGPSEC update message MUST advertise a route to only a single NLRI. If a BGPSEC speaker wishes to advertise routes to multiple NLRI, then it MUST generate a separate BGPSEC update message for each NLRI.

Note that in order to create or add a new signature to a Signature-List Block for a given algorithm suite, the BGPSEC speaker must possess a private key suitable for generating signatures for this algorithm suite. Additionally, this private key must correspond to the public key in a valid Resource PKI end-entity certificate whose AS number resource extension includes the BGPSEC speaker's AS number. Note also new signatures are only added to a BGPSEC update message when a BGPSEC speaker is generating an update message to send to an external peer (i.e., when the AS number of the peer is not equal to the BGPSEC speaker's own AS number). Therefore, a BGPSEC speaker who only sends BGPSEC update messages to peers within its own AS, it does not need to possess any private signature keys.

4.1. Originating a New BGPSEC Update

In an update message that originates a new route advertisement (i.e., an update whose AS_Path contains, possibly multiple occurrences of, a single AS number), the BGPSEC speaker creates one Signature-List Block for each algorithm suite that will be used. Typically, a BGPSEC speaker will use only a single algorithm suite. However, to ensure backwards compatibility during a period of transition from a 'current' algorithm suite to a 'new' algorithm suite, it will be necessary to originate update messages containing Signature-List Blocks for both the 'current' and the 'new' algorithm suites (see Section 6.1).

The Resource PKI enables the legitimate holder of IP address prefix(es) to issue a signed object, called a Route Origination Authorization (ROA), that authorizes a given AS to originate routes to a given set of prefixes (see [6]).Note that validation of a BGPSEC update message will fail (i.e., the validation algorithm, specified in Section 5.1, returns 'Not Good') unless there exists a valid ROA authorizing the first AS in the AS PATH attribute to originate routes to the prefix being advertised. Therefore, a BGPSEC speaker SHOULD NOT originate a BGPSEC update advertising a route for a given prefix unless there exists a valid ROA authorizing the BGPSEC speaker's AS to originate routes to this prefix.

The Expire Time field is set to specify a time at which the route advertisement specified in the update message will cease to be valid. Once the Expire Time has been reached, all BGPSEC speakers who have received the advertisement will treat it as invalid. The purpose of this field is to protect the BGPSEC speaker against attacks in which the BGPSEC speaker wishes to withdraw the route, but intermediate (malicious) BGP speakers fail to propagate the withdrawal to their peers.

It is therefore necessary for the originating BGPSEC speaker to issue a new BGPSEC update prior to reaching the Expire Time. It is RECOMMENDED that a BGPSEC speaker originate a new route advertisement for a given NLRI at intervals equal to roughly one-third the validity period of the route advertisement. (Note that it is necessary to add some small amount of random jitter to the interval to avoid synchronization effects.) For instance, if a BGPSEC speaker is originating route advertisements that are valid for one day (i.e., the Expire Time is 24 hours after the generation of the update message), then it is recommended that the BGPSEC speaker re-issue new a new BGPSEC update message for advertising the given prefix roughly once every 8 hours (plus or minus a small random value).

(Editor's Note: The parameter recommendations in the previous paragraph are preliminary and may need to be updated based on further implementation and deployment experience.)

There is a natural trade-off in setting the Expire Time. Setting a later Expire Time increases the amount of time by which a malicious intermediate can delay a future route withdrawal. Similarly, setting a later Expire Time also increases the window of opportunity for malicious replay attacks in which a previous BGPSEC announcement is replayed while suppressing a more recent withdrawal for the same prefix. However, setting a sooner Expire Timed increases the frequency with which the BGPSEC speaker needs to send new announcements for the given prefix.

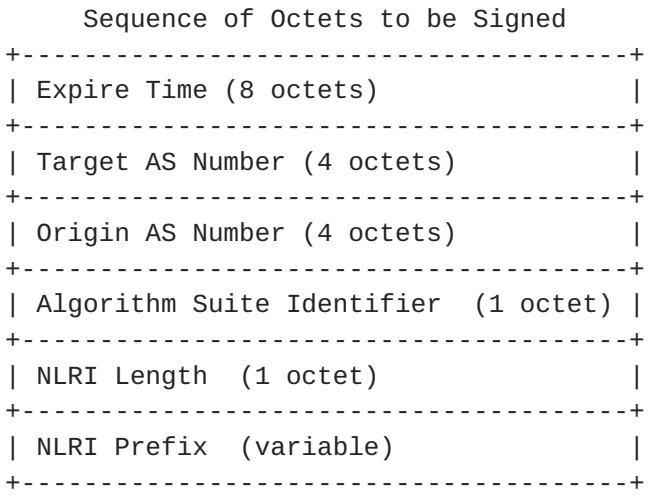
When originating a new route advertisement, each Signature-List Block MUST consist of a single Signature-Segment. The following describes how the BGPSEC speaker populates the fields of the Signature-List Block (see Section 3 for more information on the syntax of Signature-List Blocks).

The Subject Key Identifier field (see Section 3) is populated with the identifier contained in the Subject Key Identifier extension of the RPKI end-entity certificate used by the BGPSEC speaker. This Subject Key Identifier will be used by recipients of the route advertisement to identify the proper certificate to use in verifying the signature.

The Subject Key Identifier Length field is populated with the length (in octets) of the Subject Key Identifier.

The Signature field contains a digital signature that binds the NLRI, AS_Path attribute and BGPSEC_Path_Signatures attribute to the RPKI end-entity certificate used by the BGPSEC speaker. The digital signature is computed as follows:

- Construct a sequence of octets by concatenating the Expire Time, Target AS Number, Origin AS Number, Algorithm Suite Identifier, and NLRI. The Target AS Number is the AS to whom the BGPSEC speaker intends to send the update message. (Note that the Target AS number is the AS number announced by the peer in the OPEN message of the BGP session within which the update is sent.) The Origin AS number prepend to this sequence the Target AS (the AS to whom the BGPSEC speaker intends to send the update message) and the Origin AS Number refers to the AS of the BGPSEC speaker who is originating the route advertisement.



- Apply to this octet sequence the digest algorithm (for the algorithm suite of this Signature-List) to obtain a digest value.

- Apply to this digest value the signature algorithm, (for the algorithm suite of this Signature-List) to obtain the digital signature. Then populate the Signature Field with this digital signature.

4.2. Propagating a Route Advertisement

When a BGPSEC speaker receives a BGPSEC update message containing a BGPSEC_Path_Signatures algorithm (with one or more signatures) from a (internal or external) peer, it may choose to propagate the route advertisement by sending to its (internal or external) peers by creating a new BGPSEC advertisement for the same prefix.

A BGPSEC speaker MUST NOT generate an update message containing the BGPSEC_Path_Signatures attribute unless it has selected, as the best route to the given prefix, a route that it received in an update message containing the BGPSEC_Path_Signatures attribute. In particular, this means that whenever a BGPSEC speaker generates an update message with a BGPSEC_Path_Signatures attribute that it will possess a received update message for the same prefix that also contains a BGPSEC_Path_Signatures attribute.

Additionally, whenever a BGPSEC speaker selects as the best route to a given prefix a route that it received in an update message containing the BGPSEC_Path_Signatures attribute, it is RECOMMENDED that if the BGPSEC speaker chooses to propagate the route that it generate an update message containing the BGPSEC_Path_Signatures attribute. However, a BGPSEC speaker MAY propagate a route advertisement by generating a (non-BGPSEC) update message that does not contain the BGPSEC_Path_Signatures attribute. (See Section 7 for

discussion of the security ramifications of removing BGPSEC signatures.)

If the BGPSEC speaker is producing an update message which contains an AS-SET (e.g., the BGPSEC speaker is performing proxy aggregation), then the BGPSEC speaker MUST not include the BGPSEC_Path_Signatures attribute. In such a case, the BGPSEC speaker must remove any existing BGPSEC_Path_Signatures in the received advertisement(s) for this prefix and produce a standard (non-BGPSEC) update message.

To generate the BGPSEC_Path_Signatures attribute on the outgoing update message, the BGPSEC first copies the Expire Time directly from the received update message to the new update message (that it is constructing). Note that the BGPSEC speaker MUST NOT change the Expire Time as any change to Expire Time will cause the new BGPSEC update message to fail validation (see Section 5).

The BGPSEC speaker next removes from the BGPSEC_Path_Signatures attribute any Signature-List Blocks corresponding to algorithm suites that it does not support. The BGPSEC_Path_Signatures attribute for the new update message SHOULD contain a Signature-List Block for every algorithm suite that is both present in the received update message and which is supported by the BGPSEC speaker.

Note that the validation algorithm (see Section 5.1) deems a BGPSEC update message to be 'Good' if there is at least one supported algorithm suite (and corresponding Signature-List Block) that is deemed 'Good'. This means that a 'Good' BGPSEC update message may contain Signature-List Blocks which are deemed 'Not Good' (e.g., contain signatures that the BGPSEC is unable to verify). Nonetheless, such Signature-List Blocks MUST NOT be removed. (See Section 7 for a discussion of the security ramifications of this design choice.)

For each Signature-List Block corresponding to an algorithm suite that the BGPSEC speaker does support, the BGPSEC speaker then adds a new Signature-Segment to the Signature-List Block. This Signature-Segment is prepended to the list of Signature-Segments (placed in the first position) so that the list of Signature-Segments appears in the same order as the corresponding AS numbers in the AS-Path attribute. The BGPSEC speaker populates the fields of this new signature-segment as follows.

The Subject Key Identifier field in the new segment is populated with the identifier contained in the Subject Key Identifier extension of the RPKI end-entity certificate used by the BGPSEC speaker. This Subject Key Identifier will be used by recipients of the route advertisement to identify the proper certificate to use in verifying

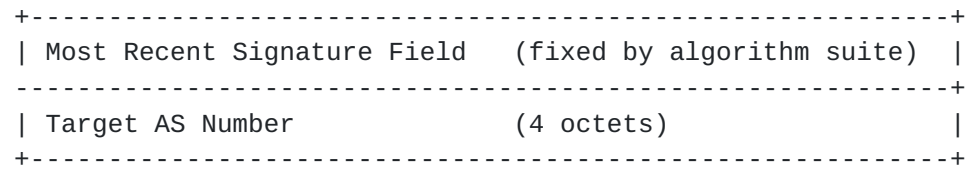
the signature.

The Subject Key Identifier Length field is populated with the length (in octets) of the Subject Key Identifier.

The Signature field in the new segment contains a digital signature that binds the NLRI, AS_Path attribute and BGPSEC_Path_Signatures attribute to the RPKI end-entity certificate used by the BGPSEC speaker. The digital signature is computed as follows:

- Construct a sequence of octets by concatenating the signature field of the most recent Signature-Segment (the one corresponding to AS from whom the BGPSEC speaker's AS received the announcement) with the Target AS (the AS to whom the BGPSEC speaker intends to send the update message). Note that the Target AS number is the AS number announced by the peer in the OPEN message of the BGP session within which the BGPSEC update message is sent.

Sequence of Octets to be Signed



- Apply to this octet sequence the digest algorithm (for the algorithm suite of this Signature-List) to obtain a digest value.

- Apply to this digest value the signature algorithm, (for the algorithm suite of this Signature-List) to obtain the digital signature. Then populate the Signature Field with this digital signature.

5. Validating a BGPSEC Update

Validation of a BGPSEC update messages makes use of data from RPKI certificates and signed Route Origination Authorizations (ROA). In particular, to validate update messages containing the BGPSEC_Path_Signatures attribute, it is necessary that the recipient have access to the following data obtained from valid RPKI certificates and ROAs:

- For each valid RPKI end-entity certificate containing an AS Number extension, the AS Number, Public Key and Subject Key Identifier are required

- For each valid ROA, the AS Number and the list of IP address prefixes

Note that the BGPSEC speaker could perform the validation of RPKI certificates and ROAs on its own and extract the required data, or it could receive the same data from a trusted cache that performs RPKI validation on behalf of (some set of) BGPSEC speakers.

To validate a BGPSEC update message containing the BGPSEC_Path_Signatures attribute, the recipient performs the validation steps specified in Section 5.1. The validation procedure results in one of two states: 'Good' and 'Not Good'.

It is expected that the output of the validation procedure will be used as an input to BGP route selection. However, BGP route selection and thus the handling of the two validation states is a matter of local policy, and shall be handled using existing local policy mechanisms. It is expected that BGP peers will generally prefer routes received via 'Good' BGPSEC update messages over routes received via 'Not Good' BGPSEC update messages as well as routes received via update messages that do not contain the BGPSEC_Path_Signatures attribute. However, BGPSEC specifies no changes to the BGP decision process and leaves to the operator the selection of an appropriate policy mechanism to achieve the operator's desired results within the BGP decision process.

BGPSEC validation need only be performed at eBGP edge. The validation status of a BGP signed/unsigned update MAY be conveyed via iBGP from an ingress edge router to an egress edge router. Local policy in the AS determines the specific means for conveying the validation status through various pre-existing mechanisms such as setting a BGP community, or modifying a metric value such as Local_Pref or MED. As discussed in Section 4, when a BGPSEC speaker chooses to forward a (syntactically correct) BGPSEC update message, it SHOULD be forwarded with its BGPSEC_Path_Signatures attribute intact (regardless of the validation state of the update message). Based entirely on local policy settings, an egress router MAY trust the validation status conveyed by an ingress router or it MAY perform its own validation.

5.1. Validation Algorithm

This section specifies an algorithm for validation of BGPSEC update messages. A conformant implementation MUST include an BGPSEC update validation algorithm that is functionally equivalent to the external behavior of this algorithm.

First, the recipient of a BGPSEC update message performs a check to

ensure that the message is properly formed. Specifically, the recipient performs the following checks:

- Check to ensure that the entire BGPSEC_Path_Signatures attribute is syntactically correct (conforms to the specification in this document).

- Check to ensure that the AS-Path attribute contains no AS-Set segments.

- Check that each Signature-List Block contains one Signature-Segment for each AS in the AS-Path attribute. (Note that the entirety of each Signature-List Block must be checked to ensure that it is well formed, even though the validation process may terminate before all signatures are cryptographically verified.)

If there are two Signature-List Blocks within the BGPSEC_Path_Signatures attribute and one of them is poorly formed (or contains the wrong number of Signature-Segments) , then the recipient should log that an error occurred, strip off that particular Signature-List Block and process the update message as though it arrived with a single Signature-List Block. If the BGPSEC_Path_Signatures attribute contains a syntax error which is not local to a single Signature-List Block, or if the AS-Path attribute contains an AS-Set segment, then the recipient should log that an error occurred, strip off the BGPSEC_Path_Signatures attribute and process the update message as though it arrived without a BGPSEC_Path_Signatures attribute.

Second, the BGPSEC speaker verifies that the update message has not yet expired. To do this, locate the Expire Time field in the BGPSEC_Path_Signatures attribute, and compare it with the current time. If the current time is later than the Expire Time, the BGPSEC update is 'Not Good' and the validation algorithm terminates.

Third, the BGPSEC speaker verifies that the origin AS is authorized to advertise the prefix in question. To do this, consult the valid ROA data to obtain a list of AS numbers that are associated with the given IP address prefix in the update message. Then locate the last (least recently added) AS number in the AS-Path. If the origin AS in the AS-Path is not in the set of AS numbers associated with the given prefix, then BGPSEC update message is 'Not Good' and the validation algorithm terminates.

Finally, the BGPSEC speaker examines the Signature-List Blocks in the BGPSEC_Path_Signatures attribute. Any Signature-List Block corresponding to an algorithm suite that the BGPSEC speaker does not support MUST be discarded. If all Signature-List Blocks are

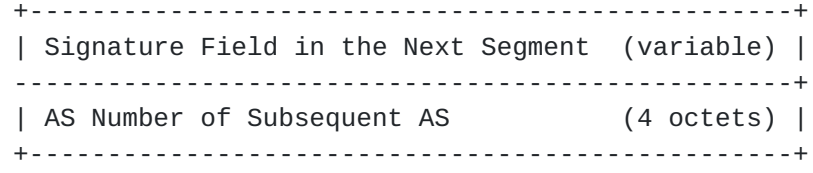
discarded in this manner then the BGPSEC speaker MUST treat the update message as though it arrived without a BGPSEC_Path_Signatures attribute.

For each remaining Signature-List Block (corresponding to an algorithm suite supported by the BGPSEC speaker), the BGPSEC speaker iterates through the Signature-Segments in the Signature-List block, starting with the most recently added segment (and concluding with the least recently added segment). Note that there is a one-to-one correspondence between Signature-Segments and AS numbers in the AS-Path attribute, and the following steps make use of this correspondence.

- (Step I): Locate the public key needed to verify the signature (in the current Signature-Segment). To do this, consult the valid RPKI end-entity certificate data and look for an SKI that matches the value in the SKI field of the Signature-Segment. If no such SKI value is found in the valid RPKI data then mark the entire Signature-List Block as 'Not Good' and proceed to the next Signature-List Block. Similarly, if the SKI exists but the AS Number associated with the SKI does NOT match the AS Number (in the AS-Path attribute) which corresponds to the current Signature-Segment, then mark the entire Signature-List Block as 'Not Good' and proceed to the next Signature-List Block.

- (Step II): Compute the digest function (for the given algorithm suite) on the appropriate data. If the segment is not the (least recently added) segment corresponding to the origin AS, then the digest function should be computed on the following sequence of octets:

Sequence of Octets to be Hashed



The 'Signature Field in the Next Segment' is the Signature field found in the Signature-Segment that is next to be processed (that is, the next most recently added Signature- Segment).

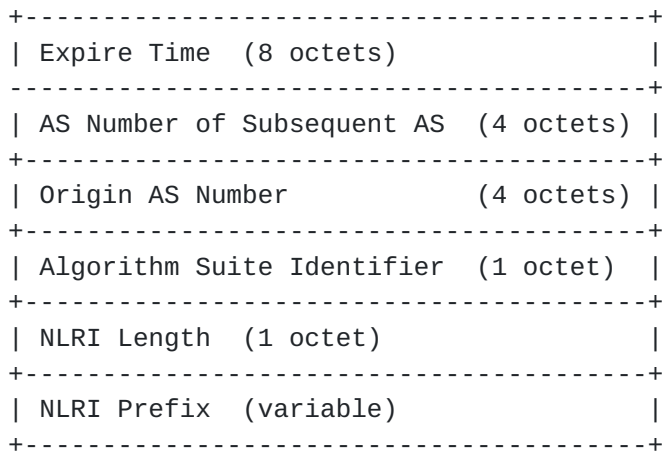
For the first segment to be processed (the most recently added segment), the 'AS Number of Subsequent AS' is the AS number of the BGPSEC speaker validating the update message. Note that if a BGPSEC speaker uses multiple AS Numbers (e.g., the BGPSEC speaker is a member of a confederation), the AS number used here MUST be the AS

number announced in the OPEN message for the BGP session over which the BGPSEC update was received.

For each other Signature-Segment, the 'AS Number of Subsequent AS' is the AS that corresponds to the Signature-Segment added immediately after the one being processed. (That is, find the AS number corresponding to the Signature-Segment currently being processed and the 'AS Number of Subsequent AS' is the next AS number that was added to the AS-Path attribute.)

Alternatively, if the segment being processed corresponds to the origin AS, then the digest function should be computed on the following sequence of octets:

Sequence of Octets to be Hashed



The NLRI Length, NLRI Prefix, Expire Time, and Algorithm Suite Identifier are all obtained in a straight forward manner from the NLRI of the update message or the BGPSEC_Path_Signatures attribute being validated.

The Origin AS Number is the same Origin AS Number that was located in Step I above. (That is, the AS number corresponding to the least recently added Signature-Segment.)

The 'AS Number of Subsequent AS' is the AS Number added to the AS-Path immediately after the Origin AS Number. (That is, the second AS Number that was added to the AS Path.)

o (Step III): Use the signature validation algorithm (for the given algorithm suite) to verify the signature in the current segment. That is, invoke the signature validation algorithm on the following three inputs: the value of the Signature field in the current segment; the digest value computed in Step II above; and

the public key obtained from the valid RPKI data in Step I above. If the signature validation algorithm determines that the signature is invalid, then mark the entire Signature-List Block as 'Not Good' and proceed to the next Signature-List Block. If the signature validation algorithm determines that the signature is valid, then continue processing Signature-Segments (within the current Signature-List Block).

If all Signature-Segments within a Signature-List Block pass validation (i.e., all segments are processed and the Signature-List Block has not yet been marked 'Not Good'), then the Signature-List Block is marked as 'Good'.

If at least one Signature-List Block is marked as 'Good', then the validation algorithm terminates and the BGPSEC update message is deemed to be 'Good'. (That is, if a BGPSEC update message contains two Signature-List Blocks then the update message is deemed 'Good' if the first Signature-List block is marked 'Good' OR the second Signature-List block is marked 'Good'.)

6. Algorithms and Extensibility

6.1. Algorithm Suite Considerations

Note that there is currently no support for bilateral negotiation between BGPSEC peers to use of a particular (digest and signature) algorithm suite using BGP capabilities. This is because the algorithm suite used by the sender of a BGPSEC update message must be understood not only by the peer to whom he is directly sending the message, but also by all BGPSEC speakers to whom the route advertisement is eventually propagated. Therefore, selection of an algorithm suite cannot be a local matter negotiated by BGP peers, but instead must be coordinated throughout the Internet.

To this end, a mandatory algorithm suites document will be created which specifies a mandatory-to-use 'current' algorithm suite for use by all BGPSEC speakers. Additionally, the document specifies an additional 'new' algorithm suite that is recommended to implement.

It is anticipated that in the future the mandatory algorithm suites document will be updated to specify a transition from the 'current' algorithm suite to the 'new' algorithm suite. During the period of transition (likely a small number of years), all BGPSEC update messages SHOULD simultaneously use both the 'current' algorithm suite and the 'new' algorithm suite. (Note that Sections 3 and 4 specify how the BGPSEC_Path_Signatures attribute can contain signatures, in parallel, for two algorithm suites.) Once the transition is

complete, use of the old 'current' algorithm will be deprecated, use of the 'new' algorithm will be mandatory, and a subsequent 'even newer' algorithm suite may be specified as recommend to implement. Once the transition has successfully been completed in this manner, BGPSEC speakers SHOULD include only a single Signature-List Block (corresponding to the 'new' algorithm).

6.2. Extensibility Considerations

This section discusses potential changes to BGPSEC that would require substantial changes to the processing of the BGPSEC_Path_Signatures and thus necessitate a new version of BGPSEC. Examples of such changes include:

- A new type of signature algorithm that produces signatures of variable length
- A new type of signature algorithm for which the number of signatures in the Signature-List Block is not equal to the number of ASes in the AS-PATH (e.g., aggregate signatures)
- Changes to the data that is protected by the BGPSEC signatures (e.g., protection of attributes other than AS-PATH)

In the case that such a change to BGPSEC were deemed desirable, it is expected that a subsequent version of BGPSEC would be created and that this version of BGPSEC would specify a new BGP Path Attribute, let's call it BGPSEC_PATH_SIG_TWO, which is designed to accommodate the desired changes to BGPSEC. In such a case, the mandatory algorithm suites document would be updated to specify algorithm suites appropriate for the new version of BGPSEC.

At this point a transition would begin which is analogous to the algorithm transition discussed in Section 6.2. During the transition period all BGPSEC speakers SHOULD simultaneously include both the BGPSEC_PATH_SIGNATURES attribute and the new BGPSEC_PATH_SIG_TWO attribute. Once the transition is complete, the use of BGPSEC_PATH_SIGNATURES could then be deprecated, at which point BGPSEC speakers SHOULD include only the new BGPSEC_PATH_SIG_TWO attribute. Such a process could facilitate a transition to a new BGPSEC semantics in a backwards compatible fashion.

7. Security Considerations

For discussion of the BGPSEC threat model and related security considerations, please see [8].

A BGPSEC speaker who receives a valid BGPSEC update message, containing a route advertisement for a given prefix, is provided with the following security guarantees:

- The origin AS number corresponds to an autonomous system that has been authorized by the IP address space holder to originate route advertisements for the given prefix.

- For each subsequent AS number in the AS-Path, a BGPSEC speaker authorized by the holder of the AS number selected the given route as the best route to the given prefix.

- For each AS number in the AS Path, a BGPSEC speaker authorized by the holder of the AS number intentionally propagated the route advertisement to the next AS in the AS-Path.

That is, the recipient of a valid BGPSEC Update message is assured that the AS-Path corresponds to a sequence of autonomous systems who have all agreed in principle to forward packets to the given prefix along the indicated path. (It should be noted BGPSEC does not offer a precise guarantee that the data packets would propagate along the indicated path; it only guarantees that the BGP update conveying the path indeed propagated along the indicated path.) Furthermore, the recipient is assured that this path terminates in an autonomous system that has been authorized by the IP address space holder as a legitimate destination for traffic to the given prefix.

Note that there may be cases where a BGPSEC speaker deems 'Good' (as per the validation algorithm in Section 5.1) a BGPSEC update message that contains both a 'Good' and a 'Not Good' Signature-List Block. That is, the update message contains two sets of signatures corresponding to two algorithm suites, and one set of signatures verifies correctly and the other set of signatures fails to verify. In this case, the protocol specifies that if the BGPSEC speaker propagates the route advertisement received in such an update message then the BGPSEC speaker SHOULD add its signature to each of the Signature-List Blocks using both the corresponding algorithm suite. Thus the BGPSEC speaker creates a signature using both algorithm suites and creates a new update message that contains both the 'Good' and the 'Not Good' set of signatures (from its own vantage point).

To understand the reason for such a design decision consider the case where the BGPSEC speaker receives an update message with both a set of algorithm A signatures which are 'Good' and a set of algorithm B signatures which are 'Not Good'. In such a case it is possible (perhaps even quite likely) that some of the BGPSEC speaker's peers (or other entities further 'downstream' in the BGP topology) do not support algorithm A. Therefore, if the BGPSEC speaker were to remove

the 'Not Good' set of signatures corresponding to algorithm B, such entities would treat the message as though it were unsigned. By including the 'Not Good' set of signatures when propagating a route advertisement, the BGPSEC speaker ensures that 'downstream' entities have as much information as possible to make an informed opinion about the validation status of a BGPSEC update.

Note also that during a period of partial BGPSEC deployment, a 'downstream' entity might reasonably treat unsigned messages different from BGPSEC updates that contain a single set of 'Not Good' signatures. That is, by removing the set of 'Not Good' signatures the BGPSEC speaker might actually cause a downstream entity to 'upgrade' the status of a route advertisement from 'Not Good' to unsigned. Finally, note that in the above scenario, the BGPSEC speaker might have deemed algorithm A signatures 'Good' only because of some issue with RPKI state local to his AS (for example, his AS might not yet have obtained a CRL indicating that a key used to verify an algorithm A signature belongs to a newly revoked certificate). In such a case, it is highly desirable for a downstream entity to treat the update as 'Not Good' (due to the revocation) and not as 'unsigned' (which would happen if the 'Not Good' Signature-List Blocks were removed).

A similar argument applies to the case where a BGPSEC speaker (for some reason such as lack of viable alternatives) selects as his best route to a given prefix a route obtained via a 'Not Good' BGPSEC update message. (That is, a BGPSEC update containing only 'Not Good' Signature-List Blocks.) In such a case, the BGPSEC speaker should propagate a signed BGPSEC update message, adding his signature to the 'Not Good' signatures that already exist. Again, this is to ensure that 'downstream' entities are able to make an informed decision and not erroneously treat the route as unsigned. It may also be noted here that due to possible differences in RPKI data at different vantage points in the network, a BGPSEC update that was deemed 'Not Good' at an upstream BGPSEC speaker may indeed be deemed 'Good' at another BGP speaker downstream.

Therefore, it is important to note that when a BGPSEC speaker signs an outgoing update message, it is not attesting to a belief that all signatures prior to its are valid. Instead it is merely asserting that:

1. The BGPSEC speaker received the given route advertisement with the indicated NLRI and AS Path;

2. The BGPSEC speaker selected this route as the best route to the given prefix; and

3. The BGPSEC speaker chose to propagate an advertisement for this route to the peer (implicitly) indicated by the 'Target AS'

The BGPSEC update validation procedure is a potential target for denial of service attacks against a BGPSEC speaker. To mitigate the effectiveness of such denial of service attacks, BGPSEC speakers should implement an update validation algorithm that performs expensive checks (e.g., signature verification) after less expensive checks (e.g., syntax checks). The validation algorithm specified in Section 5.1 was chosen so as to perform checks which are likely to be expensive after checks that are likely to be inexpensive. However, the relative cost of performing required validation steps may vary between implementations, and thus the algorithm specified in Section 5.1 may not provide the best denial of service protection for all implementations.

8. Contributors

8.1. Authors

Rob Austein
Internet Systems Consortium
sra@hactrn.net

Steven Bellovin
Columbia University
smb@cs.columbia.edu

Randy Bush
Internet Initiative Japan
randy@psg.com

Russ Housley
Vigil Security
housley@vigilsec.com

Stephen Kent
BBN Technologies
kent@bbn.com

Warren Kumari
Google

warren@kumari.net

Doug Montgomery
USA National Institute of Standards and Technology
dougm@nist.gov

Kotikalapudi Sriram
USA National Institute of Standards and Technology
kotikalapudi.sriram@nist.gov

Samuel Weiler
weiler@watson.org
Cobham

8.2. Acknowledgements

The authors would like to thank Sharon Goldberg, Ed Kern, Chris Morrow, Sandy Murphy, Mark Reynolds, Heather Schiller, Jason Schiller, John Scudder, and David Ward for their valuable input and review.

9. References

[1] Jonsson, J. and B. Kaliski, "PKCS #1", RFC 3447, February 2003.

[2] Bates, T., Chandra, R., Katz, D., and Y. Rekhter, "Multiprotocol Extensions for BGP-4", RFC 4760, January 2007.

[3] Scudder, J. and R. Chandra, "Capabilities Advertisement with BGP-4", RFC 4760, February 2009.

[4] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.

[5] Patel, K., Ward, D., and R. Bush, "Extended Message support for BGP", draft-ymbk-bgp-extended-messages, March 2011.

[6] Lepinski, M., Kent, S., and D. Kong, "A Profile for Route Origin Authorizations", draft-ietf-sidr-roa-format, February 2011.

[7] Lepinski, M. and S. Kent, "An Infrastructure to Support Secure Internet Routing", draft-ietf-sidr-arch, February 2011.

[8] Kent, S., "Threat Model for BGP Path Security", draft-kent-bgpsec-threats, February 2011.

Author's Address

(Editor) Matthew Lepinski
BBN
10 Moulton St
Cambridge, MA 55409

Phone: +1-617-873-5939
Email: mlepinski@bbn.com