Analysis of the impact of BGPsec on the amount of generated BGP update messages

Lisa Bruder

lisa.bruder@os3.nl

Research Project - Security and Network Engineering - Universiteit van Amsterdam

Supervisors: Ralph Koning (*UvA/SIDN Labs*), Moritz Müller (*UT/SIDN Labs*)

***Abstract*—This report analyses the impact of BGPsec on BGP update messages. BGPsec is an extension to BGP that aims to improve routing security by allowing BGP speakers to sign and validate AS paths. We make use of BGP data streams from two BGP speakers and evaluate how different identified types of BGP update messages are affected by the changes required by BGPsec. In addition to the lack of support for update packing, these changes include the need to tailor each update message to the receiving peer by adding its AS number to the path. We model BGPsec traffic based on these requirements. We analyse the impact of BGPsec on the generation of update messages and conclude that the increased number of messages is likely to significantly increase the computational load on routers running BGP, even without considering the computational cost of signing. However, we find that the average number of prefixes advertised per update message is lower than the ones determined in previous work, which may indicate a lower than expected impact of BGPsec on BGP traffic.**

Keywords: BGP, BGPsec, inter-domain routing, routing security

I. Introduction

Border Gateway Protocol (BGP) is the standard protocol used for inter-domain routing on the Internet [1]. While it plays a crucial role in the global Internet routing infrastructure, it was not originally designed with security in mind. This has made it susceptible to attacks like prefix hijacking, AS path forgery and route leaking. Intentional and unintentional misconfigurations have led to several high-profile incidents that have disrupted routing behaviour [2]. An example of this occurred in 2018, when China Telecom announced a part of Google's IP address range making Google services unreachable for customers [3].

The BGPsec extension attempts to improve the security of BGP. It does so by requiring BGP speakers to sign the asserted AS path included in a BGP update message with their private key. This signature can later be validated with the corresponding public key.

While improving BGP security, BGPsec comes with higher deployment costs [1]. Next to additional computational and memory requirements for BGPsec speaking routers, it is expected that the number of update messages will increase with BGPsec deployment due to the required changes in update message generation [1]. One of the main changes is the lack of support for update packing when using the BGPsec extension, meaning that it is not possible to announce multiple prefixes in one update message [4]. This paper aims to quantify this increase of update messages by modeling expected BGPsec traffic based on BGP traffic data from a customer AS of SIDN Labs[1] and one of the AMS-IX[2] route servers.

The rest of the paper is structured as follows. In sec. II, we outline BGP, different BGP security mechanisms and introduce the BGPsec extension. In sec. III we introduce related work on BGPsec optimisations and prior research on the potential increase in BGP update messages. We then present our methodology in sec. IV followed by an overview of our results in sec. VI. Finally, we discuss our results in sec. VII and conclude what we found about the impact of BGPsec on BGP update messages in sec. VIII.

A. Research Questions

We identified the following research question and corresponding sub-research questions for this project:

Research Question: *"What is the effect of BGPsec on update messages exchanged between BGP speakers?"*

- **Sub-Question 1:** *"What type of BGP update messages exist and how would BGPsec affect how they need to be generated?"*
- **Sub-Question 2:** *"How many more BGP update messages must be sent by a BGP speaker when using BGPsec due to not being able to do update packing?"*
- **Sub-Question 3:** *"What, if any, other requirements by BGPsec impact BGP update messages?*

II. Background

A. BGP

BGP is an inter-Autonomous System routing protocol and the current de-facto standard for inter-domain routing [1]. A router implementing BGP is called a BGP speaker [5]. To enable routing between different Autonomous Systems (ASes), BGP speakers exchange BGP messages containing routing information that they use to populate their Routing Information Bases (RIBs) [5]. When a BGP speaker receives new information on a route, it updates its own RIB and forwards this new information about the network topology to its peers. However, it can take some time until all routers reflect this change. This period is referred to as converging and the time it takes until convergence is reached should be

[1] *https://www.sidnlabs.nl/*

[2] *https://www.ams-ix.net/*

kept as short as possible to ensure stable routing behaviour [1,6].

As specified in RFC4271, there are four types of BGP messages: open (type 1), update (type 2), notification (type 3) and keep-alive (type 4) [5]. Update messages are used to exchange routing information. They include a Network Layer Reachability Information (NLRI) field and several path attributes, one of them being the *AS_PATH* attribute. The NLRI field contains one or more announced prefixes and their length. The *AS_PATH* consists of a list of Autonomous System Number (ASN) representing the path that needs to be traversed to reach the given prefix(es) [5]. Based on this information BGP speakers can determine the best path to reach a destination.

While this decentralised system works well with co-operative actors, BGP has no built-in security mechanisms against hostile actors joining the network [7]. With BGP, there is no way to verify received routing information. For this reason, intentional and unintentional misconfigurations can lead to disruptions if incorrect information is propagated throughout the network.

B. Proposed BGP security mechanisms

Wang et al. classify technologies attempting to improve inter-domain routing security into Route Origin Verification (ROV), Route Path Plausibility (RPP) and Route Path Verification (RPV) [8].

Approaches for ROV intend to mitigate prefix hijacking attacks. As defined by Mitseva et al., to execute a prefix hijacking attack, “an AS falsely claims to originate a prefix not delegated to it” [9]. The AS can also advertise a subnetwork of a prefix, in that case it is a subprefix hijacking attack. One approach to mitigate these attacks is the Resource Public Key Infrastructure (RPKI). The RPKI is a Public Key Infrastructure (PKI) that can be used for ROV [10]. RPKI objects contain certificates and Route Origin Authorisation (ROA) objects binding them to ASes. This allows BGP peers receiving a BGP message to validate if the announcing AS is authorised to advertise the sent prefix by looking up the associated certificate in the RPKI [10]. However, ROA objects do not allow validation of the asserted AS path to the prefix.

RPP and RPV mechanisms focus on the validation of the *AS_PATH* attribute. Autonomous System Provider Authorisation (ASPA) is the currently most mature approach for RPP [11]. ASPA is defined in a current Internet-Draft [12] and aims to allow verification of the *AS_PATH* attribute. This is supposed to mitigate route leaks and help BGP speakers detect improbable AS paths. To achieve this, ASPA does not rely on cryptographically signing the path but purely evaluates if a path is plausible based on AS relationship information included in ASPA objects [11]. ASPA objects are signed. Cryptographic validation takes place on dedicated machines and not on the routers themselves lowering the performance requirements that routers need to meet [11].

One approach for RPV is BGPsec. BGPsec is an extension of BGP that requires BGP speakers to cryptographically sign the asserted AS path in an update message with a private key corresponding to a certificate stored in the RPKI. Based on that receiving BGP peers are able to verify the received path (RPV) [8].

C. BGPsec

BGPsec was specified in RFC8205 [13] in 2017. It is a BGP extension that aims to make the AS path cryptographically verifiable. The adoption of BGPsec has been slow due to concerns over performance degradation and slower convergence in the BGP network [1]. The use of BGPsec is expected to result in increased computational overhead, increased memory requirements and an increase in the number of update messages.

As mentioned before, it is expected that BGPsec will demand an increase in update messages because of the required modifications to the process of generating valid messages. BGPsec replaces the *AS_PATH* attribute with the *BGPsec_PATH* attribute. The *BGPsec_PATH* attribute contains a *Secure_Path* and a signature block. The *Secure_Path* contains the list of ASNs, associated flags and a *pCount* that allows specifying the number of times the ASN should be included. The *pCount* is used for path prepending, as used in BGP. Setting the *pCount* allows the speaker to achieve the same semantics without the additional processing overhead of having to generate multiple *Secure_Path* segments [13].

The signature block contains each of the traversed ASes signatures as well as their Subject Key Identifier related to the RPKI router certificate [13]. In contrast to the *AS_PATH* attribute, *BGPsec_PATH* must include the ASN of the peer to whom the update message is sent. This means that a generated BGPsec update message announcing a prefix and an associated AS path can not be sent to multiple peers. A new message needs to be generated and signed for each peer the message is sent to [13].

Additionally, while BGP allows for so-called update packing, several prefixes with the same attributes being announced in one update message, the BGPsec extension does not support this. This means each BGPsec update can contain only exactly one prefix [4,13]. This decision was made because if a BGP speaker wanted to create an update containing only a subset of the packed announced prefixes in a received message, this would greatly increase the complexity of message creation [14].

Finally, because BGPsec messages are signed with a private key that is associated with a certificate, it has to be ensured that keys are rolled over and messages are regenerated and resigned with a new key before a certificate expires. While key rollovers can be scheduled as planned based on the expiration of a certificate, they can also be required due to changes in certificate data or a compromised key [15]. Any messages that were previously signed with a no longer valid key, might be treated as unauthenticated by receiving BGPsec speakers. This in turn means that while in BGP, announced routes are valid until they are actively withdrawn, this is not the case with BGPsec [16].

III. Related Work

As discussed in sec. II-C, some of the requirements set by BGPsec lead to higher deployment costs. These include an increased computational load on routers, higher memory requirements and an expected decrease in speed with regards to message creation and validation. Due to that, adoption of BGPsec has been slow [1]. To mitigate these deployment costs, there are several works related to optimising operations required by BGPsec.

Sriram and Montgomery (2017) analyse three update processing algorithms to lower processing costs of update messages [14]. They evaluate optimising the verification process by caching parts of the AS path that have already been verified (Cache Common Segments) and only verifying the signatures of updates that the BGP speaker determines to be best path considerations (Best Path Only).

Kim and Kim (2015) compare BGPsec signature algorithms and advocate for the usage of RSA instead of ECDSA to improve performance when validating BGPsec update messages [2]. They determine that while ECDSA has performance benefits, in the context of BGPsec, this is less relevant. RSA allows for quick verification, which leads to a better performance when it comes to validating BGP update messages.

Takemura et al. (2021) research options to lower the memory required for routers to run the BGPsec extension. They propose the aggregation of signatures with their protocol APVAS+. APVAS+ lowers memory requirements in certain topologies to a level that is much closer to memory available in currently used routers [17].

Next to the mentioned deployment costs, another reason for concern is the expected increase in the amount of required update messages. There are some related works mentioning this and estimating its potential impact.

Huston and Bush (2011) published an article called "Securing BGP with BGPsec" in *The ISP Column* examining the BGPsec mechanisms as well as the implications they could have on BGP operation [4]. Next to the potential impact on the size of update messages and the computational load validation of the signatures could put on BGPsec speaking routers, they also mention the potential impact on BGP traffic caused by the lack of support for update packing. They suggest that while there will be an increase, it will not come "at an unreasonable cost" [4].

Sriram et al. held a presentation on "RIB Size Estimation for BGPSEC" in 2011 including an analysis on the amount of prefixes announced in eBGP announcements. They found an average of 3.832 prefixes packed in update messages. Based on this, RFC8374 discusses that there are on average four times fewer messages than announced prefixes [18].

Oesterle et al. published a paper titled "Challenges with BGPsec" in 2021, reviewing deployment challenges with BGPsec. Next to concerns about BGPsec being slower than BGP due to performed signature validations, they also mention the lack of support for update packing as a reason for an increase in update messages. They identify that these two factors could impact processing time and with that slow down convergence of BGP [1].

We wanted to get a more detailed insight into how much the number of BGP update messages would increase with BGPsec deployment. To the best of our knowledge, no work has been done on this yet.

IV. Methodology

To determine the impact of BGPsec on BGP update messages, we analyse BGP update messages and identify modifications required by BGPsec. Based on that, we estimate the impact that BGPsec would have on BGP traffic and the amount of messages that a router needs to generate. This section describes the steps we took in detail.

A. Identifying necessary additional BGPsec update messages

To identify required additional BGPsec update messages based on existing BGP traffic data, we split up update messages into three categories that aid in the analysis. The first category includes BGP updates that announce exactly one prefix. An example of an NLRI field containing one prefix is shown in lst. 1. BGPsec requires these messages to be adapted and resigned for each destination AS. This impacts message generation.

```
Network Layer Reachability Information (NLRI)
    x.x.x.x/24
        NLRI prefix length: 24
        NLRI prefix: x.x.x.x
```

Listing 1. IPv4 NLRI that includes exactly one prefix.

The second category contains BGP updates that contain two or more prefixes. An example for an NLRI field containing three prefixes is shown in lst. 2. Similar to the first category, these messages need to be adapted for each destination AS and can not be just generated once and sent to multiple peers. In addition, the announced prefixes must be separated into several individual BGPsec messages. This has an impact on the amount of BGP traffic.

```
Network Layer Reachability Information (NLRI)
    x.x.x.x/24
        NLRI prefix length: 24
        NLRI prefix: x.x.x.x
    x.x.x.x/22
        NLRI prefix length: 22
        NLRI prefix: x.x.x.x
    x.x.x.x/24
        NLRI prefix length: 24
        NLRI prefix: x.x.x.x
```

Listing 2. IPv4 NLRI that includes more than one prefix.

Finally, the third category includes messages that do not announce any prefix. These are called withdrawal messages. An example is shown in lst. 3. BGPsec does not require withdrawals to be signed, which is why no changes are required for messages in this category.

```
Withdrawn Routes
        x.x.x.x/24
            Withdrawn route prefix length: 24
            Withdrawn prefix: x.x.x.x
```

Listing 3. IPv4 withdrawal message.

B. Data selection and preparation

To analyse BGP traffic we use two data streams available to us. The first one is BGP traffic from a BGP session with the BGP speaker at *SIDN Labs* (AS 215088), from now on referred to as *Customer AS*. This AS is a downstream from two *SURF* ASes (AS 1101 and AS 1103). *SURF* is the national research and education network of the Netherlands.[3] The data used was collected in the time frame 2024-06-10 18:30 to 2024-06-11 18:00. Our second data source is BGP traffic from the AMS-IX route server 1 (AS 6777) in the time frame 2024-06-06 00:02 to 2024-06-07 00:02, from now on referred to as *AMS-IX route server*. This BGP speaker is an example of an Internet Exchange. The statistics on the AMS-IX website declare a total of 886 IPv4 and 795 IPv6 peers.[4] AMS-IX is “one of the largest public peering interconnection platforms”.[5] According to statistics published by AMS-IX, route server 1 has received 329,858 IPv4 and 69,000 IPv6 prefixes and has sent 206,798 IPv4 and 45,978 IPv6 prefixes.[3]

We filter the traffic data by the source IP addresses for the BGP speakers of AS 215088 and AS 6777. Through that we discard all incoming traffic. We do this to be able to focus on the impact that BGPsec has on update messages generated and sent by these two speakers. In addition, we filter by BGP message type 2 (update) and separate update messages into two groups: messages carrying IPv4 routing information and messages carrying IPv6 routing information.

Finally, we identify the amount of prefixes contained in each update message and add a field containing this count to the data. A withdrawal message is assigned a prefix count of 0.

C. Analysing impact on traffic

To estimate the impact that the use of BGPsec would have on the amount of BGP traffic, we make use of the counted prefixes per message. We base our analysis on the assumption that each message that contains more than one prefix needs to be split up into as many messages as prefixes are announced in that message. Therefore, our estimation for BGPsec messages per time interval (M_{BGPsec}) equates to the sum over all counted prefixes within the given time interval (P_{BGP}) plus the count of withdrawal messages in that same time interval (W_{BGP}). This is represented by the following formula:

$$M_{BGPsec} = \sum P_{BGP} + W_{BGP}$$

We compare the increase in BGP traffic by comparing the total amount of update messages sent in our data to our estimation for BGPsec (M_{BGPsec}). In addition, we compare the peak load of update messages sent per minute.

D. Analysing impact on message generation

To estimate the impact that BGPsec would have on BGP message generation by the router, we only analyse the *AMS-IX route server* data. In contrast to the data from the peering session with the *Customer AS*, the AMS-IX data includes messages that are sent to several peers, which allows us to make estimations on how many more messages would need to be generated.

We make the following assumptions when estimating the number of messages generated by the *AMS-IX route server*: We consider identical BGP update messages in our traffic data, where identical means that every field is exactly the same, to have been generated once if the timestamps of the frames containing them are no more than 100ms apart. We set the timestamp of message generation to be the timestamp of the first occurrence of a message.

We choose this specific time frame of 100ms due to computational limitations. If we extended it to 500ms or 1000ms, we would not be able to process the IPv4 data efficiently due to the large number of message comparisons required when a large number of messages are sent in a very short time frame.

In order to analyse update generation in more detail, it would be necessary to modify the BGP speaker in question to output trace messages as update messages are generated and sent. This is not feasible with the *AMS-IX route server* within the scope of this project.

To estimate required BGPsec message generation, we count the number of announced prefixes in the BGP traffic data and based on that estimate additional required messages. Because our traffic data already reflects how many peers receive a message, we make use of this count for our message generation estimation: each message sent that is not a withdrawal message needs to be generated separately. We determine the amount of generated withdrawal messages per time interval and add them to the count. This is expressed in the following formula, where P_{BGP} represents the count of prefixes over all sent BGP update messages, $WGen_{BGP}$ represents the estimation for generated withdrawals and $MGen_{BGPsec}$ represents the estimation for generated BGPsec messages.

$$MGen_{BGPsec} = \sum P_{BGP} + WGen_{BGP}$$

V. Implementation

To conduct our analysis based on the methodology presented in sec. IV, we use tcpdump[6] to capture BGP traffic from AS 215088 (SIDN Labs) and tshark[7] to filter and reformat packet captures. In addition, we used the Python programming language[8] as well as the ELK stack[9] (Elasticsearch, Logstash, Kibana) to analyse and visualise the data. An overview of the processing workflow can be found in fig. 1.

After the first processing step (pcap to JSON), the *AMS-IX route server* data takes up 463GB and the *Customer AS* data takes up 3.4GB of storage. To be able to efficiently process and analyse the data, we use a VM with 12 cores on Xeon Gold 5115 CPU @ 2.40GHz with 1.2TB of storage and 400GB

[3] *https://www.surf.nl/*

[4] *https://stats.ams-ix.net/rs-stats.html*

[5] *https://www.ams-ix.net/ams/service/internet-peering*

[6] *https://www.tcpdump.org/*

[7] *https://tshark.dev/*

[8] *https://www.python.org/*

[9] *https://www.elastic.co/elastic-stack/*

TABLE I
IMPACT ON BGP UPDATE MESSAGE TRAFFIC

Customer AS

	Total # updates			**Peak load (per min)**		
IP version	BGP	BGPsec	increase by[1]	BGP	BGPsec	increase by[1]
IPv4	888,174	2,007,969	126.08%	5,757	31,849	453.22%
IPv6	978,860	1,287,344	31.51%	1,706	8,158	378.19%

AMS-IX route server

	Total # updates			**Peak load (per min)**		
IP version	BGP	BGPsec	increase by[1]	BGP	BGPsec	increase by[1]
IPv4	126,813,802	282,403,732	122.69%	2,581,402	3,166,746	22.68%
IPv6	49,760,361	74,929,830	50.58%	356,845	796,900	123.32%

[1] Rounded to the second decimal place

TABLE II
IMPACT ON BGP UPDATE MESSAGE GENERATION

AMS-IX route server

	Total # updates generated			**Peak load (per min)**		
IP version	BGP	BGPsec	increase by[1]	BGP	BGPsec	increase by[1]
IPv4	83,415,053	277,631,142	232.83%	2,207,170	3,143,740	42.43%
IPv6	14,520,796	70,415,274	384.93%	312,952	791,543	152.93%

[1] Rounded to the second decimal place

of RAM. All scripts and Kibana dashboards can be found on GitHub[10].

The captured data for both BGP speakers is in pcap format. We use tshark to convert the pcaps into a JSON format suitable for importing into Elasticsearch using the *-T ek* option. To allow more efficient processing of the data, we discard all fields in the frame except for the timestamp and the IP and BGP fields using the *-J "bgp ip"* option. We filter the traffic data by the source IP addresses of the BGP speakers for AS 215088 and AS 6777 respectively and filter by BGP message type 2, leaving us with outgoing BGP update messages. In order to separate the traffic into IPv4 and IPv6, we filter on the IPv4 source IP address and the IPv6 source IP address separately.

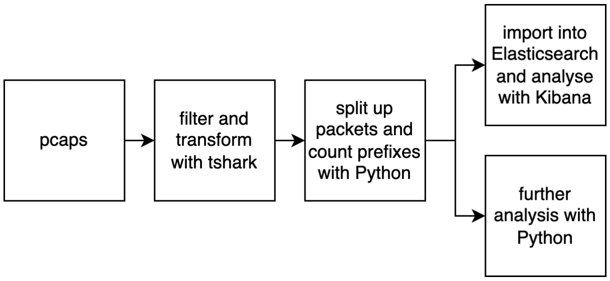

pcaps
filter and transform with tshark
split up packets and count prefixes with Python
import into Elasticsearch and analyse with Kibana
further analysis with Python

Fig. 1. Overview of data preparation and analysis steps.

As a final data preparation step, we use Python to split up frames containing more than one BGP update message into separate JSON objects to ensure that each document in Elasticsearch will represent exactly one update message and not multiple. In the same process, we count the amount of prefixes contained in each update message. Update messages carrying IPv6 routing information make use of different fields than messages carrying IPv4 routing information due to the changes required by the Multiprotocol extension [19]. To account for that, we count the prefixes contained in the *NLRI* field for IPv4 and the prefixes contained in the *MP_REACH_NLRI* field for IPv6.

We import the data into Elasticsearch using Logstash to perform data analysis and create visualisations in Kibana. To generate estimates for update message generation, we process the *AMS-IX route server* data using a Python script that checks for duplicate update messages and compares their timestamps.

VI. RESULTS

We summarise the findings of our analysis on update packing in table III. All results related to the impact of BGPsec on BGP traffic can be found in table I, all results related to the impact on BGP message generation can be found in table II.

A. Update packing

We determine the average, median, maximum, and standard deviation for the number of prefixes per update message for IPv4 and IPv6 for both BGP speakers. For the *Customer AS*, we find an average of 2.266 announced prefixes per update message for IPv4 and an average of 1.347 for IPv6. The median for both is 1. Excluding withdrawals, update messages with one announced prefix make up 79.06% of IPv4 and 87.46% of IPv6 update messages. The maximum amount of prefixes in an update message is 999 for IPv4 and 558 for IPv6. We find that IPv4 has a higher standard deviation than IPv6.

For the *AMS-IX route server*, we find an average of 2.281 for IPv4 update messages and an average of 1.558 for IPv6 update messages. Again, the median for both is 1. Excluding withdrawals, update messages with one announced prefix make

[10] *https://github.com/lmbruder/bgpsec_update_messages*

up 77.81% of IPv4 and 81.91% of IPv6 update messages. We determine the maximum number of prefixes in a message to be 1,010 for IPv4 and 570 for IPv6. Again the standard deviation is higher for IPv4 than for IPv6 with 10.351 compared to 7.247.

TABLE III
OVERVIEW OF UPDATE PACKING

Customer AS

IP version	Average	Median	Maximum	SD[1]
IPv4	2.266	1	999	8.783
IPv6	1.347	1	558	4.691

AMS-IX route server

IP version	Average	Median	Maximum	SD[1]
IPv4	2.281	1	1,010	10.351
IPv6	1.558	1	570	7.247

[1] Standard Deviation

B. Impact on BGP traffic

We find an increase by 126.08% for total IPv4 update messages sent from the *Customer AS*. For IPv6, we find a smaller increase by 31.51%. Diagram (a) in fig. 2 shows our estimation for IPv4 traffic from the *Customer AS* with BGPsec compared to without BGPsec per 30 minutes. While traffic without BGPsec never reaches more than 40,000 update messages in 30 minutes, our estimation shows peaks of up to around 100,000 update messages in 30 minutes for BGPsec. For all estimated BGPsec spikes in the graph with over 80,000 update messages, we find that the average number of prefixes per BGP update message in the BGP traffic data is over 3.5. This is much higher than the overall average of 2.266.

Fig. 2 (b) shows the same visualisation for IPv6. Similarly to the IPv4 data, the spikes at 20:30 and 21:00 have a higher average number of prefixes per message (over 1.8) compared to the overall average (1.347) explaining the higher increase in required messages in the BGPsec estimation. The spike at 18:30 is due to a much higher amount of messages sent in that time frame. On a per-minute basis, peak loads for IPv4 increase by 453.22% and for IPv6 by 378.19%.

For the *AMS-IX route server*, we find an increase by 122.69% for total update messages sent. For IPv6, we find an increase by 50.58%. With regards to peak load per minute, we find an increase by only 22.68% for IPv4 but an increase by 123.32% for IPv6. Fig. 2 (c) shows the BGPsec estimation compared to the BGP traffic for IPv4. We find that the BGP data shows a recurring pattern of about an hour of higher traffic (about 4,000,000) followed by about an hour of lower traffic (about 2,000,000). Fig. 2 (d) shows the same graph for IPv6. We see that the spikes at 10:00 and 13:30 are again related to a higher average number of prefixes per message in these time frames compared to the overall average (2.49 and 2.709).

C. Impact on message generation

To estimate the impact on message generation, we determine duplicate messages that are sent to different peers as described in sec. IV. Using this method, we filter out 43,398,749 messages from the IPv4 traffic data and 35,239,565 messages from the IPv6 traffic data. Within the analysed time frame 790 IPv4 peers and 680 IPv6 peers receive update messages from the router.

We determine that 83,415,053 IPv4 update messages were generated during the 24 hour time frame captured in our data. In contrast to that, BGPsec requires 277,631,142 messages to be generated. An increase by 232.83%. The peak amount of messages that need to be generated per minute increases as well, however, by a comparably low percentage of 42.43%.

For our IPv6 traffic data, we determine a count of 14,520,796 generated messages with BGP. Our estimation for BGPsec requires an increase by 384.93%. The peak load per minute more than doubles with an increase by 152.93% from 312,952 to 791,543 messages generated.

Fig. 3 shows overviews for message generation per 30 minutes for IPv4 and IPv6 update messages. The estimations show that the impact on IPv6 message generation is much higher than for IPv4. Relative to the number of update messages, many more messages were filtered out for IPv6, leading to a larger estimated increase for BGPsec.

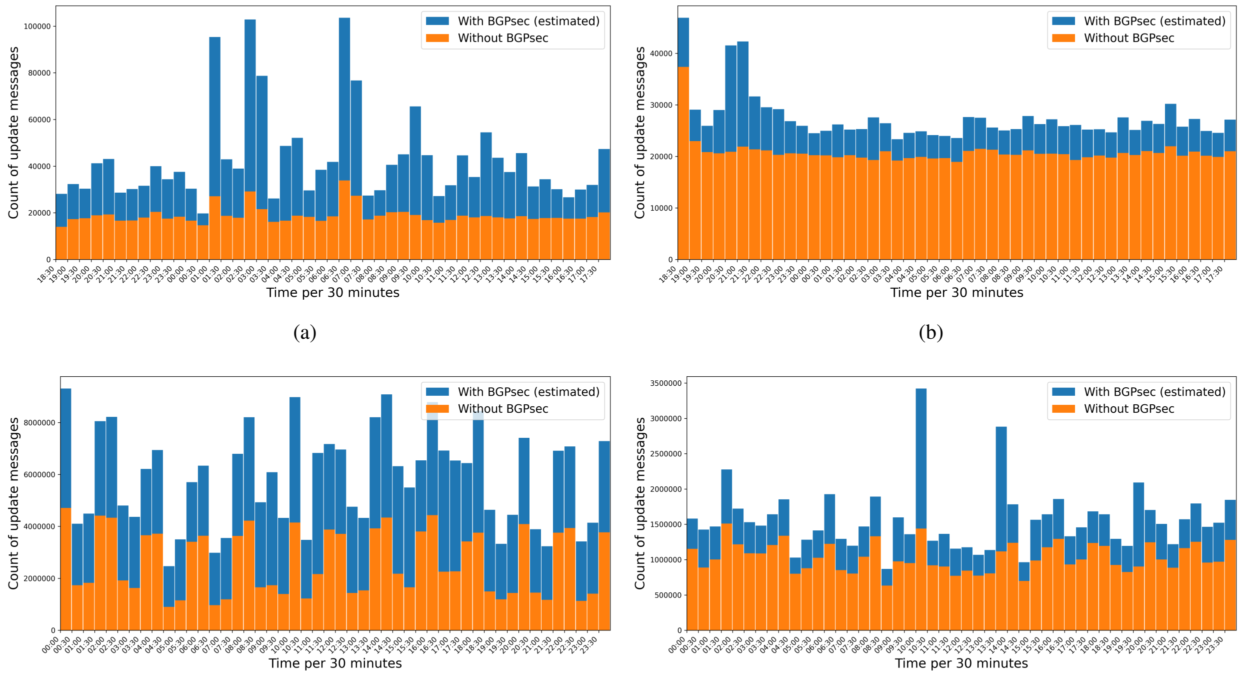
The graph has two outstanding spikes at 10:00 and 13:30. Estimations for 10:00 and 13:30 show increases by 538.55% and 528.03% respectively for the 30-minute time frame. In contrast to that, for a more average time frame like 06:00 - 06:30, we estimate an increase by 336.88%. Next to the higher average of prefixes per update message in the time frames 10:00 and 13:30 as discussed in sec. VI-B, these time frames include more messages that are generated once and then sent to a high number of peers leading to a lower estimation for BGP message generation.

VII. Discussion

The results of our data analysis show that BGPsec would require a higher amount of update messages to exchange the same routing information as BGP. We determine that the amount of additional required messages is highly dependent on the amount of packed updates and the amount of peers that receive the messages. In contrast to earlier work from Sriram et al., we do not find an average of about four prefixes per update message in our data [20].

In our data, the average number of prefixes per update message is between 1.347 and 2.281. Update messages carrying IPv6 routing information contain fewer prefixes on average. In our model, this leads to a more drastic increase in IPv4 BGP traffic than for IPv6. While the average is higher, we find a median of one prefix per update message for both IPv4 and IPv6 traffic. This means that for the vast majority of sent update messages in our data, BGPsec does not require any changes that impact the amount of BGP traffic.

In contrast to other determined values, we find a comparably low increase in peak load per minute of IPv4 traffic for the *AMS-IX route server* (22.68%). It seems as though high peaks in load do not imply a high number of prefixes announced per update message. If the number of prefixes per message is low, BGPsec does not have a big effect on traffic amount even if a high number of messages is sent.



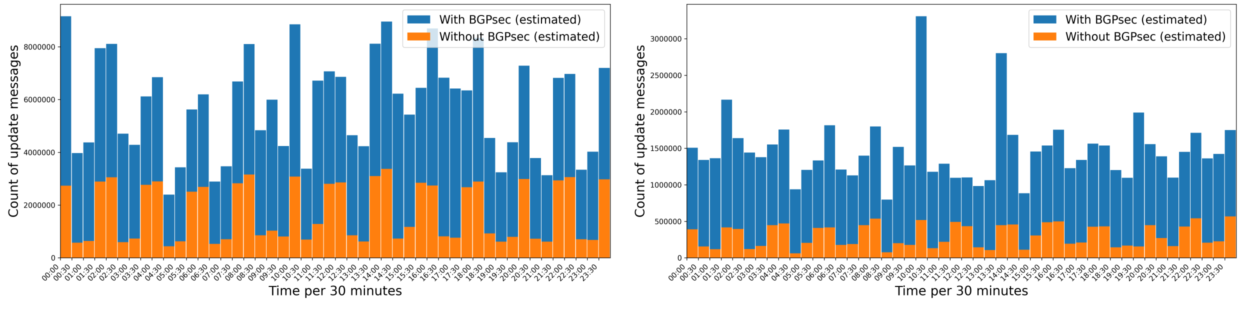
(a)

(b)

(c)

(d)

Fig. 2. **BGP traffic analysis** (a) *Customer AS*: IPv4 BGP update messages sent per 30 minutes. (b) *Customer AS*: IPv6 BGP update messages sent per 30 minutes. (c) *AMS-IX route server*: IPv4 BGP update messages sent per 30 minutes. (d) *AMS-IX route server*: IPv6 BGP update messages sent per 30 minutes.



(a)

(b)

Fig. 3. **BGP message generation analysis** (a) *AMS-IX route server*: IPv4 BGP update messages generated per 30 minutes. (b) *AMS-IX route server*: IPv6 BGP update messages generated per 30 minutes.

We notice that the IPv4 data shows a recurring pattern of higher numbers of messages followed by lower numbers of messages. We were not able to fully investigate this, however, we deem that it could be related to BGP route flapping generated by unstable ASes [21].

Additionally, our results show that a BGP speaker with a lot of peers will have to generate a substantially bigger amount of BGPsec messages compared to the amount generated for BGP to adhere to the BGPsec requirements. In our analysis, we find an increase by 232.83% for IPv4 and 384.93% for IPv6 routing information carrying update messages. As mentioned in sec. III, prior work highlights additional computational requirements that BGPsec would place on routers. Combined with the need to sign and validate update messages, we determine that BGPsec could place more strain on existing routers running BGP.

While our data shows this, these conclusions can not be easily generalised for other BGP speakers and their BGP traffic. The presented results are specific to the two BGP speakers and their peers in the time frame we analysed. These can vary greatly to other BGP speakers even in similar network topologies.

We made assumptions about the operation of the BGP speakers analysed as specified in sec. IV-D. The chosen time frame of 100ms affects our estimates for BGP message generation and with that our estimates for the increase required by BGPsec. This could differ from the actual operation of the router, which could impact the validity of our results.

We did not analyse actual BGPsec traffic, which is why all presented results are only an estimation based on requirements posed by the BGPsec specification. We were not able to analyse actual BGPsec traffic because adoption of it is so low that it does not exist in a practical setting yet. While we did look into current proof-of-concept implementations, we determined that they are relatively outdated and not suitable for the scope of this project.

VIII. Conclusion

In this work, we determined three types of BGP update messages and analysed BGP traffic data to identify the impact the BGPsec extension would have on it. We determined the prevalence of update packing in our data and compared our findings to existing results. In addition, we analysed the impact of BGPsec on update message generation by the router.

While prior work by Sriram et al. that was referenced in RFC8374 on BGPsec design choices, shows an average of four prefixes per update message for BGP, we found lower averages in our data [14,20]. The highest average in our data is 2.281, which we found for the IPv4 data from the *AMS-IX route server*. The median for all analysed data lies at 1. This indicates that the impact of BGPsec on BGP traffic could be lower than expected.

Next to changes required due to update packing, BGPsec also impacts how update messages are generated by the router. We found that update messages are regularly sent to multiple peers thereby confirming the assumption that BGPsec would require additional messages to be generated to be able to adapt them for each peer. Next to that, key rollovers can lead to increased BGPsec traffic because routing information signed with an expired key is no longer valid and needs to be resigned and resent.

Looking ahead, future work could explore reasons for the differences we found between update packing in IPv4 and IPv6 BGP update messages. Moreover, we did not extend our analysis to the impact that a higher number of update messages might have on the number of IP packets and Ethernet frames required to carry this data. Finally, we did not investigate existing BGPsec implementations and did not analyse real-world BGPsec data. Working with actual BGPsec data could refine the insights provided in this report and enable a more accurate evaluation of the effects BGPsec deployment would have.

IX. Acknowledgements

Thanks to Stavros Konstantaras from AMS-IX for kindly providing data for this analysis and SIDN Labs for allowing me to set up a peering session with their BGP speaker to analyse exchanged traffic.

References

[1] Jan Oesterle. Challenges with BGPSec. 2021. Medium: PDF Publisher: Chair of Network Architectures and Services, Department of Computer Science, Technical University of Munich.

[2] Kyoungha Kim and Yanggon Kim. Comparative analysis on the signature algorithms to validate AS paths in BGPsec. In *2015 IEEE/ACIS 14th International Conference on Computer and Information Science (ICIS)*, pages 53–58, June 2015.

[3] ThousandEyes. https://www.thousandeyes.com/blog/internet-vulnerability-takes-down-google, Accessed: 01/07/2024.

[4] Geoff Huston and Randy Bush. Securing bgp with bgpsec. In *The Internet Protocol Forum*, volume 14, 2011.

[5] Yakov Rekhter, Susan Hares, and Tony Li. A Border Gateway Protocol 4 (BGP-4). Request for Comments RFC 4271, Internet Engineering Task Force, January 2006. Num Pages: 104.

[6] Ravi Malhotra. *IP routing*. " O'Reilly Media, Inc.", 2002.

[7] Sana L. Murphy. BGP Security Vulnerabilities Analysis. Request for Comments RFC 4272, Internet Engineering Task Force, January 2006. Num Pages: 22.

[8] Cuicui Wang, Yu Fu, and Lei Xu. Applications and Challenges of Inter-domain Routing Security Technologies. In *2023 IEEE 3rd International Conference on Data Science and Computer Application (ICDSCA)*, pages 153–157, October 2023.

[9] Asya Mitseva, Andriy Panchenko, and Thomas Engel. The state of affairs in BGP security: A survey of attacks and defenses. *Computer Communications*, 124:45–60, June 2018.

[10] Taejoong Chung, Emile Aben, Tim Bruijnzeels, Balakrishnan Chandrasekaran, David Choffnes, Dave Levin, Bruce M. Maggs, Alan Mislove, Roland Van Rijswijk-Deij, John Rula, and Nick Sullivan. RPKI is Coming of Age: A Longitudinal Study of RPKI Deployment and Invalid Route Origins. In *Proceedings of the Internet Measurement Conference*, pages 406–419, Amsterdam Netherlands, October 2019. ACM.

[11] Nils Rodday, Gabi Dreo Rodosek, and Aiko Pras. Exploring the Benefit of Path Plausibility Algorithms in BGP.

[12] Alexander Azimov, Eugene Bogomazov, Randy Bush, Keyur Patel, Job Snijders, and Kotikalapudi Sriram. BGP AS_path Verification Based on Autonomous System Provider Authorization (ASPA) Objects. Internet Draft draft-ietf-sidrops-aspa-verification-17, Internet Engineering Task Force, February 2024. Num Pages: 23.

[13] Matt Lepinski and Kotikalapudi Sriram. BGPsec Protocol Specification. Request for Comments RFC 8205, Internet Engineering Task Force, September 2017. Num Pages: 45.

[14] Vinay K. Sriram and Doug Montgomery. Design and analysis of optimization algorithms to minimize cryptographic processing in BGP security protocols. *Computer Communications*, 106:75–85, July 2017.

[15] Brian Weis, Roque Gagliano, and Keyur Patel. BGPsec Router Certificate Rollover. Request for Comments RFC 8634, Internet Engineering Task Force, August 2019. Num Pages: 11.

[16] Geoff Huston, Mattia Rossi, and Grenville Armitage. Securing BGP — A Literature Survey. *IEEE Communications Surveys & Tutorials*, 13(2):199–222, 2011. Conference Name: IEEE Communications Surveys & Tutorials.

[17] Tatsuya Takemura, Naoto Yanai, Naoki Umeda, Masayuki Okada, Shingo Okamura, and Jason Paul Cruz. APVAS+: A Practical Extension of BGPsec with Low Memory Requirement. In *ICC 2021 - IEEE International Conference on Communications*, pages 1–7, June 2021. ISSN: 1938-1883.

[18] Kotikalapudi Sriram. BGPsec Design Choices and Summary of Supporting Discussions. Request for Comments RFC 8374, Internet Engineering Task Force, April 2018. Num Pages: 50.

[19] Yakov Rekhter, Tony J. Bates, Ravi Chandra, and Dave Katz. Multiprotocol Extensions for BGP-4. Request for Comments RFC 2283, Internet Engineering Task Force, February 1998. Num Pages: 9.

[20] K Sriram. RIB Size Estimation for BGPSEC.

[21] Bahaa Al-Musawi, Philip Branch, and Grenville Armitage. Recurrence behaviour of BGP traffic. In *2017 27th International Telecommunication Networks and Applications Conference (ITNAC)*, pages 1–7, November 2017. ISSN: 2474-154X.