Interdomain Working Group
Internet-Draft
Intended status: Standards Track
Expires: September 24, 2015

S. Litkowski
Orange Business Service
K. Patel
Cisco Systems
J. Haas
Juniper Networks

March 23, 2015

Timestamp support for BGP paths
draft-litkowski-idr-bgp-timestamp-02

Abstract

BGP is more and more used to transport routing information for critical services. Some BGP updates may be critical to be received as fast as possible : for example, in a layer 3 VPN scenario where a dual-attached site is loosing primary connection, the BGP withdraw message should be propagated as fast as possible to restore the service. The same criticity exists for other address-families like multicast VPNs where "join" messages should also be propagated very fast.

Experience of service providers shows that BGP path propagation time may vary depending on network conditions (especially load of BGP speaker on the path) and too long propagation time are affecting customer service.

It is important for service providers to keep track of BGP updates propagation time to monitor quality of service for the customers. It is also important to be able to identify BGP Speakers that are slowing down the propagation.

This document presents a solution to transport timestamps of a BGP path. The solution is targeted to be used using special identified beacon prefixes that are single-homed.

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 [RFC2119].

Status of This Memo

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

This Internet-Draft will expire on September 24, 2015.

Table of Contents

1. Problem statement 3
2. Requirements for monitoring BGP path propagation time 4
2.1. Architecture 4
2.2. Measurement accuracy 6
2.2.1. Clock synchronization 6
2.2.2. Beacon accuracy 6
2.3. Churn 6
2.4. Path propagation complexity 7
3. Proposal 8
4. BGP timestamp attribute 9
5. Processing the BGP timestamp attribute 10
5.1. Inspection list 10
5.2. Originating a timestamped route in BGP 11
5.3. Receiving a timestamped route in BGP 11
5.4. Sending a timestamped route in BGP 13
5.4.1. Propagating the BGP Timestamp attribute 13
5.4.2. Setting the send timestamp 13
5.5. Limiting churn 14
5.6. Marking stale entries 15

5.7. Inter-AS considerations 19
5.7.1. Drop option 19
5.7.2. Drop AS option 20
5.7.3. Summary option 21
5.7.4. Propagate option 22
5.8. Retrieving timestamp vector 23
5.9. Handling malformed attribute 23
5.10. Impact on update packing 23
6. Compared to BMP 23
7. Deployment considerations 24
8. Security considerations 25
9. Acknowledgements 25
10. IANA Considerations 25
11. Normative References 25
Authors' Addresses 26

1. Problem statement

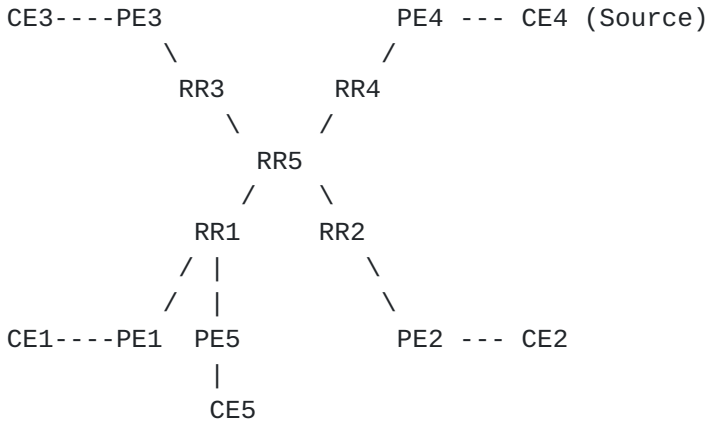


Figure 1

The figure 1 describes a typical hierarchical RR design where PEs are meshed to local RRs and local RRs are meshed to more centric RRs. We consider a single multicast VPN between all CEs. CE4 is the source, all others may be receivers. The BGP controlplane also supports some other BGP service like L3VPN service.

We consider an event in L3VPN service leading to RR1 being temporarily overloaded (for example, RR1 is processing massive updates due to a router failure or formatting updates for a route-refresh). In the same timeframe, CE1 wants to join the multicast flow from CE4. PE1 propagates the C-multicast route to RR1, but RR1 fails to propagate the route to RR5 because it is busy processing L3VPN. When RR1 finishes the L3VPN job, it would send the C-multicast route to RR5 and updates would be imported by PE4. The

long time to join the flow may cause CE4 to miss part of the multicast flow.

All BGP implementations are different in term of internal processing within an address family or between address family. The issue described above is just given as an example, and the document does not presume that all implementations are suffering from this exact issue. But whatever the implementation, their always be cases where BGP path propagation could be delayed.

Service providers currently lack of efficient solution to keep track of BGP path propagation time as well as solution to identify the BGP speakers causing issues.

BMP (BGP Monitoring Protocol) may be a solution but as several drawbacks (see Section 6).

2. Requirements for monitoring BGP path propagation time

2.1. Architecture

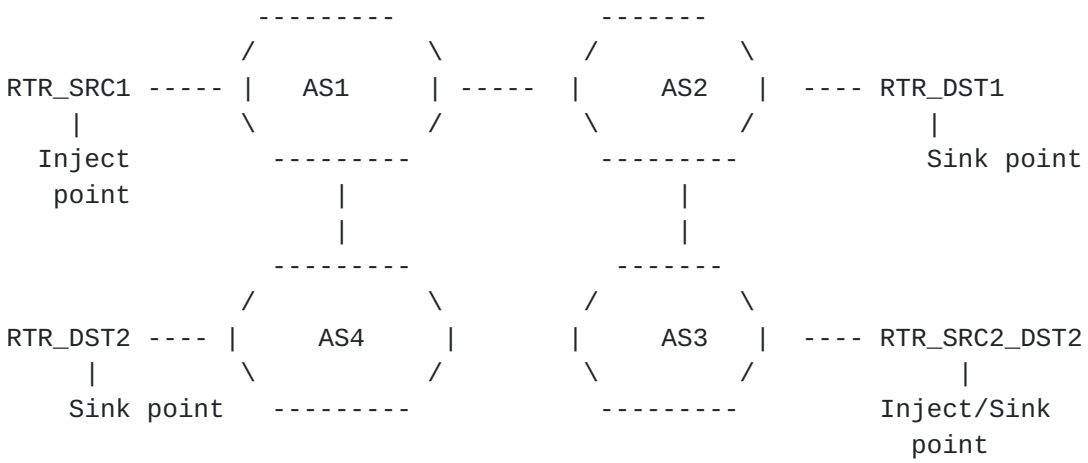


Figure 2

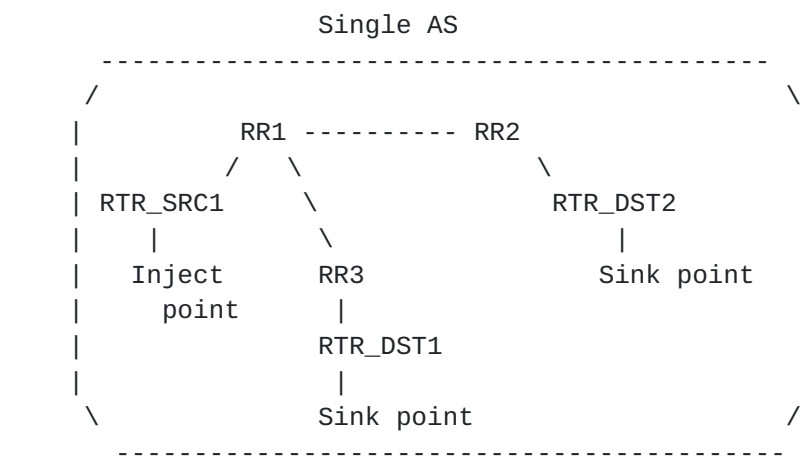


Figure 3

Figure 2 and Figure 3 describes an interAS and a single AS scenario where a service provider wants to monitor BGP path propagation time from a router to multiple routers. In Figure 2, multiple probing routers are attached to multiple ASes. In Figure 3, all probing routers are in the same AS.

The architecture requires some BGP Speaker to originate some NLRI within the BGP controlplane. In the diagram above, they are identified as "Inject point". In order to provide information about propagation delays, the architecture requires introduction of timestamp information. Architecture also needs to identify BGP Speaker causing high propagation delays. As only, specific advertisement will serve for measurement, the architecture requires BGP Speaker to identify NLRIs that must be timestamped. The architecture also requires some BGP Speaker to serve as sink point where a timestamp vector information can be retrieved. The timestamp vector must contain propagation time information for all BGP Speaker that participated in the BGP path. It is so required that each BGP Speaker along the path to add timestamp information. There may be multiple sink points in the network to perform measurement at different location and also different inject points. An external tool may be connected to Sink Points to retrieve the timestamp information. But this is out of scope of the document.

In case of interAS, for security reason, the architecture MUST support hiding detailed timestamp information to the other AS.

Example of usage :

An external tool should command RTR_SRC to originate a probing BGP NLRI. All the BGP Speakers are configured to measure timestamp for this NLRI. The BGP path would propagate across BGP Speakers. Each BGP Speaker may provide timestamp informations. An external tool

connected to sink points will retrieve timestamp vector information for the NLRI.

2.2. Measurement accuracy

2.2.1. Clock synchronization

For the solution to be accurate, it is mandatory for BGP Speaker to be synchronized. This could be ensured easily within a single AS but in a inter domain scenario, it is hard to ensure that all Speakers are synchronized to a good clock source.

The solution MUST include synchronization information associated with the timestamp in order to be able to compare timestamps between them.

2.2.2. Beacon accuracy

In order to be accurate, an implementation SHOULD :

- ensure that the timestamped NLRIs are processed with the same priority as non timestamped NLRIs.

- ensure that the processing of adding timestamp information is as lightweight as possible. If some limitation exists, the vendor SHOULD document them.

Using a unique special prefix advertisement from a single location to evaluate propagation time will not provide a detail view of min/max propagation time values as the user will not know where the path for the prefix may be located in a processing queue. Considering a BGP Speaker handling high churn, the advertisement of the path for the special prefix may have a specific place in the long processing queue of the churn depending on the implementation : it may be first, last or somewhere in the middle.

It is required from user to perform sampling to establish propagation time boundaries based on multiple advertisements. Repeated operations of advertisement then withdraw may help in this. See Section 7 for more details.

2.3. Churn

The target solution MUST NOT create more churn in the BGP controlplane.

2.4. Path propagation complexity

When a NLRI is originated in BGP from a point, a BGP path is created. Nothing ensures that all nodes within the BGP controlplane will receive this BGP path. When a concurrent path already exists from the NLRI, the concurrent path may be prefered by some BGP Speaker leading to hiding of the new path. Moreover, even if the NLRI is originated in BGP from a single point, multiple paths may be created within the BGP controlplane, this is inherent to the BGP meshing in place.

As soon as multiple BGP paths are involved, controlplane convergence may be done in multiple steps in order to find the final best path. This convergence may involve multiple BGP path advertisement (replacing each other) between peers.

The goal of our proposal is not to measure the convergence time but to focus on the path propagation time. In a controlplane convergence involving multiple paths for a NLRI, the solution MUST identify timestamp for the event where the NLRI was seen for the first time on a BGP Speaker.

Example :

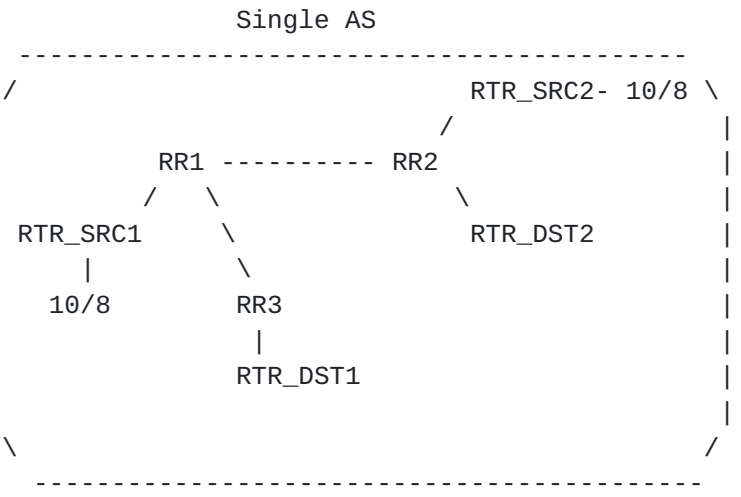


Figure 4

In the figure above, consider that the service provider is keep tracking of propagation time for real NLRIs (corresponding to customer routes). All the BGP Speakers in our figure are configured to inspect the NLRI 10/8 which is multihomed. We consider that the network is starting and the NLRI has not been propagated yet.

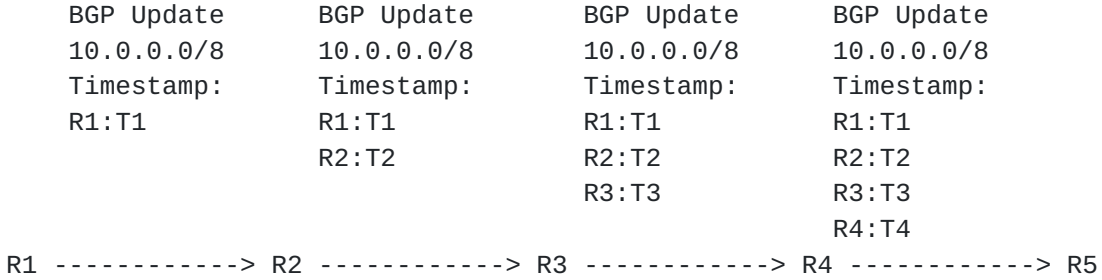
RTR_SRC1 starts to propagate 10/8 within the BGP controlplane. All BGP Speakers considers the path as best and this path will be propagated within the whole controlplane. Each BGP Speaker would add its timestamp information and RTR_DST1 and RTR_DST2 would be able to record the timestamp vector. In this case, the timestamp vector is quite accurate because it represents an end to end propagation.

Now RTR_SRC2 starts to propagate its own path. RR2 has two paths for 10/8 and will choose the best one, let's consider that RTR_SRC2 path is the best one, RTR_SRC2 path will so be propagated and timestamp vector will be updated. RR1 will also have two paths, and we consider that RR1 prefers RTR_SRC1 path, so RTR_SRC2 path will not be propagated by RR1. In this situation, RTR_DST2 will receive the path from RR2 with accurate timestamp (end to end propagation) but RTR_DST1 will never receive it.

We could also consider a stable network situation, where both paths have been advertised for a long time. A network event may occur (e.g. IGP metric change) that would cause a BGP Speaker within a path vector to change its best path. In Figure 10, an IGP event, may cause RR1 to change its decision and prefers the path originated by RTR_SRC2 as best, the path will be propagated with previous received timestamp information that are no more accurate. RTR_DST1 will receive a BGP timestamp vector containing stale (old) timestamp informations as well as new ones.

3. Proposal

Our proposal is based on tagging NLRI with timestamp values along its BGP path propagation. Each BGP Speaker along the path will add timestamp values, so creating a timestamp vector. An ordered list of timestamps would so be built along the path.



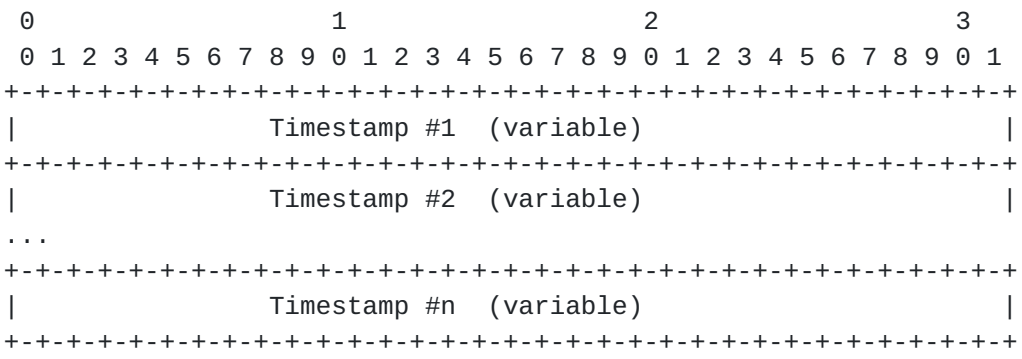
Using this mechanism, we can easily identify if a hop within a path is slowing down the propagation.

We propose to use a new BGP attribute, BGP timestamp attribute to encode timestamps information.

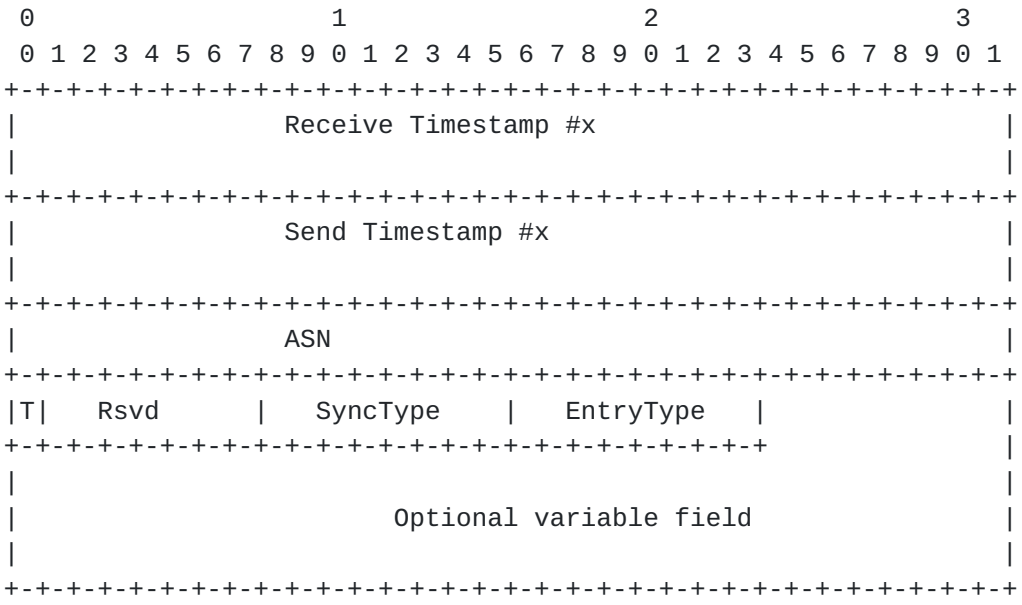
4. BGP timestamp attribute

The BGP timestamp (BGP-TS) Attribute is an optional transitive BGP Path Attribute. The attribute type code is TBD.

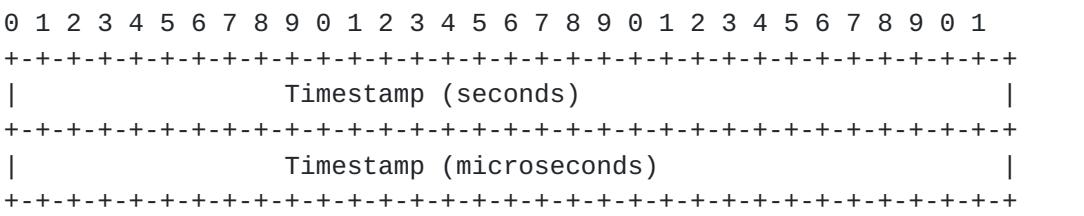
The value field of the BGP timestamp attribute is defined as an ordered list of timestamp entries, the first entry being the first timestamp entry added (origin):



The timestamps entries are encoded as follows :



o Receive timestamp : the time at which the BGP path was received. When originating a path in BGP, the timestamp is the originating time. Expressed in seconds and microseconds since midnight (zero hour), January 1, 1970 (UTC). If zero, the time is unavailable. Precision of the timestamp is implementation- dependent.



- Send timestamp : the time at which the BGP path was exported to the peer. Expressed in seconds and microseconds since midnight (zero hour), January 1, 1970 (UTC). If zero, the time is unavailable. Precision of the timestamp is implementation-dependent.

- ASN : AS Number of the local node creating the timestamp entry.

- Flags :

  * T : Synchronized, if set, the BGP speaker clock is synchronized to an external system.

- SyncType : defines the stratum as defined in [RFC5905].

- EntryType : defines the type of Timestamp entry, the following types are defined :

  * Type 0 : empty. There is no following variable field. This type is to be used in case of timestamp summarization.

  * Type 1 : IPv4 address, the following variable field will be 4 bytes long and will contain the IPv4 router ID of the local node.

  * Type 2 : IPv6 address, the following variable field will be 16 bytes long and will contain the IPv6 router ID of the local node.

  * Type 3 : Stale Indicator, Stale indicates that previous timestamp entries are old. There is no following variable field. The receive timestamp and send timestamp should be set to zero. The ASN is set to the ASN of the local BGP Speaker.

5. Processing the BGP timestamp attribute

5.1. Inspection list

A BGP Speaker supporting the BGP-TS can decide to timestamp only some specific NLRIs. An inspection list may be configured by the user (filter) to apply timestamping on a specific set of BGP NLRIs. By

default, we suggest that a BGP Speaker supporting BGP-TS SHOULD NOT timestamp any BGP NLRIs.

User of our proposal must be aware that using a complex policy to express inspection list may result in more processing that will influence the end to end propagation time. It is expected that the inspection list policy should be kept as simple as possible.

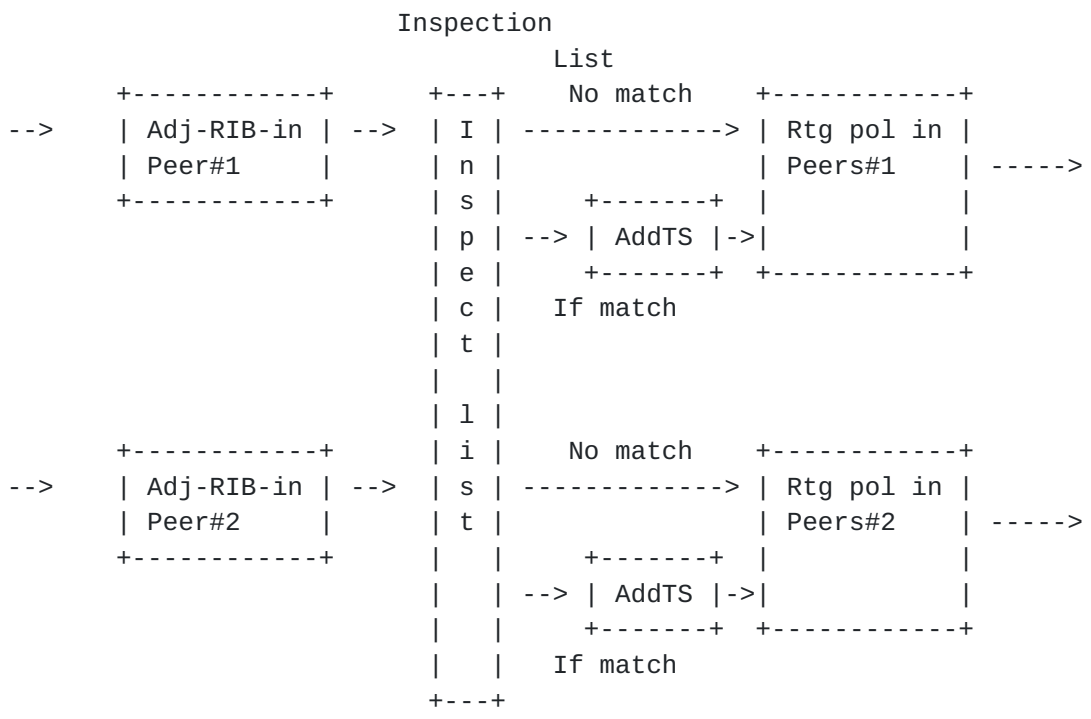
5.2. Originating a timestamped route in BGP

When a BGP Speaker supporting BGP-TS originates a new path in BGP that matches the inspection list, it MUST add the BGP-TS attribute to the BGP path and MUST set the receive timestamp field to the time the path was originated in BGP. At this time of processing, the send timestamp will be set to 0. If the BGP Speaker is synchronized to an external system when originating the route, the S-bit MUST be set in the attribute and the SyncType MUST be set to the current stratum. As mentioned above, the BGP path of the originated route will have a send timestamp value of zero in the BGP LOC-RIB.

5.3. Receiving a timestamped route in BGP

When a BGP Speaker supporting BGP-TS receives a BGP path that matches the inspection list, the implementation MUST record the current time associated with the received path.

The time recording MUST append before the inbound routing policies.



If the path that matches the inspection list and does not contains a BGP-TS attribute, it MUST add a BGP-TS attribute with a timestamp entry :

- The receive timestamp MUST be set to the recorded time for this BGP path.
- If the BGP Speaker is synchronized to an external system when receiving the route, the S-bit MUST be set in the attribute and the SyncType MUST be set to the current stratum.
- The send timestamp MUST be set to zero.

If the path that matches the inspection list and contains a BGP-TS attribute, it MUST append a new timestamp entry in the existing attribute :

- The receive timestamp MUST be set to the recorded time for this BGP path.

- If the BGP Speaker is synchronized to an external system when receiving the route, the S-bit MUST be set in the attribute and the SyncType MUST be set to the current stratum.

- The send timestamp MUST be set to zero.

The process of adding a timestamp entry or adding BGP-TS attribute SHOULD be as light as possible in order to influence the propagation time as lowest as possible.

When a BGP Speaker supporting BGP-TS receives a BGP path that does not the inspection list and contains a BGP-TS attribute, it MUST NOT change the existing attribute.

When a BGP Speaker not supporting BGP-TS receives a BGP path that contains a BGP-TS attribute, it MUST follow the standard BGP procedures described in [RFC4271].

5.4. Sending a timestamped route in BGP

5.4.1. Propagating the BGP Timestamp attribute

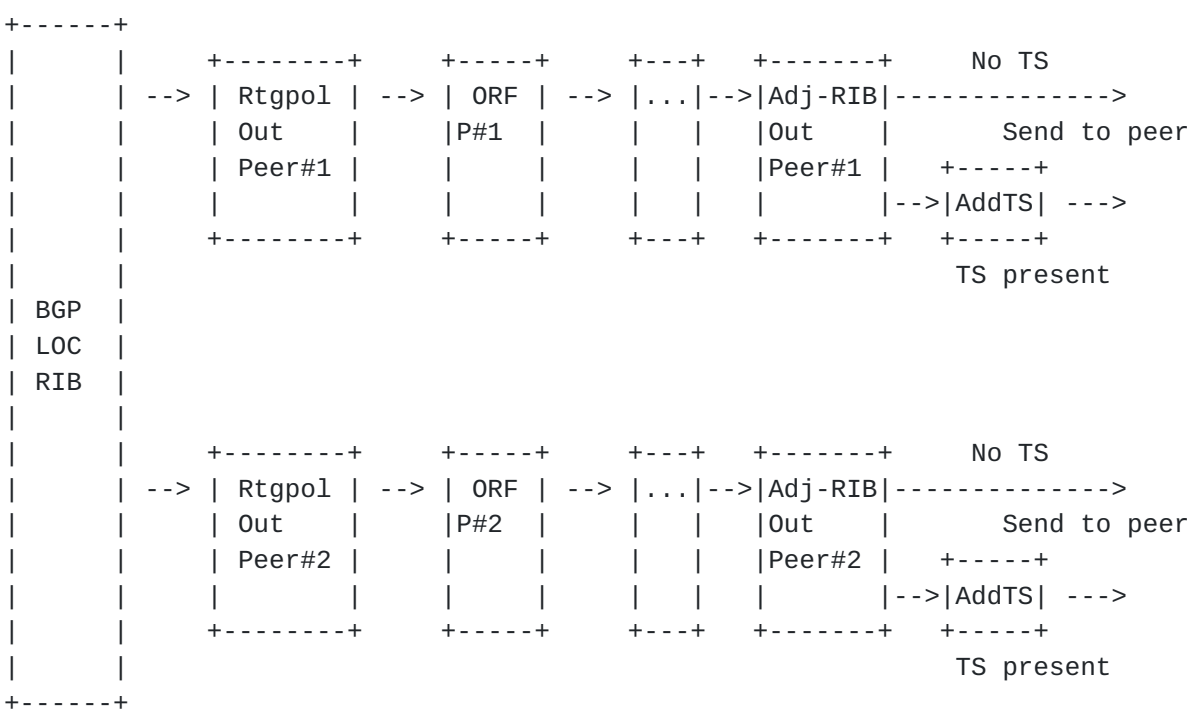
For a manageability/security purpose, the authors suggest that BGP timestamp attribute MAY NOT be sent to a peer unless it was explicitly configured for. This would prevent timestamp and internal address informations to be propagated to some external peers for example. See Section 5.7 for more information.

If a BGP path containing a BGP-TS attribute must be sent to be peer not configured with BGP timestamp option, the BGP-TS attribute should be dropped when the update message is sent to the peer.

5.4.2. Setting the send timestamp

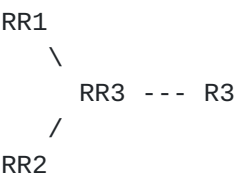
If sending timestamp attribute is authorized for a specific peer, and path has a BGP-TS attribute, the outgoing BGP processing MUST fill the send timestamp field when exporting the path to a peer. The time recording MUST occur after all BGP filtering policies (outgoing routing policies, ORF, ...) and after placing path in Adj-RIB-Out. An implementation SHOULD set timestamp at the nearest possible step before sending the BGP Update to the peer. Depending of the implementation, the timestamping may occur at different stage of the outgoing BGP processing. Each implementer SHOULD document their timestamping process in order to make users understand correctly timestamp values. As most of implementations are using the concept of peer-groups, in case, timestamp is set too early in the BGP outgoing processing, all peers within a group may have the same timestamp value. Implementation should avoid this.

The process of adding the send timestamp must be as light as possible in order to influence the propagation time as lowest as possible.



5.5. Limiting churn

Adding timestamp informations to BGP path will make all received paths to be unique.



In the figure above, we consider that RR1 and RR2 are part of the same cluster (cluster ID : 1). RR3 is client of RR1 and RR2. R3 is client from RR3, R1 is client from RR1 and RR2.

Without BGP timestamp, when R1 originates the BGP prefix 10/8, it sends it to RR1 and RR2. Consider that RR3 receives path from RR1 first, it will reflect it to R3. When it will receive the path from RR2, it may consider that path from RR2 is best (lowest router ID) but as BGP attributes of the path are exactly the same as for RR1 path, there is no need to send an update to R3.

With BGP timestamp, when R1 originates the BGP prefix 10/8, it sends it to RR1 and RR2. Consider that RR3 receives path from RR1 first,

it will reflect it to R3. When it will receive the path from RR2, it may consider that path from RR2 is best (lowest router ID) but as BGP attributes of the two paths are not more equal due to the timestamp difference, RR3 may need to advertise an update to R3.

In order to prevent introducing more churn, we propose to modify the behavior described in Section 9.2. of [RFC4271]. An implementation MUST NOT consider BGP-TS attribute when evaluating the need to send a new update. As the BGP-TS attribute is purely informational, even if BGP Speakers have a different view of the timestamp attribute, there will be no impact on routing.

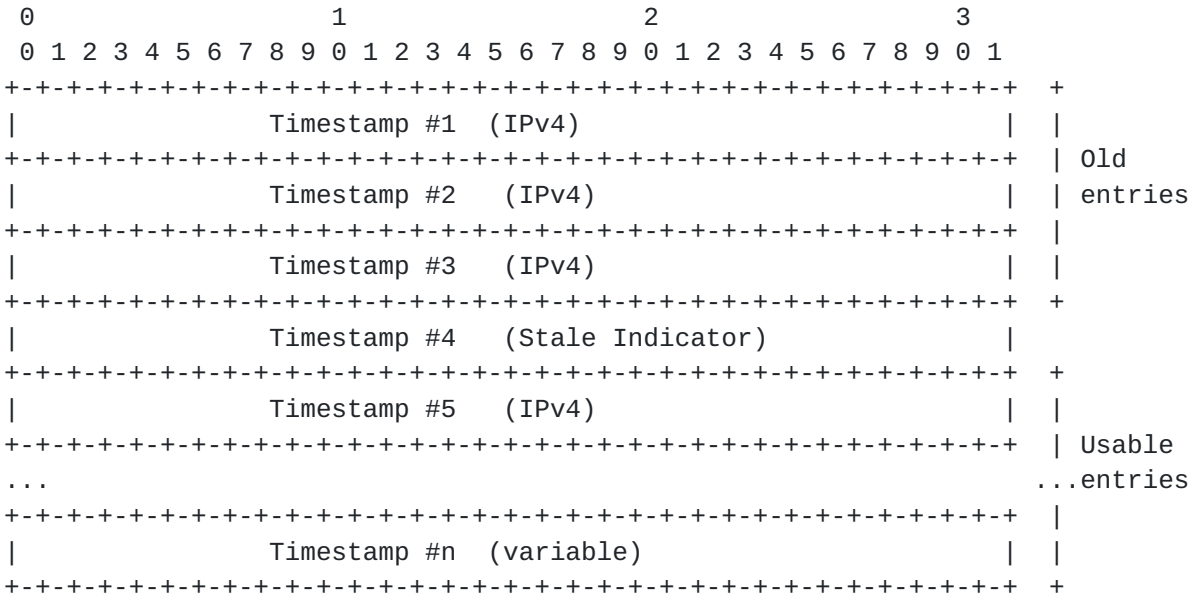
Considering our example, when RR3 will receive the path from RR2, even if it considers RR2 path as best, it will not send an update to R3 as all the attributes, except BGP-TS are equal.

5.6. Marking stale entries

Section 2.4 describes some cases where advertised timestamp information is no more relevant because it is old and also requires identification of first propagation timestamps.

In order to do this, we propose to mark old entries by adding a Stale Indicator within the timestamp vector. The presence of Stale Indicator must be interpreted as all previous timestamp entries need to be considered as old and not considered as a first propagation.

BGP-TS attribute example :

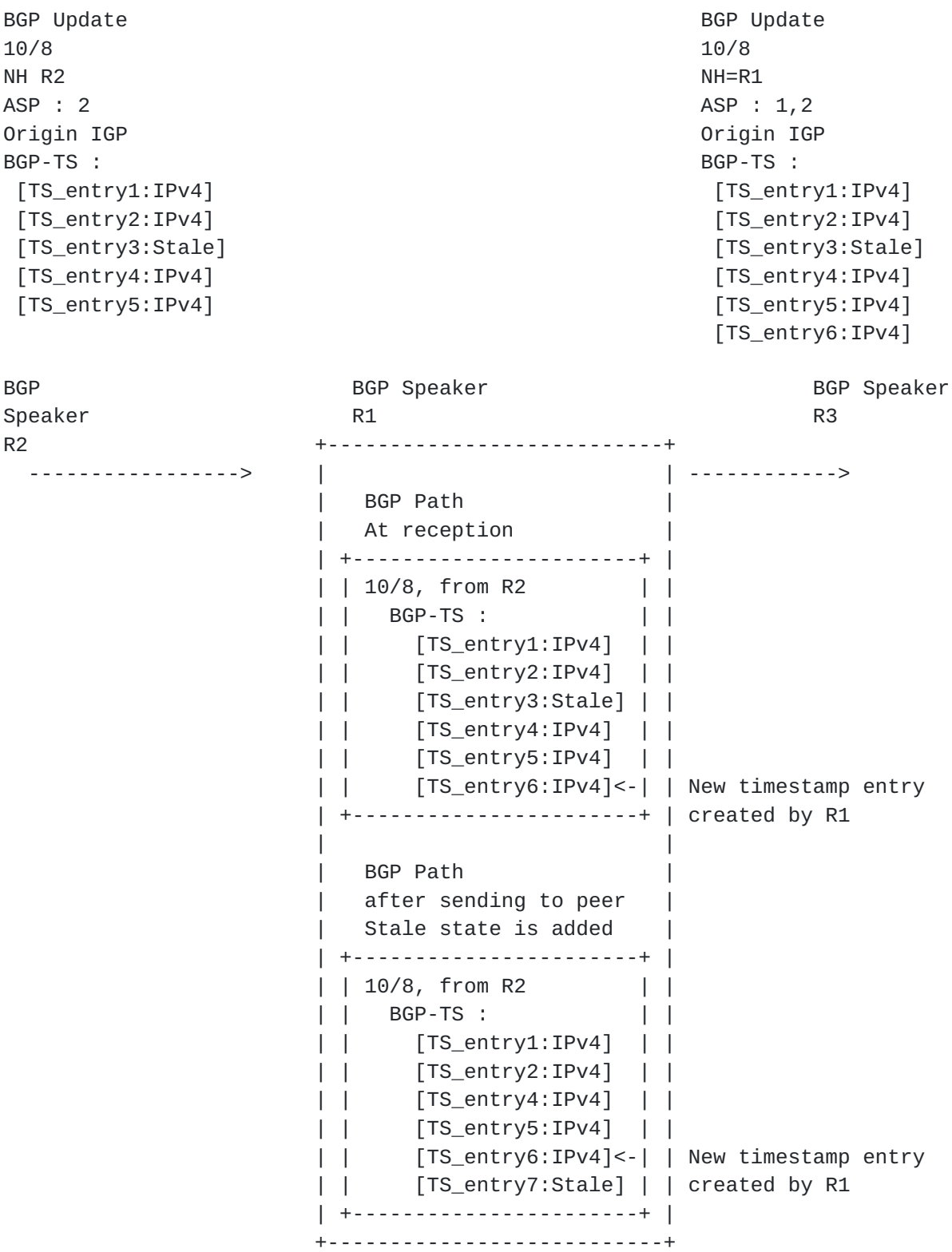


Insertion of Stale Indicator in a BGP-TS attribute may happen in the following conditions :

- A path is received from a peer containing BGP-TS attribute or originated locally, the path matches the inspection list, and the decision process does not select the path as best path. Then the Stale Indicator SHOULD be inserted after decision process happened.

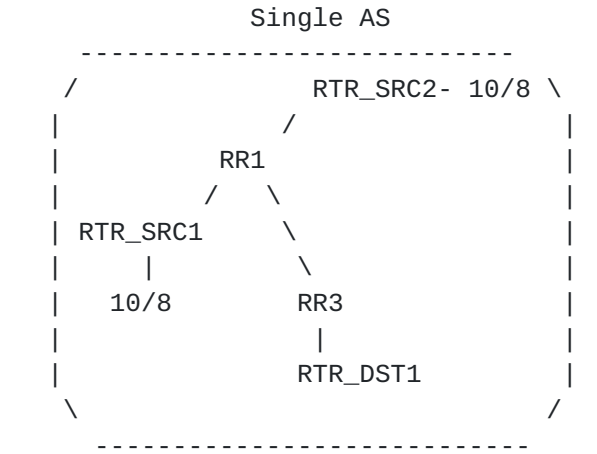
- A path is received from a peer containing BGP-TS attribute or originated locally, the path matches the inspection list, and the decision process does select the path as best path. The path is exported to peers and then the Stale Indicator MUST be inserted. The path MUST NOT be repropagated as per Section 5.5.

When inserting a Stale indicator, if a Stale Indicator already exists in the timestamp vector, the implement SHOULD remove it before adding the new one.



In the example above, R2 sends a BGP path with some existing stale timestamps. When R1 receives the route, it creates a new timestamp

entry in the BGP-TS attribute. We consider that the decision process decides that the path is best, the path is exported with the new timestamp entry and old timestamps coming from R2. Then R1 will update its local path by removing the previous Stale Indicator and replace a new one at the latest position to mark that it is no more the first propagation.



In the figure above, we consider that all BGP Speaker apply timestamp for prefix 10/8. RTR_SRC1 originates 10/8 in BGP, the decision process will decide that the path is best. RTR_SRC1 will export path to RR1 and then it will add locally the Stale Indicator within the timestamp vector. The path exported does not have the Stale Indicator. RR1 will receive the path and add a timestamp entry, the path is considered as best, RR1 will export it to RTR_SRC2 and RR3 and then it will add a stale indicator. RR3 will proceed in the same way.

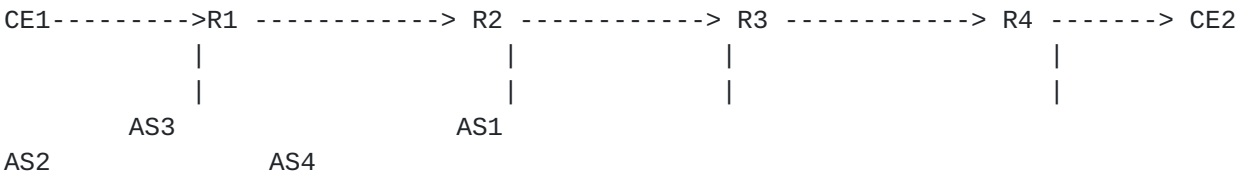
When RTR_SRC2 will originate a new path for 10/8, if this new path is best on RTR_SRC2, it will export the path to RR1 and then it will add locally the Stale Indicator to the path. When RR1 will receive the route :

- If the path from RTR_SRC2 is best, RR1 will export the new path to RTR_SRC1 and RR3 and then will add Stale indicator to the path.If RTR_SRC2 fails after some time, RR1 will pick up RTR_SRC1 path as best, and will export it to RR3. RR3 will know that the received timestamp entries are stale thanks to the stale indicator.

- If the path from RTR_SRC2 is not best, RR1 will add Stale indicator to the path. If RTR_SRC1 fails after some time, RR1 will pick up RTR_SRC2 path as best, and will export it to RR3. RR3 will know that the received timestamp entries are stale thanks to the stale indicator.

5.7. Inter-AS considerations

```
   BGP update
   10.0.0.0/8
   TS:
    AS3;CE1:rT1,sT2
```



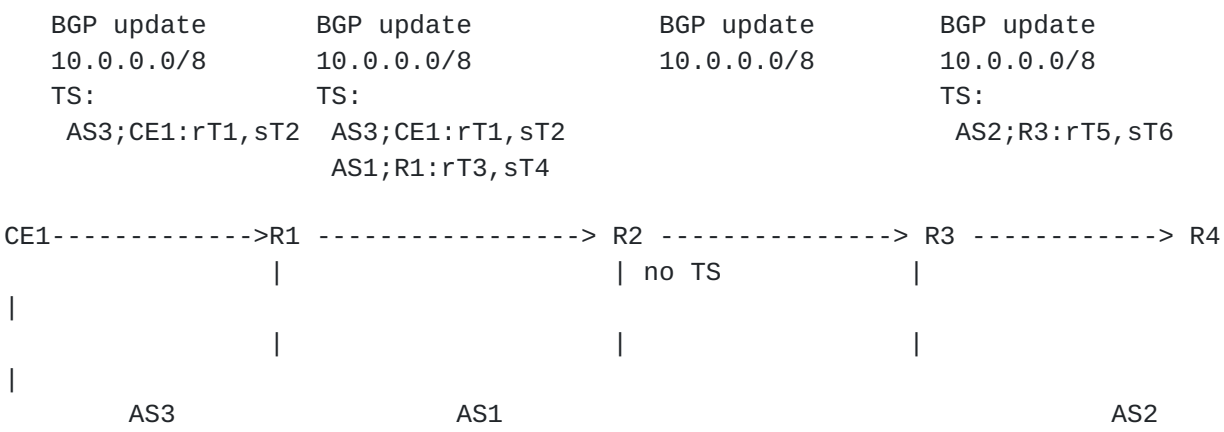
In the figure above, we consider that customer wants to monitor BGP updates propagation time between its two sites.

If AS1 and AS2 BGP Speakers does not support BGP-TS, the attribute will be transported transparently accross AS1 without any processing. CE2 will so receive the BGP path with only a single timestamp entry from CE1.

If AS1 and AS2 BGP Speakers does support BGP-TS, four different options are offered : drop, drop-as, summarize, propagate. It must be noted that using drop-as or summarize options may involve more processing and so may impact the end to end propagation time.

5.7.1. Drop option

If AS1 and/or AS2 BGP Speakers support BGP-TS, they may not want to expose any timestamp information between each other. If a service does not want to propagate timestamp information to external peers, it can decide to not activate the "timestamp" option on the peer configuration , as explained in Section 5.4.

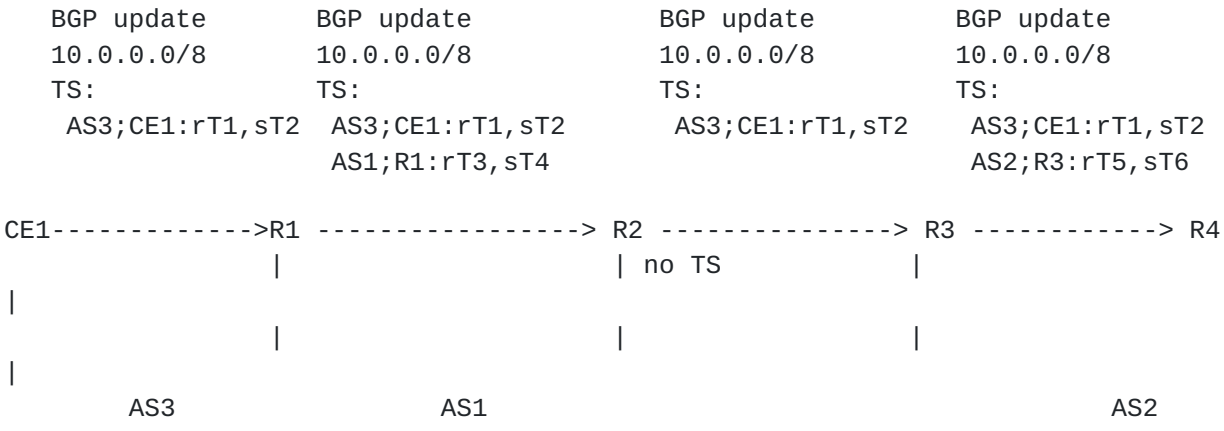


In the example above, CE1 is configured to send timestamp to R1, as well as R1 to R2. But R2 does not want to send timestamp to R3.

When sending BGP route for 10/8, CE1 adds timestamp attribute and a timestamp entry (AS3, entry type : IPv4=CE1_IP, receive timestamp = T1, send timestamp=T2). R1 receives the path, we suppose that the inspection list matches, so R1 adds a timestamp entry. When sending to R2, R1 will send the following information in its timestamp entry : AS1,entry type : IPv4=R1_IP, receive timestamp T3, send timestamp T4. As R2 is configured to not send timestamp information to R3, it will drop the BGP attribute when sending to R3.

5.7.2. Drop AS option

If AS1 and/or AS2 BGP Speakers support BGP-TS, they may not want to expose their timestamps or internal BGP topology to other ASes. If a service does not want to propagate local AS related timestamp information to external peers, it can decide to use the "drop-as" option towards the peer.

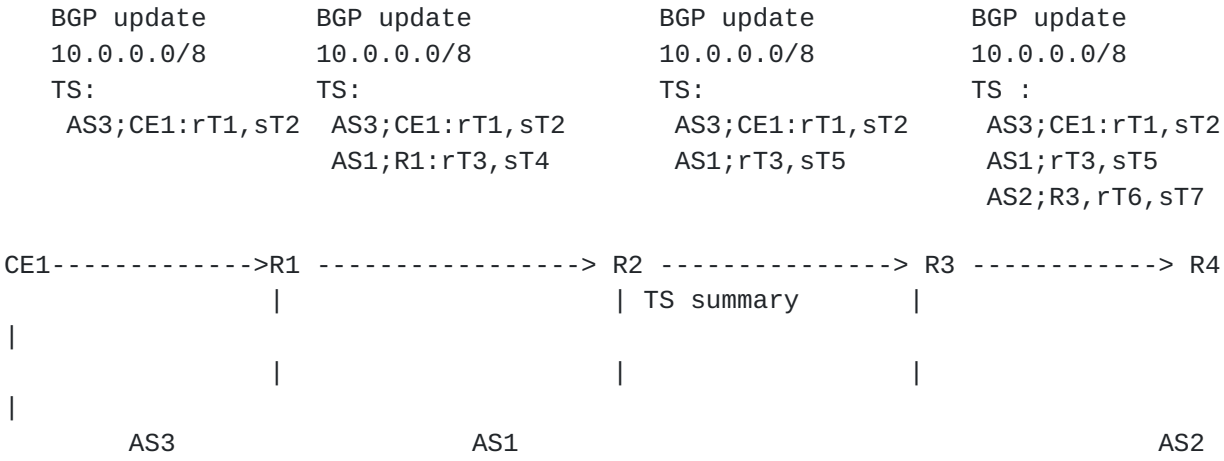


In the example above, CE1 is configured to send timestamp to R1, as well as R1 to R2. But R2 does not want to send AS1 internal timestamp to R3. "Drop-as" option is configured on R2 towards R3.

When sending BGP route for 10/8, CE1 adds timestamp attribute and a timestamp entry (AS3, entry type : IPv4=CE1_IP, receive timestamp = T1, send timestamp=T2). R1 receives the path, we suppose that the inspection list matches, so R1 adds a timestamp entry. When sending to R2, R1 will send the following information in its timestamp entry : AS1,entry type : IPv4=R1_IP, receive timestamp T3, send timestamp T4. As R2 is configured with "drop-as" option to R3, it will remove all timestamp entries where the ASN is equal to its autonomous system number and then send the update to R3.

5.7.3. Summary option

If AS1 and/or AS2 BGP Speakers support BGP-TS, they may want to offer timestamp service to their customers but they want to hide their internal topology. In order to achieve the expected behavior, AS1/ AS2 can activate a timestamp summary option on the external peer.



When using summary option, the BGP-TS attribute is modified as follows when exporting the route :

- All timestamp entries containing the local AS in AS field are removed.
- A new timestamp entry is created and inserted in place of removed entries (n entries replaced by 1).
- The new timestamp entry will use an entry type zero.

- The new timestamp entry MUST have the S bit set.

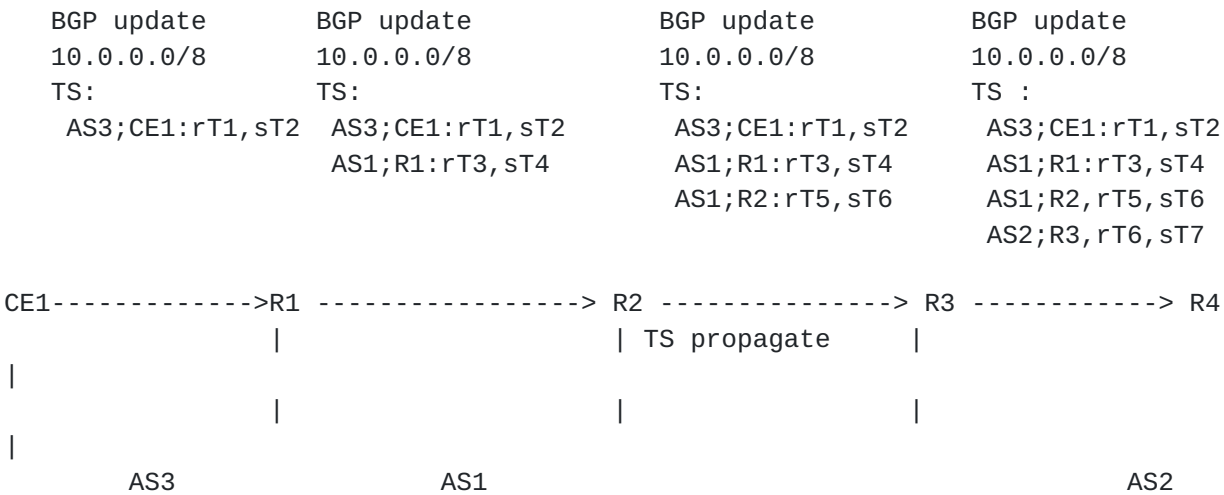
- The new timestamp entry MUST NOT have any EntryType.
- The receive timestamp of the new timestamp entry is the receiving timestamp of the first timestamp entry that has been removed.
- The send timestamp of the new timestamp entry will be added as usual.

In the example above, CE1 is configured to send timestamp to R1, as well as R1 to R2. But R2 wants summarize timestamp information to AS2.

When sending BGP route for 10/8, CE1 adds timestamp attribute and a timestamp entry (AS3, entry type : IPv4=CE1_IP, receive timestamp = T1, send timestamp=T2). R1 receives the path, we suppose that the inspection list matches, so R1 adds a timestamp entry. When sending to R2, R1 will send the following information in its timestamp entry : AS1,entry type : IPv4=R1_IP, receive timestamp T3, send timestamp T4. As R2 is configured with "summarize" option to R3, it will remove all timestamp entries where the ASN is equal to its autonomous system number and add a new timestamp entry with an entry type zero. The receive timestamp will be retrieved from R1 timestamp entry.

5.7.4. Propagate option

If AS1 and/or AS2 BGP Speakers support BGP-TS, they may want to offer timestamp service to their customers with a full view. This MUST be the default behavior when timestamp is activated on a peer.



5.8. Retrieving timestamp vector

Authors suggest to implementers to use a local wrapping buffer on each node and record entries in the buffer each time a BGP path is timestamped. An external tool should then retrieve timestamps information from sink points. How the information is retrieved is out of scope of the document but we can imagine using :

- BMP from the external tool to the sink point.
- NetConf get to retrieve wrapping buffer information.
- SNMP get to retrieve wrapping buffer information.
- CLI command to retrieve wrapping buffer information.

5.9. Handling malformed attribute

When receiving a BGP Update message containing a malformed BGP-TS attribute, an "attribute discard" action MUST be applied as defined in [I-D.ietf-idr-error-handling].

5.10. Impact on update packing

Introducing timestamps information will make update packing less efficient for the timestamps path. In the deployment we are targeting (Section 7), this is not considered as an issue. In the case where a site is generating a special prefix with path timestamped and others not timestamped, these prefixes will not be packed together, so two update messages will be generated. Even if two updates are generated, we do not consider, that the propagation time will be highly affected.

6. Compared to BMP

BMP (BGP Monitoring Protocol) [I-D.ietf-grow-bmp] is a solution to monitor BGP sessions and provides a convenient interface for obtaining route views. BMP is a complete suite of messages to exchange informations regarding a BGP session.

We can imagine to use BMP as a solution to monitor BGP update propagation time but there is multiple drawbacks associated with such solution :

- BMP provides dump of all received BGP update (per peer). If we are interested only in probing BGP routes, a strong filtering of information may be needed in BMP messages.

- BMP does not mandate timestamping of messages (as per [I-D.ietf-grow-bmp] Section 5) : "If the implementation is able to provide information about when routes were received, it MAY provide such information in the BMP timestamp field. Otherwise, the BMP timestamp field MUST be set to zero, indicating that time is not available."

- BMP may provide (if implementation available) timestamps information only for a single router point of view. If we want to retrieve timestamps of all BGP Speakers on a path, a BMP session is required to all BGP speakers. Correlation (based on known design) is also required at the external tool to order timestamps from each BMP session.

- If BMP provides timestamp information, it does not provide information on how the router clock is synchronized (free run, NTP, GPS ...).

- BMP only provides Adj-RIB-in view and does not provide outgoing information.

Using BMP to monitor BGP update propagation may complexify the design of the monitor solution. But as mentioned in Section 1, BMP can be used on specific sink routers to retrieve BGP TS vector.

7. Deployment considerations

This solution is not intended to perform timestamp imposition on all BGP prefixes.

The deployment scenario we are targeting is really to monitor some specific single-homed NLRIs identified by the service provider (see Section 2 as an example).

These NLRIs may be advertised at some injection point in the network, and timestamp vector will be retrieved at some sink points. As pointed in Section 2.2.2 , multiple samples of measurement will be necessary in order to evaluate the propagation time.

These NLRIs should be single-homed in order to ensure an end to end propagation from injection point to sink point. A coordination between injection and sink points based on an external tool is necessary : once a NLRI to be monitored has been advertised, the tool would retrieve the timestamp vector from the sink point.

Service provider may use real prefixes (used for routing) or special prefixes (standard IP prefix but allocated for beaconing). In case of special prefix used, the tool can at regular interval command the

advertisement and withdrawal of the prefix. The tool must ensure that it has retrieved the timestamp vector before withdrawing the prefix and also wait for convergence after withdrawal before advertising back the prefix.

The inspection list should be kept as small as possible by users in order to not introduce processing overhead and as a consequence slow down propagation.

8. Security considerations

Depending of the implementation and router capacity, adding timestamps to BGP path may consume some router resources. As proposed in Section 5.1, by default a BGP Speaker will not timestamp any path and inspection list should be configured to activate timestamping on a subset of paths. Using this approach, we consider that overhead that may be introduced by timestamping BGP paths is well controlled by operators. An external router cannot force an internal router to timestamp.

Providing detailed timestamps information to other ASes may introduce security issues by exposing internal datas (part of BGP topology, IP addresses, internal performance) to external entities. The proposal we make in Section 5.7 solves this security issue by giving flexibility to operators on the level of information he wants to expose to external peers.

9. Acknowledgements

10. IANA Considerations

IANA shall assign a codepoint for the BGP Timestamp attribute. This codepoint will come from the "BGP Path Attributes" registry.

11. Normative References

[I-D.ietf-grow-bmp]
Scudder, J., Fernando, R., and S. Stuart, "BGP Monitoring Protocol", draft-ietf-grow-bmp-07 (work in progress), October 2012.

[I-D.ietf-idr-error-handling]
Chen, E., Scudder, J., Mohapatra, P., and K. Patel, "Revised Error Handling for BGP UPDATE Messages", draft-ietf-idr-error-handling-18 (work in progress), December 2014.

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.

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

[RFC5905] Mills, D., Martin, J., Burbank, J., and W. Kasch, "Network Time Protocol Version 4: Protocol and Algorithms Specification", RFC 5905, June 2010.

Authors' Addresses

Stephane Litkowski
Orange Business Service

Email: stephane.litkowski@orange.com

Keyur Patel
Cisco Systems

Email: keyupate@cisco.com

Jeff Haas
Juniper Networks

Email: jhaas@juniper.net