

Version de table BGP

Contenu

[Introduction](#)

[Diagramme du réseau](#)

[Le meilleur chemin](#)

[Types de versions de table](#)

[Nombre de version de table initial](#)

[Conditions pour un changement de version de table BGP](#)

[Utilisation de version de table](#)

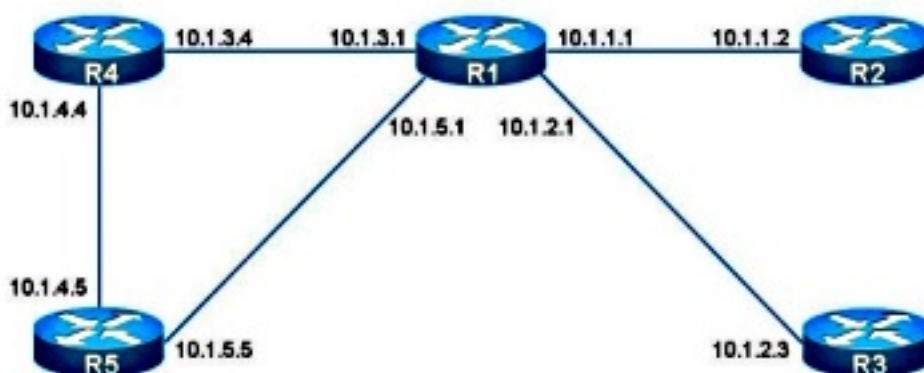
[Utilisation pour le dépannage](#)

Introduction

Ce document décrit la version de table, qui est un nombre utilisé par le Protocole BGP (Border Gateway Protocol) afin de dépister que les meilleures modifications de chemin des préfixes BGP sont propagé auxquels le BGP scrute. C'est un nombre utilisé par le logiciel BGP. Vous pouvez visualiser le nombre de version de table si vous sélectionnez des commandes show, qui aide l'administrateur réseau à dépanner des problèmes.

[Diagramme du réseau](#)

C'est le schéma de réseau qui est utilisé pour cet article :



Le meilleur chemin

Un préfixe BGP a un ou plusieurs chemins, parce que le préfixe BGP est appris de différents pairs et de sources BGP.

Voici un exemple d'un préfixe BGP avec des plusieurs chemins. Il y a deux chemins, et le meilleur chemin est le second.

```
R1#show bgp ipv4 unicast 10.100.1.1
BGP routing table entry for 10.100.1.1/32, version 2
Paths: (2 available, best #2, table default)
  Advertised to update-groups:
    1
  Refresh Epoch 1
  5 4
    10.1.5.5 from 10.1.5.5 (10.1.5.5)
      Origin IGP, localpref 100, valid, external
      rx pathid: 0, tx pathid: 0
  Refresh Epoch 1
  4
    10.1.3.4 from 10.1.3.4 (10.100.1.1)
      Origin IGP, metric 0, localpref 100, valid, external, best
      rx pathid: 0, tx pathid: 0x0
```

Seulement un chemin est sélectionné comme meilleur chemin en fonction BGP sur le meilleur algorithme de chemin BGP. C'est toujours le cas. Référez-vous au pour en savoir plus d'article d'[algorithme de sélection du meilleur chemin BGP](#).

Le chemin est appris d'un pair BGP ou d'une source, comme de la redistribution d'un protocole de routage dans le BGP. Quand il y a un changement du meilleur chemin, le BGP doit informer son pair en envoyant une mise à jour ou un retrait. Le retrait est envoyé quand le dernier chemin du préfixe BGP est retiré.

Voici un exemple où le préfixe est originaire localement par la commande réseau :

```
R4#show bgp ipv4 unicast 10.100.1.1
BGP routing table entry for 10.100.1.1/32, version 4
Paths: (1 available, best #1, table default)
  Advertised to update-groups:
    1
  Refresh Epoch 1
  Local
    0.0.0.0 from 0.0.0.0 (10.1.3.4)
      Origin IGP, metric 0, localpref 100, weight 32768, valid, sourced, local, best
      rx pathid: 0, tx pathid: 0x0
```

La sortie affiche l'**IGP d'origine**.

Voici un exemple où le préfixe est originaire localement par la commande connectée par redistribution :

```
R4#show bgp ipv4 unicast 10.100.1.1
BGP routing table entry for 10.100.1.1/32, version 7
Paths: (1 available, best #1, table default)
Flag: 0x820
  Not advertised to any peer
  Refresh Epoch 1
  Local
    0.0.0.0 from 0.0.0.0 (10.1.3.4)
      Origin incomplete, metric 0, localpref 100, weight 32768, valid, sourced, best
```

rx pathid: 0, tx pathid: 0x0

La sortie affiche l'origine inachevée.

Types de versions de table

Le nombre de version de table est une valeur 32 bits, et il y a quatre types de versions de table :

- Version de table BGP
- Version de table de Routing Information Base (NERVURE)
- Version de table de pair
- Version de table de préfixe

Ceux-ci sont encore expliqués dans l'utilisation de la section de version de table.

Nombre de version de table initial

Quand le BGP ne s'est renseigné sur aucun préfixe encore, la version de table globale, la version de table de NERVURE, et la version de table de pair sont 1, qui est le point commençant pour le nombre de version de table.

La commande BGP avec le mot clé **récapitulatif** te donne trois nombres de version de table. Le mot clé récapitulatif peut être donné pour toutes les familles d'adresse dans le BGP.

```
R1#show bgp ipv4 unicast summary
```

```
BGP router identifier 10.1.3.1, local AS number 1
```

```
BGP table version is 1, main routing table version 1
```

Neighbor	V	AS	MsgRcvd	MsgSent	TblVer	InQ	OutQ	Up/Down	State/PfxRcd
10.1.1.2	4	2	4	4	1	0	0	00:01:15	0
10.1.2.3	4	3	4	4	1	0	0	00:01:06	0
10.1.3.4	4	4	4	4	1	0	0	00:01:33	0

Vous pouvez visualiser la version de table de préfixe si vous regardez un préfixe dans la table BGP.

```
R1#show bgp ipv4 unicast 10.100.1.1/32
```

```
BGP routing table entry for 10.100.1.1/32, version 2
```

```
Paths: (1 available, best #1, table default)
```

```
Advertised to update-groups:
```

```
1
```

```
Refresh Epoch 1
```

```
4
```

```
10.1.3.4 from 10.1.3.4 (10.1.3.4)
```

```
Origin IGP, metric 0, localpref 100, valid, external, best
```

```
rx pathid: 0, tx pathid: 0x0
```

Vous pouvez visualiser la version de table si vous sélectionnez la commande interne de **show ip bgp** :

```
R1#show ip bgp internal
```

```
Time left for bestpath timer: 964 secs
```

```
Consistency-checker not enabled
```

```
Update generation pool version 8, messages 0, in pool 0, below 00:00:24.432.
```

```
Enhanced Refresh EOR Stalepath-time disabled
```

```
Enhanced Refresh max-eor-time disabled
```

```
Total number of BGP Acceptor process: 50, Spawned count: 0
```

Total number of neighbors: 4
Total number of sessions : 4
Established : 4
OpenConfirm : 0
OpenSent : 0
Active : 0
Connect : 0
Idle : 0
Closing : 0
Uninitialized : 0

Address-family IPv4 Unicast, Mode : RW

Table Versions : Current 39 Init 2 RIB 39

Start time : 00:00:18.919 Time elapsed 22:15:38.198
First Peer up in : 00:00:06.830 Exited Read-Only in : 00:01:07.966
Done with Install in : 00:01:07.967 Last Update-done in : 00:01:07.969
0 updates expanded
L3VPN Tunnel Encapsulated Paths : 0
Slow-peer detection is disabled BGP Nexthop scan:-
penalty: 0, Time since last run: 21:19:42.174, Next due in: none
Max runtime : 0 ms Latest runtime : 0 ms Scan count: 2
BGP General Scan:-
Max runtime : 1 ms Latest runtime : 0 ms Scan count: 0

BGP future scanner version: 1333

BGP scanner version: 0

Address-family IPv4 Multicast, Mode : RW

Table Versions : Current 1 Init 1 RIB 1

Start time : 00:00:18.919 Time elapsed 22:15:38.199
First Peer up in : never Exited Read-Only in : 00:00:10.286
Done with Install in : 00:00:10.286 Last Update-done in : never
0 updates expanded
L3VPN Tunnel Encapsulated Paths : 0
Slow-peer detection is disabled BGP Nexthop scan:-
penalty: 0, Time since last run: never, Next due in: none
Max runtime : 0 ms Latest runtime : 0 ms Scan count: 0
BGP General Scan:-
Max runtime : 1 ms Latest runtime : 0 ms Scan count: 0

BGP future scanner version: 1334

BGP scanner version: 0

Address-family MVPNv4 Unicast, Mode : RW

Table Versions : Current 1 Init 1 RIB 1

Start time : 00:00:18.919 Time elapsed 22:15:38.200
First Peer up in : never Exited Read-Only in : 00:00:10.286
Done with Install in : 00:00:10.286 Last Update-done in : never
0 updates expanded
L3VPN Tunnel Encapsulated Paths : 0
Slow-peer detection is disabled BGP Nexthop scan:-
penalty: 0, Time since last run: never, Next due in: none
Max runtime : 0 ms Latest runtime : 0 ms Scan count: 0
BGP General Scan:-
Max runtime : 1 ms Latest runtime : 0 ms Scan count: 0

BGP future scanner version: 1334

TX VPN optimization enabled.

Conditions pour un changement de version de table BGP

Pour que le nombre de version de table BGP change, il doit y a un changement du meilleur chemin et une modification propagée à la NERVURE. Une modification à la NERVURE pour un préfixe BGP se produit seulement si le préfixe est dans la NERVURE comme préfixe BGP. Si n'importe quel autre protocole de routage place le préfixe dans le routage, alors le préfixe BGP est marqué comme Nervure-panne. Dans ce cas, même si le meilleur chemin change, la version de table ne change pas.

Voici un exemple où la version de table BGP ne change pas. Le préfixe **10.100.1.1/32** BGP instruit de **R4** est également appris par une route statique configurée sur **R1**. Ainsi, **R1** installe l'artère statique dans la NERVURE, et le BGP sur **R1** marque le préfixe comme Nervure-panne, parce que ce n'est pas le BGP qui installe le préfixe dans la NERVURE. Aucune modification aux chemins BGP pour ce préfixe n'est propagée à la NERVURE. Ainsi quoiqu'il y ait une meilleure modification de chemin, la version de table BGP n'est pas envoyée, parce qu'il n'y a aucune mise à jour à la NERVURE.

```
R1#show bgp ipv4 unicast 10.100.1.1/32
BGP routing table entry for 10.100.1.1/32, version 8
Paths: (2 available, best #1, table default, RIB-failure(17))
  Advertised to update-groups:
    2
  Refresh Epoch 2
  4
    10.1.3.4 from 10.1.3.4 (10.100.1.1)
      Origin IGP, metric 0, localpref 100, valid, external, best
      rx pathid: 0, tx pathid: 0x0
  Refresh Epoch 2
  5 4
    10.1.5.5 from 10.1.5.5 (10.1.5.5)
      Origin IGP, localpref 100, valid, external
      rx pathid: 0, tx pathid: 0
```

```
R1#show ip route 10.100.1.1
Routing entry for 10.100.1.1/32
  Known via "static", distance 1, metric 0 (connected)
  Routing Descriptor Blocks:
  * directly connected, via Loopback0
    Route metric is 0, traffic share count is 1
```

Utilisation de version de table

Quand le meilleur chemin change les préfixes BGP, quelques choses doivent se produire :

- On doit annoncer La NERVURE.
- Les pairs BGP doivent être au courant.
- Le routeur doit dépister qui le pair BGP est au courant dont le meilleur chemin change.

La version de table BGP est le numéro principal utilisé. Ce nombre est identique que la version de table de préfixe la plus élevée de n'importe quel préfixe BGP pour une famille d'adresse spécifique. Supposez qu'il y a cinq préfixes dans la table BGP, avec les versions de table de préfixe 3, 6, 8, 10, et 18. La version de table BGP sera alors 18.

La version de table de pair est utilisée afin de dépister quels pairs doivent être au courant dont les préfixes pour lesquels il y avait des changements du meilleur chemin. La version de table de pair de chaque pair est vérifiée contre la version de table de préfixe des préfixes. Si la version de table de préfixe d'un préfixe est inférieure à la version de table de pair, alors le BGP doit envoyer une mise à jour pour ce préfixe à ce pair BGP. Par exemple, si le pair **10.1.1.2** a une version de table

de pair de **60**, puis ce pair est à jour pour tous les préfixes avec la version de table de préfixe de **60** et diminue. Le routeur doit envoyer une mise à jour BGP pour tous les préfixes avec une version de table de préfixe qui est le supérieur à **60**.

Une fois le routeur met à jour le pair BGP pour les préfixes changés le meilleur par chemin, le routeur met à jour la version de table de pair pour ce pair. Cette valeur de version de table de pair est ajustée pour apparier la valeur de la version de table de préfixe la plus élevée de tous les préfixes pour lesquels ce pair BGP était mis à jour. Supposez que la version de table de pair était **60**, et il y a deux préfixes avec les versions de table **61** et **62** de préfixe. Une fois que le routeur envoie les nouveaux meilleurs chemins pour ces deux préfixes à ce BGP scrutent, puis la version de table de pair est mise à jour à **62**.

La version de table de préfixe est le nombre de version de table relié au préfixe BGP. Il est changé quand le meilleur chemin change pour ce préfixe. Chaque fois que le meilleur chemin change pour un préfixe BGP, sa version de table de préfixe est envoyée, qui signifie qu'elle est mise à jour pour être égale au prochain numéro de version disponible. Supposez que le préfixe **10.0.0.0/8** a la version de table **27** de préfixe, et la version de table BGP est **30**. Dans ce cas, quand le meilleur chemin change pour le préfixe **10.0.0.0/8**, sa version de table de préfixe est envoyée à **31**.

La version de table de NERVURE est utilisée afin de dépister si la NERVURE doit être mise à jour après que les meilleures modifications de chemin BGP se produisent. La NERVURE doit être au courant des préfixes BGP qui ont un supérieur à de version de table de préfixe la version de table de NERVURE. Pour ces préfixes, il y a une NERVURE AJOUTENT, SUPPRIMENT, ou MODIFIENT l'événement.

Utilisation pour le dépannage

Afin de connaître quand le BGP a convergé, sélectionnez la commande de **show bgp summary**. Si la version de table BGP de pair égale la version de table BGP, ce pair a convergé. Si la version de table de routage de canalisation égale la version de table BGP, la NERVURE a convergé.

```
R1#show bgp ipv4 unicast summary
BGP router identifier 10.1.3.1, local AS number 1
BGP table version is 2, main routing table version 2
1 network entries using 144 bytes of memory
1 path entries using 80 bytes of memory
1/1 BGP path/bestpath attribute entries using 144 bytes of memory
1 BGP AS-PATH entries using 24 bytes of memory
0 BGP route-map cache entries using 0 bytes of memory
0 BGP filter-list cache entries using 0 bytes of memory
BGP using 392 total bytes of memory
BGP activity 1/0 prefixes, 1/0 paths, scan interval 60 secs
```

Neighbor	V	AS	MsgRcvd	MsgSent	TblVer	InQ	OutQ	Up/Down	State/PfxRcd
10.1.1.2	4	2	69	69	2	0	0	01:00:54	0
10.1.2.3	4	3	69	70	2	0	0	01:00:45	0
10.1.3.4	4	4	72	70	2	0	0	01:01:12	1

Il peut y avoir beaucoup de modifications aux versions de table BGP, et cela ne signifie pas toujours que quelque chose est erronée.

Supposez que le routeur est connecté à l'Internet, et avez la pleine table de routage d'Internet. Typiquement, il y a quelques modifications presque chaque seconde sur la table BGP d'Internet. Puis, le routeur doit recalculer le meilleur chemin pour quelques préfixes, et met à jour sa

NERVURE et ses pairs BGP. C'est comportement prévu.

Supposez que vous effacez un pair BGP (la session est remise à l'état initial), puis le routeur doit annoncer sa pleine table BGP à ce pair. On s'attend à ce qu'il pour que ce pair ait une version de table croissante. La version de table augmente pendant que le pair reçoit les préfixes BGP de nouveau. Le pair de envoi BGP n'augmente pas la version de table pour les préfixes BGP.

Voici un exemple. Les débuts de version de table avec **28**.

```
R1#show bgp ipv4 unicast summary
```

```
BGP router identifier 10.1.3.1, local AS number 1
BGP table version is 28, main routing table version 281
network entries using 144 bytes of memory2 path entries using 160 bytes of memory
2/1 BGP path/bestpath attribute entries using 288 bytes of memory
2 BGP AS-PATH entries using 48 bytes of memory
0 BGP route-map cache entries using 0 bytes of memory
0 BGP filter-list cache entries using 0 bytes of memory
BGP using 640 total bytes of memory
BGP activity 1/0 prefixes, 16/14 paths, scan interval 60 secs
```

Neighbor	V	AS	MsgRcvd	MsgSent	TblVer	InQ	OutQ	Up/Down	State/PfxRcd
10.1.1.2	4	2	117	125	28	0	0	01:43:50	0
10.1.2.3	4	3	117	125	28	0	0	01:43:53	0
10.1.3.4	4	4	10	12	28	0	0	00:04:22	1
10.1.5.5	4	5	55	63	28	0	0	00:45:45	1

```
R1#show bgp ipv4 unicast 10.100.1.1/32
```

```
BGP routing table entry for 10.100.1.1/32, version 28
```

```
Paths: (2 available, best #1, table default)
```

```
Advertised to update-groups:
```

```
1
```

```
Refresh Epoch 2
```

```
4
```

```
10.1.3.4 from 10.1.3.4 (10.100.1.1) <<< path from R4
```

```
Origin IGP, metric 0, localpref 100, valid, external, best
```

```
rx pathid: 0, tx pathid: 0x0
```

```
Refresh Epoch 2
```

```
5 4
```

```
10.1.5.5 from 10.1.5.5 (10.1.5.5) <<< path from R5
```

```
Origin IGP, localpref 100, valid, external
```

```
rx pathid: 0, tx pathid: 0
```

Conduisez un clair dur pour la session BGP vers R1 sur le pair **10.1.3.4 (R4)**. Le pair annonce seulement un préfixe **10.100.1.1/32** vers R1. **10.100.1.1/32** est appris de **R4** et de **R5**. Le meilleur chemin est le chemin de **R4**.

Assurez-vous que vous faites activer **interne de debug ip bgp** afin de voir ce qui arrive aux versions de table BGP. Vous devriez faire activer le **debug ip bgp updates** afin de voir ce qui se produit quand la mise à jour arrive.

```
R1#debug ip bgp updates
```

```
BGP updates debugging is on for address family: IPv4 Unicast
```

```
R1#debug ip bgp internal
```

```
BGP internal debugging is on
```

```
R1#show debugging
```

```
IP routing:
```

```
BGP internal debugging is on
```

BGP updates debugging is on for address family: IPv4 Unicast R1#
%BGP-5-NBR_RESET: Neighbor 10.1.3.4 reset (Peer closed the session) <<< BGP
session to R4 goes down
BGP: TX IPv4 Unicast Net global 10.100.1.1/32 Changed.
BGP: TX IPv4 Unicast Net global 10.100.1.1/32 RIB done.
BGP: TX IPv4 Unicast Net global 10.100.1.1/32 Changed.
BGP: TX IPv4 Unicast Mem global 1 1 10.1.3.4 Resetting counters.
BGP: TX IPv4 Unicast Mem global 1 1 10.1.3.4 Ignoring dummy policy change.
BGP: TX IPv4 Unicast Mem global 1 1 10.1.3.4 Resetting counters.
BGP: TX IPv4 Unicast Mem global 1 1 10.1.3.4 Ignoring dummy policy change.
BGP: TX IPv4 Unicast Mem global 1 1 10.1.3.4 Changing state from ACTIVE to DOWN
(session not established).
BGP: TX IPv4 Unicast Mem global 1 1 10.1.3.4 Removing from group (3 members left).
%BGP-5-ADJCHANGE: neighbor 10.1.3.4 Down Peer closed the session
%BGP_SESSION-5-ADJCHANGE: neighbor 10.1.3.4 IPv4 Unicast topology base removed
from session Peer
closed the session
BGP: TX IPv4 Unicast Mem global 10.1.3.4 State is DOWN (session not established).
BGP: net global:IPv4 Unicast:base 10.100.1.1/32 RIB-INSTALL Attempting to
install. <<< RIB gets informed
BGP: net global:IPv4 Unicast:base 10.100.1.1/32 RIB-INSTALL Built route type:
1024, flags: 200000, tag: 5,
metric: 0 path: 1.
BGP: net global:IPv4 Unicast:base 10.100.1.1/32 RIB-INSTALL Path 1, type: DEF,
gw: 10.1.5.5, idb: N/A,
topo_id: 0, src: 1.1.5.5, lbl: 1048577, flags: 0.
BGP: net global:IPv4 Unicast:base 10.100.1.1/32 RIB-INSTALL Installing 1 paths,
multipath limit 1 (from 1).
BGP(0): Revise route installing 1 of 1 routes for 10.100.1.1/32 -> 10.1.5.5
(global) to main IP table <<< **The remaining path through R5 gets installed
in the RIB**
BGP: net global:IPv4 Unicast:base 10.100.1.1/32 RIB-INSTALL Install successful.
BGP: TX IPv4 Unicast Net global 10.100.1.1/32 RIB done.
BGP: TX IPv4 Unicast Net global 10.100.1.1/32 RIB done.
BGP: TX IPv4 Unicast Tab RIB walk done version 29, added 1 topologies.
BGP: TX IPv4 Unicast Tab Executing.
BGP: TX IPv4 Unicast Wkr global 1 Cur Processing.
BGP: TX IPv4 Unicast Top global Appending nets from attr 0x9362CB4.
BGP: TX IPv4 Unicast Wkr global 1 Cur Attr change from 0x0 to 0x9362CB4.
**BGP(0): (base) 10.1.1.2 send UPDATE (format) 10.100.1.1/32, next 10.1.1.1,
metric 0, path 5 4 <<< R1 sends update for 10.100.1.1/32 for Table Version 29.
(bestpath is still the one from R5, i.e. the only one R1 has at this moment)**
BGP: TX IPv4 Unicast Wkr global 1 Cur Net 10.100.1.1/32 (Pxt 0x9F58FA0:0x0)
Formatted.
BGP: TX IPv4 Unicast Top global No attributes with modified nets.
BGP: TX IPv4 Unicast Top global Added tail marker with version 29.
BGP: TX IPv4 Unicast Wkr global 1 Cur Reached marker with version 29.
BGP: TX IPv4 Unicast Top global No attributes with modified nets.
BGP: TX IPv4 Unicast Wkr global 1 Cur Replicating.
BGP: TX IPv4 Unicast Wkr global 1 Cur Done (end of list), processed 1 attr(s),
1/1 net(s), 0 pos.
BGP: TX IPv4 Unicast Grp global 1 Checking EORs again (3/3).
BGP: TX IPv4 Unicast Grp global 1 Start minimum advertisement timer (30 secs).
BGP: TX IPv4 Unicast Wkr global 1 Cur Blocked (minimum advertisement interval).
BGP: TX IPv4 Unicast Wkr global 1 Cur Reached end of list.
BGP: TX IPv4 Unicast Grp global 1 Converged.
BGP: TX IPv4 Unicast Tab Processed 1 walker(s).
BGP: TX IPv4 Unicast Tab Generation completed.
BGP: TX IPv4 Unicast Top global Deleting first marker with version 28.
BGP: TX IPv4 Unicast Top global Collection reached marker 28 after 0 path
extension(s).
BGP: TX IPv4 Unicast Top global Collection done on marker 29 after 1 path
extension(s).

BGP: TX IPv4 Unicast Top global Collection done on marker 29 after 0 path extension(s).

BGP: TX IPv4 Unicast Mem global 10.1.3.4 Policy change while no group and member is DOWN.

BGP: TX IPv4 Unicast Mem global 10.1.3.4 Changing state from DOWN to WAIT (pending advertised bit allocation).

BGP: TX IPv4 Unicast Mem global 1 1 10.1.3.4 Added to group (now has 4 members).

BGP: TX IPv4 Unicast Mem global 1 1 10.1.3.4 Continuing into ACTIVE state.

BGP: TX IPv4 Unicast Mem global 1 1 10.1.3.4 Refresh Start-of-rib for afi 1, safi 1.

BGP: TX IPv4 Unicast Mem global 1 1 10.1.3.4 Full refresh requested.

BGP: TX IPv4 Unicast Mem global 1 1 10.1.3.4 Refresh has to wait for pathext prepend.

%BGP-5-ADJCHANGE: neighbor 10.1.3.4 Up <<< BGP session to R4 is up again. But, R1 did not learn the prefix 10.100.1.1/32 yet from R4.

BGP: nbr_topo global 10.1.3.4 IPv4 Unicast:base (0x63D50D0:1) rcvd Refresh Start-of-RIB

BGP: nbr_topo global 10.1.3.4 IPv4 Unicast:base (0x63D50D0:1) refresh_epoch is 2

BGP: TX IPv4 Unicast Top global Start pathext prepend.

BGP: TX IPv4 Unicast Tab Pathext prepend full table refresh.

BGP: TX IPv4 Unicast Tab Pathext prepend full table refresh.

BGP: TX IPv4 Unicast Top global Inserting initial marker.

BGP: TX IPv4 Unicast Top global Done pathext prepend (1 attrs).

BGP: TX IPv4 Unicast Grp global 1 Starting refresh after prepend completion.

BGP: TX IPv4 Unicast Mem global 1 1 10.1.3.4 Starting refresh (first member, 1, 0, marker).

BGP: TX IPv4 Unicast Wkr global 1 Ref Start at marker 1.

BGP: TX IPv4 Unicast Wkr global 1 Ref Unblocked

BGP: TX IPv4 Unicast Top global Collection done on marker 1 after 0 path extension(s).

BGP: TX IPv4 Unicast Tab Executing.

BGP: TX IPv4 Unicast Wkr global 1 Ref Processing.

BGP: TX IPv4 Unicast Wkr global 1 Ref Attr change from 0x0 to 0x9362CB4.

BGP(0): (base) 10.1.1.2 send UPDATE (format) 10.100.1.1/32, next 10.1.1.1, metric 0, path 5 4

BGP: TX IPv4 Unicast Wkr global 1 Ref Net 10.100.1.1/32 (Pxt 0x9F58FA0:0x0) Formatted.

BGP: TX IPv4 Unicast Wkr global 1 Ref Reached marker with version 29.

BGP: TX IPv4 Unicast Wkr global 1 Ref Replicating (pending member_pos processing).

BGP: TX IPv4 Unicast Mem global 1 1 10.1.3.4 Completed refresh.

BGP: TX IPv4 Unicast Mem global 1 1 10.1.3.4 Refresh stop.

BGP: TX IPv4 Unicast Grp global 1 Refresh complete.

BGP: TX IPv4 Unicast Wkr global 1 Ref Stop.

BGP: TX IPv4 Unicast Wkr global 1 Ref Blocked (not in list).

BGP: TX IPv4 Unicast Grp global 1 Converged.

BGP: TX IPv4 Unicast Mem global 1 1 10.1.3.4 Send EOR.

BGP: TX IPv4 Unicast Wkr global 1 Ref Suspending / blocked (member marker), processed 1 attr(s), 1/1 net(s), 1 pos.

BGP: TX IPv4 Unicast Tab Processed 1 walker(s).

BGP: TX IPv4 Unicast Tab Generation completed.

BGP: TX IPv4 Unicast Top global Deleting first marker with version 1.

BGP: TX IPv4 Unicast Top global Collection reached marker 1 after 0 path extension(s).

BGP: TX IPv4 Unicast Top global Collection done on marker 29 after 1 path extension(s).

BGP: TX IPv4 Unicast Top global Collection done on marker 29 after 0 path extension(s).

BGP(0): 10.1.3.4 rcvd UPDATE w/ attr: nexthop 10.1.3.4, origin i, metric 0, merged path4, AS_PATH

BGP(0): 10.1.3.4 rcvd 10.100.1.1/32 <<< R1 received 10.100.1.1/32 from

R4 again

```
BGP: TX IPv4 Unicast Net global 10.100.1.1/32 Changed.
BGP: nbr_topo global 10.1.3.4 IPv4 Unicast:base (0x63D50D0:1) rcvd Refresh
End-of-RIB
BGP: net global:IPv4 Unicast:base 10.100.1.1/32 RIB-INSTALL Attempting to install.
BGP: net global:IPv4 Unicast:base 10.100.1.1/32 RIB-INSTALL Built route type:
1024, flags: 200000, tag: 4, metric: 0 path: 1.
BGP: net global:IPv4 Unicast:base 10.100.1.1/32 RIB-INSTALL Path 1, type: DEF,
gw: 10.1.3.4, idb: N/A, topo_id: 0, src: 1.1.3.4, lbl: 1048577, flags: 0.
BGP: net global:IPv4 Unicast:base 10.100.1.1/32 RIB-INSTALL Installing 1 paths,
multipath limit 1 (from 1).
BGP(0): Revise route installing 1 of 1 routes for 10.100.1.1/32 -> 10.1.3.4
(global) to main IP table
BGP: net global:IPv4 Unicast:base 10.100.1.1/32 RIB-INSTALL Install successful.
BGP: TX IPv4 Unicast Net global 10.100.1.1/32 RIB done.
BGP: TX IPv4 Unicast Net global 10.100.1.1/32 RIB done.
BGP: TX IPv4 Unicast Tab RIB walk done version 30, added 1 topologies.
BGP: TX IPv4 Unicast Tab Executing.
BGP: TX IPv4 Unicast Tab Generation completed.
BGP: TX Member message pool under period (60 < 600).
BGP: TX IPv4 Unicast Mem global 1 1 10.1.2.3 State is ACTIVE (ready).
BGP: TX IPv4 Unicast Grp global 1 Minimum advertisement timer expired.
BGP: TX IPv4 Unicast Wkr global 1 Cur Unblocked
BGP: TX IPv4 Unicast Tab Executing.
BGP: TX IPv4 Unicast Wkr global 1 Cur Processing.
BGP: TX IPv4 Unicast Top global Appending nets from attr 0x9362D54.
BGP: TX IPv4 Unicast Wkr global 1 Cur Attr change from 0x0 to 0x9362D54.
BGP(0): (base) 10.1.1.2 send UPDATE (format) 10.100.1.1/32, next 10.1.1.1,
metric 0, path 4 <<< R1 sends an update for 10.100.1.1/32 for Table Version
30 (bestpath is again the one from R4)
BGP: TX IPv4 Unicast Wkr global 1 Cur Net 10.100.1.1/32 (Pxt 0x9F58FA0:0x0)
Formatted.
BGP: TX IPv4 Unicast Top global No attributes with modified nets.
BGP: TX IPv4 Unicast Top global Added tail marker with version 30.
BGP: TX IPv4 Unicast Wkr global 1 Cur Reached marker with version 30.
BGP: TX IPv4 Unicast Top global No attributes with modified nets.
BGP: TX IPv4 Unicast Wkr global 1 Cur Replicating.
BGP: TX IPv4 Unicast Wkr global 1 Cur Done (end of list), processed 1
attr(s), 1/1 net(s), 0 pos.
BGP: TX IPv4 Unicast Grp global 1 Checking EORs again (4/4).
BGP: TX IPv4 Unicast Grp global 1 Start minimum advertisement timer (30 secs).
BGP: TX IPv4 Unicast Wkr global 1 Cur Blocked (minimum advertisement interval).
BGP: TX IPv4 Unicast Wkr global 1 Cur Reached end of list.
BGP: TX IPv4 Unicast Grp global 1 Converged.
BGP: TX IPv4 Unicast Tab Processed 1 walker(s).
BGP: TX IPv4 Unicast Tab Generation completed.
BGP: TX IPv4 Unicast Top global Deleting first marker with version 29.
BGP: TX IPv4 Unicast Top global Collection reached marker 29 after 0 path
extension(s).
BGP: TX IPv4 Unicast Top global Collection done on marker 30 after 1 path
extension(s).
BGP: TX IPv4 Unicast Top global Collection done on marker 30 after 0 path
extension(s).
BGP: TX IPv4 Unicast Tab RIB walk done version 30, added 0 topologies.
```

Toutes les versions de table sont à 30 maintenant :

R1#show bgp ipv4 unicast summary

```
BGP router identifier 10.1.3.1, local AS number 1
BGP table version is 30, main routing table version 30
1 network entries using 144 bytes of memory
2 path entries using 160 bytes of memory
2/1 BGP path/bestpath attribute entries using 288 bytes of memory
2 BGP AS-PATH entries using 48 bytes of memory
```

```
0 BGP route-map cache entries using 0 bytes of memory
0 BGP filter-list cache entries using 0 bytes of memory
BGP using 640 total bytes of memory
BGP activity 1/0 prefixes, 17/15 paths, scan interval 60 secs
```

Neighbor	V	AS	MsgRcvd	MsgSent	TblVer	InQ	OutQ	Up/Down	State/PfxRcd
10.1.1.2	4	2	127	135	30	0	0	01:52:42	0
10.1.2.3	4	3	126	136	30	0	0	01:52:45	0
10.1.3.4	4	4	12	14	30	0	0	00:06:25	1
10.1.5.5	4	5	64	73	30	0	0	00:54:37	1

```
R1#show bgp ipv4 unicast 10.100.1.1/32
```

```
BGP routing table entry for 10.100.1.1/32, version 30
```

```
Paths: (2 available, best #1, table default)
```

```
Advertised to update-groups:
```

```
1
```

```
Refresh Epoch 2
```

```
4
```

```
10.1.3.4 from 10.1.3.4 (10.100.1.1)
```

```
Origin IGP, metric 0, localpref 100, valid, external, best
```

```
rx pathid: 0, tx pathid: 0x0
```

```
Refresh Epoch 2
```

```
5 4
```

```
10.1.5.5 from 10.1.5.5 (10.1.5.5)
```

```
Origin IGP, localpref 100, valid, external
```

```
rx pathid: 0, tx pathid: 0
```

En fin de compte, sur **R1**, il y avait deux meilleures modifications de chemin. Ainsi, la version de table obtenue a envoyé par **2**.

D'abord, le pair **10.1.3.4** est allé vers le bas sur **R1**. Le meilleur chemin a changé au chemin reçu de **R5**. La version de table a grimpé jusqu'au prochain nombre disponible, qui était **29**. La version de table de préfixe a été aussi bien envoyée à **29**. La NERVURE a été mise à jour avec ce nouveau meilleur chemin. La version de table de la NERVURE a été grimpée jusqu'à **29**. Puis, **R1** a envoyé une mise à jour au pair **10.1.1.2** BGP pour le nouveau meilleur chemin et a mis à jour la version de table de pair à **29**. Chaque autre pair a été aussi bien mis à jour.

En second lieu, une fois que le pair **10.1.3.4** était de nouveau, **R1** a reçu la mise à jour pour **10.100.1.1/32** de **R4** et a recalculé le meilleur chemin. Le chemin de **R4** était le nouveau meilleur chemin, qui a entraîné la version de table et la version de table de préfixe à envoyer au prochain nombre disponible de **30**. De nouveau, la NERVURE et tous autres pairs BGP étaient mis à jour, et les versions de table de NERVURE et de pair ont été mises à jour à **30**. La version de table a été envoyée seulement par on chaque fois ici. Cependant, si les autres préfixes BGP subissaient d'autres modifications, cette version de table serait envoyée par plus d'une, parce qu'elle branche chaque fois au prochain nombre disponible.

Si vous entrez dans le **clear ip bgp** commandez pour un pair BGP, ce routeur renvoie ses préfixes BGP à ce pair. Ceci n'entraîne pas un changement du meilleur chemin sur le pair de réception BGP. Par conséquent, il n'y a aucun changement de la version de table sur ce pair.

Quand vous exécutez le **debug ip bgp updates** sur le routeur récepteur, vous voyez :

```
BGP(0): 10.1.3.4 rcvd UPDATE w/ attr: nexthop 10.1.3.4, origin i,
metric 0, merged path 4, AS_PATH
```

```
BGP(0): 10.1.3.4 rcvd 10.100.1.1/32...duplicate ignored
```

La mise à jour reçue est identifiée comme doublon, ainsi elle est ignorée et aucune meilleure modification de chemin ne se produit.

Supposez que vous avez un routeur avec 100.000 préfixes dans la table BGP, et la version de table BGP augmente de 100.000 chaque minute. Ceci n'est pas prévu, et le comportement doit être examiné. Une raison pour le comportement pourrait être que le prochain-saut pour les préfixes BGP s'agit pour tous les préfixes chaque minute.

Un des résultats quand la version de table BGP augmente rapidement est que le routeur BGP et le BGP de processus E/S sont occupés, qui pourraient entraîner une CPU de routeur élevée constante.