Comprendere e risolvere i problemi relativi al controllo dei percorsi nelle implementazioni SD-WAN Secure Firewall

Sommario

- Introduzione
- Prerequisiti
  - Requisiti
  - Componenti usati
  - Informazioni sulle funzionalità
- Scenario di distribuzione
  - Hub e spoke doppi con ISP doppio
    - Topologia sottostante
    - Topologia overlay
    - Configurazione
    - Verifica E Risoluzione Dei Problemi
    - Configurazione comune a tutti i dispositivi
    - Spoke 1 e 2 (IBGP con HUB1 e EBGP con HUB2)
    - HUB1 (peer IBGP con spoke)
    - HUB2 (peering EBGP con gli spoke)
    - Topologia di routing
    - Raggio1
    - HUB1
    - HUB2
    - Raggio 2
- Conclusioni
- Informazioni correlate

Introduzione

Questo documento descrive il controllo del routing in BGP per VPN basate su route che usano Cisco SD-WAN su un firewall sicuro.

Prerequisiti

Requisiti

Cisco raccomanda la conoscenza dei seguenti argomenti:

- IKEv2
- VPN basata sul percorso
- VTI (Virtual Tunnel Interfaces)

- IPSec
- BGP
- Attributi BGP come tag della community e riflettori di route
- Funzione SD-WAN su un firewall sicuro

Componenti usati

Le informazioni fornite in questo documento si basano su:

- Cisco Secure Firewall Threat Defense 7.7.10
- Cisco Secure Firewall Management Center 7.7.10

Le informazioni discusse in questo documento fanno riferimento a dispositivi usati in uno specifico ambiente di emulazione. Su tutti i dispositivi menzionati nel documento la configurazione è stata ripristinata ai valori predefiniti. Se la rete è operativa, valutare attentamente eventuali conseguenze derivanti dall'uso dei comandi.

Informazioni sulle funzionalità

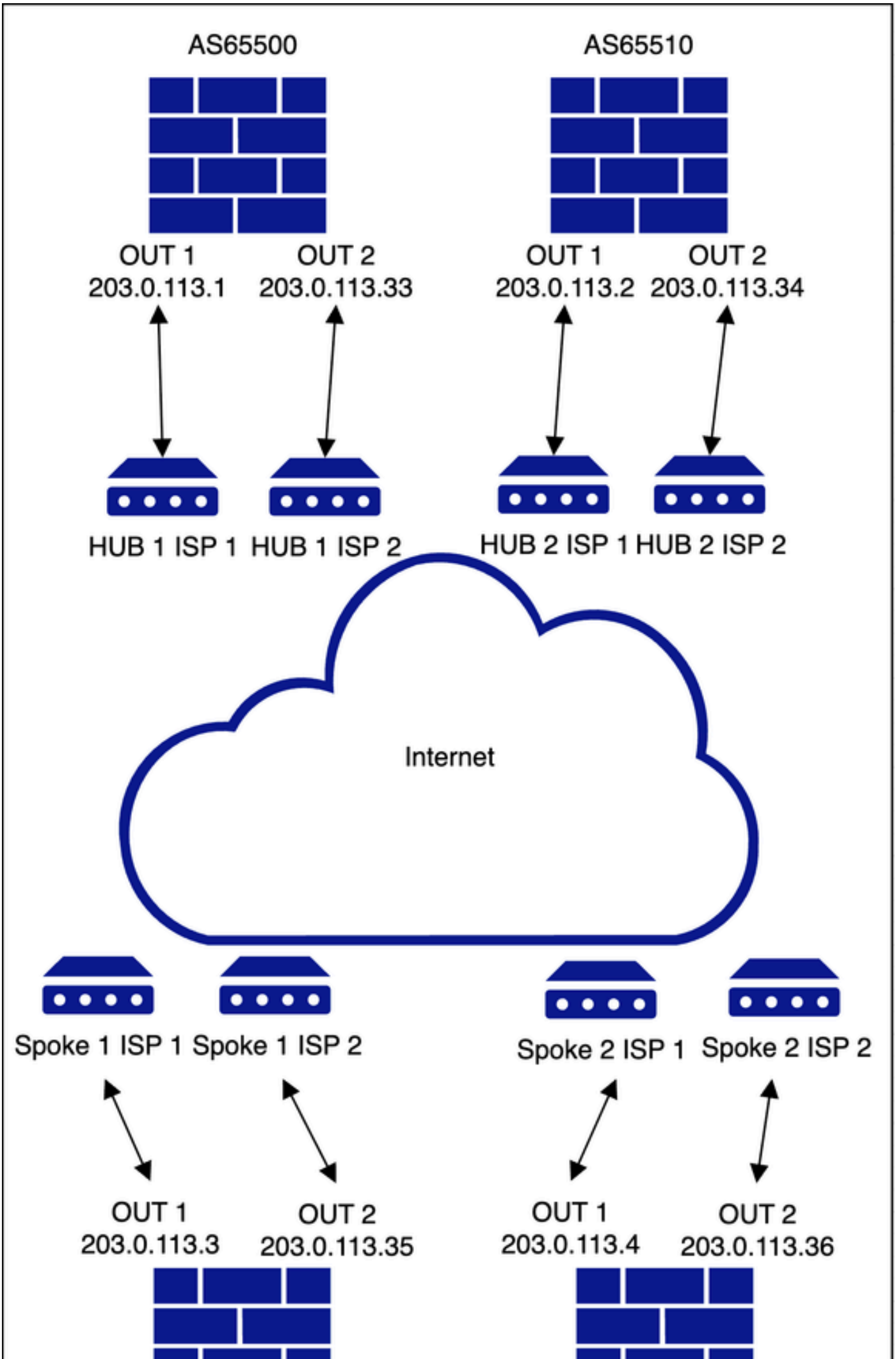
Con la nuova implementazione SD-WAN per la VPN da sito a sito basata su route con BGP abilitato per la sovrimpressione, Cisco concentra la propria attenzione sugli attributi BGP chiave per implementare il routing della sovrimpressione sicura e senza loop, garantendo che le reti di sovrimpressione e sovrapposizione rimangano segregate in tutta la topologia. Questa distribuzione garantisce inoltre che non sia necessario alcun intervento manuale per regolare gli attributi rilevanti.

Scenario di distribuzione

Selezionare una topologia che includa sia le connessioni iBGP che eBGP tra HUB e spoke. Questo approccio offre la massima visibilità sui controlli di routing implementati come parte della soluzione SD-WAN sui firewall sicuri di Cisco.

Hub e spoke doppi con ISP doppio

Topologia sottostante


AS65500
AS65510
OUT 1
203.0.113.1
OUT 2
203.0.113.33
OUT 1
203.0.113.2
OUT 2
203.0.113.34
HUB 1 ISP 1
HUB 1 ISP 2
HUB 2 ISP 1
HUB 2 ISP 2
Internet
Spoke 1 ISP 1
Spoke 1 ISP 2
Spoke 2 ISP 1
Spoke 2 ISP 2
OUT 1
203.0.113.3
OUT 2
203.0.113.35
OUT 1
203.0.113.4
OUT 2
203.0.113.36

```
community-list standard FMC_VPN_COMMUNITY_101010 permit 101010

<<<<<<<<<<


community-list standard FMC_VPN_COMMUNITY_202020 permit 202020

<<<<<<<<<<
```

Si noti che esiste un'unica coppia di route map in entrata e in uscita per topologia anche se le configurazioni sono identiche per entrambe le topologie, solo la convenzione di denominazione è univoca per topologia. Nel nostro scenario, FMC_VPN_RMAP_COMMUNITY_IN_8589939614 e FMC_VPN_RMAP_COMMUNITY_OUT_858939614 sono per la topologia 1 mentre FMC_VPN_RMAP_COMMUNITY_IN_8589942200 e FMC_VPN_RMAP_COMMUNITY_OUT_8589942200 sono destinati alla topologia 2.

<#root>

firepower# show running-config route-map

Topology 1
Inbound
route-map

FMC_VPN_RMAP_COMMUNITY_IN_8589939614

 permit 10
match community FMC_VPN_COMMUNITY_101010 exact-match

set community 202020

route-map

FMC_VPN_RMAP_COMMUNITY_IN_8589939614

 permit 20
match community FMC_VPN_COMMUNITY_202020 exact-match

Outbound
route-map

FMC_VPN_RMAP_COMMUNITY_OUT_8589939614

 permit 10
match community FMC_VPN_COMMUNITY_101010 exact-match
set metric 1

route-map

FMC_VPN_RMAP_COMMUNITY_OUT_8589939614

 permit 20
match community FMC_VPN_COMMUNITY_202020 exact-match
set metric 100

route-map

FMC_VPN_RMAP_COMMUNITY_OUT_8589939614

deny 100

Topology 2
Inbound
route-map

FMC_VPN_RMAP_COMMUNITY_IN_8589942200

permit 10
match community FMC_VPN_COMMUNITY_101010 exact-match

set community 202020

route-map

FMC_VPN_RMAP_COMMUNITY_IN_8589942200

permit 20
match community FMC_VPN_COMMUNITY_202020 exact-match

Outbound
route-map

FMC_VPN_RMAP_COMMUNITY_OUT_8589942200

permit 10
match community FMC_VPN_COMMUNITY_101010 exact-match
set metric 1

route-map

FMC_VPN_RMAP_COMMUNITY_OUT_8589942200

permit 20
match community FMC_VPN_COMMUNITY_202020 exact-match
set metric 100

route-map

FMC_VPN_RMAP_COMMUNITY_OUT_8589942200

deny 100

Common Across All The Hubs & Spokes Wherever Redistribution Of Inside Network Is Present

route-map

FMC_VPN_CONNECTED_DIST_RMAP_101010

permit 10
match interface inside
set community 101010

Viene mostrata la configurazione BGP tra i dispositivi nella topologia:

Spoke 1 e 2 (IBGP con HUB1 e EBGP con HUB2)

<#root>

```
firepower# show running-config router bgp


router bgp 65500
bgp log-neighbor-changes
address-family ipv4 unicast
neighbor 198.51.100.1 remote-as 65500
<<<<< tunnel from spokes to HUB 1 via ISP1


neighbor 198.51.100.1 activate
neighbor 198.51.100.1 send-community
neighbor 198.51.100.1 route-map FMC_VPN_RMAP_COMMUNITY_IN_8589939614 in
neighbor 198.51.100.1 route-map FMC_VPN_RMAP_COMMUNITY_OUT_8589939614 out
neighbor 198.51.100.2 remote-as 65510
<<<<< tunnel from spokes to HUB 2 via ISP1


neighbor 198.51.100.2 ebgp-multihop 2
neighbor 198.51.100.2 activate
neighbor 198.51.100.2 send-community
neighbor 198.51.100.2 route-map FMC_VPN_RMAP_COMMUNITY_IN_8589939614 in
neighbor 198.51.100.2 route-map FMC_VPN_RMAP_COMMUNITY_OUT_8589939614 out
neighbor 198.51.100.3 remote-as 65500
<<<<< tunnel from spokes to HUB 1 via ISP2


neighbor 198.51.100.3 activate
neighbor 198.51.100.3 send-community
neighbor 198.51.100.3 route-map FMC_VPN_RMAP_COMMUNITY_IN_8589942200 in
neighbor 198.51.100.3 route-map FMC_VPN_RMAP_COMMUNITY_OUT_8589942200 out
neighbor 198.51.100.4 remote-as 65510
<<<<< tunnel from spokes to HUB 2 via ISP2


neighbor 198.51.100.4 ebgp-multihop 2
neighbor 198.51.100.4 activate
neighbor 198.51.100.4 send-community
neighbor 198.51.100.4 route-map FMC_VPN_RMAP_COMMUNITY_IN_8589942200 in
neighbor 198.51.100.4 route-map FMC_VPN_RMAP_COMMUNITY_OUT_8589942200 out
redistribute connected route-map FMC_VPN_CONNECTED_DIST_RMAP_101010
<<<<<<< route-map to redistribute inside network into BGP


maximum-paths 8
maximum-paths ibgp 8
no auto-summary
no synchronization
exit-address-family
```

HUB1 (peer IBGP con spoke)

<#root>

```
firepower# show running-config router bgp
```

```
router bgp 65500
bgp log-neighbor-changes
address-family ipv4 unicast
neighbor 198.51.100.10 remote-as 65500
```

<<<<< tunnel from HUB 1 to Spoke 1 via ISP 1

```
neighbor 198.51.100.10 activate
neighbor 198.51.100.10 send-community
neighbor 198.51.100.10 route-reflector-client
neighbor 198.51.100.10 next-hop-self
neighbor 198.51.100.10 route-map FMC_VPN_RMAP_COMMUNITY_IN_8589939614 in
neighbor 198.51.100.10 route-map FMC_VPN_RMAP_COMMUNITY_OUT_8589939614 out
neighbor 198.51.100.11 remote-as 65500
```

<<<<< tunnel from HUB 1 to Spoke 2 via ISP 1

```
neighbor 198.51.100.11 activate
neighbor 198.51.100.11 send-community
neighbor 198.51.100.11 route-reflector-client
neighbor 198.51.100.11 next-hop-self
neighbor 198.51.100.11 route-map FMC_VPN_RMAP_COMMUNITY_IN_8589939614 in
neighbor 198.51.100.11 route-map FMC_VPN_RMAP_COMMUNITY_OUT_8589939614 out
neighbor 198.51.100.70 remote-as 65500
```

<<<<< tunnel from HUB 1 to Spoke 1 via ISP 2

```
neighbor 198.51.100.70 activate
neighbor 198.51.100.70 send-community
neighbor 198.51.100.70 route-reflector-client
neighbor 198.51.100.70 next-hop-self
neighbor 198.51.100.70 route-map FMC_VPN_RMAP_COMMUNITY_IN_8589942200 in
neighbor 198.51.100.70 route-map FMC_VPN_RMAP_COMMUNITY_OUT_8589942200 out
neighbor 198.51.100.71 remote-as 65500
```

<<<<< tunnel from HUB 1 to Spoke 2 via ISP 2

```
neighbor 198.51.100.71 activate
neighbor 198.51.100.71 send-community
neighbor 198.51.100.71 route-reflector-client
neighbor 198.51.100.71 next-hop-self
neighbor 198.51.100.71 route-map FMC_VPN_RMAP_COMMUNITY_IN_8589942200 in
neighbor 198.51.100.71 route-map FMC_VPN_RMAP_COMMUNITY_OUT_8589942200 out
no auto-summary
no synchronization
exit-address-family
```

HUB2 (peering EBGP con gli spoke)

<#root>

firepower# show running-config router bgp

```
router bgp 65510
```

```
bgp log-neighbor-changes
address-family ipv4 unicast
neighbor 198.51.100.40 remote-as 65500
```

<<<<< tunnel from HUB 2 to Spoke 1 via ISP 1

```
neighbor 198.51.100.40 ebgp-multihop 2
neighbor 198.51.100.40 activate
neighbor 198.51.100.40 send-community
neighbor 198.51.100.40 next-hop-self
neighbor 198.51.100.40 as-override
neighbor 198.51.100.40 route-map FMC_VPN_RMAP_COMMUNITY_IN_8589939614 in
neighbor 198.51.100.40 route-map FMC_VPN_RMAP_COMMUNITY_OUT_8589939614 out
neighbor 198.51.100.41 remote-as 65500
```

<<<<< tunnel from HUB 2 to Spoke 2 via ISP 1

```
neighbor 198.51.100.41 ebgp-multihop 2
neighbor 198.51.100.41 activate
neighbor 198.51.100.41 send-community
neighbor 198.51.100.41 next-hop-self
neighbor 198.51.100.41 as-override
neighbor 198.51.100.41 route-map FMC_VPN_RMAP_COMMUNITY_IN_8589939614 in
neighbor 198.51.100.41 route-map FMC_VPN_RMAP_COMMUNITY_OUT_8589939614 out
neighbor 198.51.100.100 remote-as 65500
```

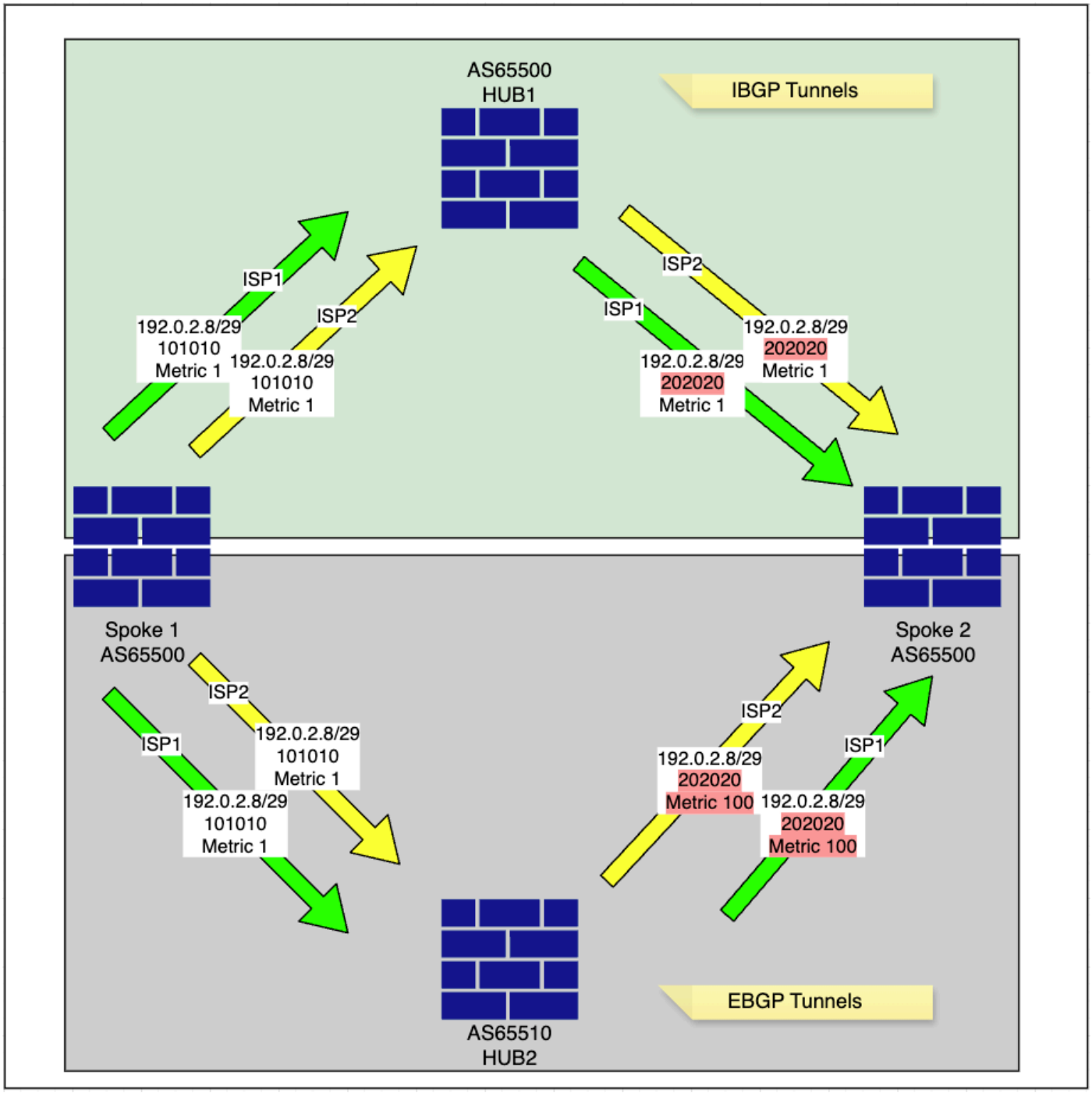
<<<<< tunnel from HUB 2 to Spoke 1 via ISP 2

```
neighbor 198.51.100.100 ebgp-multihop 2
neighbor 198.51.100.100 activate
neighbor 198.51.100.100 send-community
neighbor 198.51.100.100 next-hop-self
neighbor 198.51.100.100 as-override
neighbor 198.51.100.100 route-map FMC_VPN_RMAP_COMMUNITY_IN_8589942200 in
neighbor 198.51.100.100 route-map FMC_VPN_RMAP_COMMUNITY_OUT_8589942200 out
neighbor 198.51.100.101 remote-as 65500
```

<<<<< tunnel from HUB 2 to Spoke 2 via ISP 2

```
neighbor 198.51.100.101 ebgp-multihop 2
neighbor 198.51.100.101 activate
neighbor 198.51.100.101 send-community
neighbor 198.51.100.101 next-hop-self
neighbor 198.51.100.101 as-override
neighbor 198.51.100.101 route-map FMC_VPN_RMAP_COMMUNITY_IN_8589942200 in
neighbor 198.51.100.101 route-map FMC_VPN_RMAP_COMMUNITY_OUT_8589942200 out
no auto-summary
no synchronization
exit-address-family
```

Topologia di routing



- Lo spoke annuncia la propria rete interna, 192.0.2.8/29, in BGP con un tag della community specifico di 101010, come configurato nella route-map FMC_VPN_CONNECTED_DIST_RMAP_101010.

Raggio1

<#root>

```
Spoke1# show bgp community 101010 exact-match <<<< to verify the exact network redistributed into BGP
```

```
BGP table version is 4, local router ID is 203.0.113.35
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale, m multipath
```

```
Origin codes: i - IGP, e - EGP, ? - incomplete

   Network          Next Hop        Metric LocPrf Weight  Path
*> 192.0.2.8/29     0.0.0.0              0         32768  ?
```

<<<<<<<<<<< local inside network

- Spoke modifica il valore metrico della propria rete interna, 192.0.2.8/29, e lo annuncia agli hub, come configurato nelle route-map FMC_VPN_RMAP_COMMUNITY_OUT_8589939614 e FMC_VPN_RMAP_COMMUNITY_OUT_8589942200.

<#root>

```
route-map
```

FMC_VPN_RMAP_COMMUNITY_OUT_8589942200

```
 permit 10
match community FMC_VPN_COMMUNITY_101010 exact-match
set metric 1

route-map
```

FMC_VPN_RMAP_COMMUNITY_OUT_8589942200

```
 permit 20
match community FMC_VPN_COMMUNITY_202020 exact-match
set metric 100

route-map
```

FMC_VPN_RMAP_COMMUNITY_OUT_8589942200

```
 deny 100
```

- L'HUB1 apprende la rete Spoke 1 192.0.2.8/29 con il tag della community 101010 e cambia il tag della community in 202020, mantenendo la metrica prima di inoltrarla ad altri spoke, come definito nelle route-map configurate.

HUB1

<#root>

```
Route-Map for ISP1 DVTI
Inbound
route-map
```

FMC_VPN_RMAP_COMMUNITY_IN_8589939614

```
 permit 10
match community FMC_VPN_COMMUNITY_101010 exact-match
set community 202020

route-map
```

FMC_VPN_RMAP_COMMUNITY_IN_8589939614

```
 permit 20
match community FMC_VPN_COMMUNITY_202020 exact-match
```

Outbound
route-map

FMC_VPN_RMAP_COMMUNITY_OUT_8589939614

```
 permit 10
match community FMC_VPN_COMMUNITY_101010 exact-match
set metric 1
set ip next-hop 198.51.100.1
```

<<<<<<<<<< only next-hop is changed in ISP2 tunnel route-map with ISP2 DVTI IP

route-map

FMC_VPN_RMAP_COMMUNITY_OUT_8589939614

```
 permit 20
match community FMC_VPN_COMMUNITY_202020 exact-match
set metric 100
set ip next-hop 198.51.100.1
```

<<<<<<<<<< only next-hop is changed in ISP2 tunnel route-map with ISP2 DVTI IP

route-map

FMC_VPN_RMAP_COMMUNITY_OUT_8589939614

```
 deny 100
```

Route-Map for ISP2 DVTI

Inbound
route-map

FMC_VPN_RMAP_COMMUNITY_IN_8589942200

```
 permit 10
match community FMC_VPN_COMMUNITY_101010 exact-match
set community 202020
```

route-map

FMC_VPN_RMAP_COMMUNITY_IN_8589942200

```
 permit 20
match community FMC_VPN_COMMUNITY_202020 exact-match
```

Outbound
route-map

FMC_VPN_RMAP_COMMUNITY_OUT_8589942200

```
 permit 10
match community FMC_VPN_COMMUNITY_101010 exact-match
set metric 1
set ip next-hop 198.51.100.3
```

<<<<<<<<<< only next-hop is changed in ISP2 tunnel route-map with ISP2 DVTI IP

route-map

FMC_VPN_RMAP_COMMUNITY_OUT_8589942200

```
 permit 20
match community FMC_VPN_COMMUNITY_202020 exact-match
set metric 100
set ip next-hop 198.51.100.3
```

<<<<<<<<<< only next-hop is changed in ISP2 tunnel route-map with ISP2 DVTI IP

route-map

FMC_VPN_RMAP_COMMUNITY_OUT_8589942200

```
 deny 100
```

<#root>

HUB1# show bgp community 202020 exact-match <<<< this will confirm if received prefixes have community t

```
BGP table version is 5, local router ID is 198.51.100.3
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale, m multipath
Origin codes: i - IGP, e - EGP, ? - incomplete

   Network          Next Hop        Metric LocPrf Weight  Path
* i192.0.2.8/29     198.51.100.70        1    100      0  ?
*>i                 198.51.100.10        1    100      0  ?
* i192.0.2.16/29    198.51.100.71        1    100      0  ?
*>i                 198.51.100.11        1    100      0  ?
```

<#root>

HUB1# show bgp 192.0.2.8 <<<< this will display available paths in BGP for the network

```
BGP routing table entry for 192.0.2.8/29, version 4
Paths: (2 available, best #2, table default)
  Advertised to update-groups:
             1          2
  Local, (Received from a RR-client)
    198.51.100.70 from 198.51.100.70 (203.0.113.35)
```

<<<<< spoke 1 ISP 2 tunnel to HUB 1

```
      Origin incomplete, metric 1, localpref 100, valid, internal
      Community: 202020
  Local, (Received from a RR-client)
    198.51.100.10 from 198.51.100.10 (203.0.113.35)
```

<<<<< spoke 1 ISP 1 tunnel to HUB 1

```
      Origin incomplete, metric 1, localpref 100, valid, internal, best
      Community: 202020
```

<<<<< community updated as per the route-map configured on spoke side

<#root>

HUB1# show route 192.0.2.8

```
Routing entry for 192.0.2.8 255.255.255.248
  Known via "bgp 65500", distance 200, metric 1, type internal
  Last update from 198.51.100.10 0:09:18 ago
  Routing Descriptor Blocks:
```

*** 198.51.100.10, from 198.51.100.10, 0:09:18 ago**

```
      Route metric is 1, traffic share count is 1
      AS Hops 0
      MPLS label: no label string provided
```

<#root>

HUB1# show bgp ipv4 unicast neighbors 198.51.100.10 routes <<<<<< to check specific prefixes learnt via

```
BGP table version is 5, local router ID is 198.51.100.3
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale, m multipath
Origin codes: i - IGP, e - EGP, ? - incomplete

   Network          Next Hop        Metric LocPrf Weight  Path
*>i192.0.2.8/29     198.51.100.10        1    100      0  ?
```

<<< preferred route

```
Total number of prefixes 1
```

<#root>

HUB1# show bgp ipv4 unicast neighbors 198.51.100.70 routes <<<<<< to check specific prefixes learnt via

```
BGP table version is 5, local router ID is 198.51.100.3
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale, m multipath
Origin codes: i - IGP, e - EGP, ? - incomplete

   Network          Next Hop        Metric LocPrf Weight  Path
* i192.0.2.8/29     198.51.100.70        1    100      0  ?

Total number of prefixes 1
```

- L'HUB2 apprende anche la rete Spoke 1 192.0.2.8/29 con il tag della community 101010 e cambia il tag della community in 202020 e aggiorna la metrica a 100 prima di inoltrarla ad altri spoke, come specificato nelle route-map configurate. La modifica della metrica ha effetto a causa del peering eBGP. Ciò è dovuto al fatto che MED (Multi-Exit Discriminator) è un attributo BGP facoltativo e non transitivo utilizzato per influenzare il traffico in entrata suggerendo un punto di ingresso preferito in un AS. MED in genere non viene propagato tra peer iBGP all'interno dello stesso AS e viene invece pubblicizzato nei peer BGP (eBGP) esterni in sistemi autonomi diversi.

HUB2

<#root>

```
HUB2# show bgp community 202020 exact-match <<<< this will confirm if receieved prefixes have community

BGP table version is 5, local router ID is 198.51.100.4
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale, m multipath
Origin codes: i - IGP, e - EGP, ? - incomplete

   Network          Next Hop            Metric LocPrf Weight  Path
*  192.0.2.8/29     198.51.100.1

100

             0  65500 ?

<<<<<< advertised back by spoke 2 ISP1 to HUB2 previously learnt via HUB1 iBGP

*                   198.51.100.1

100

             0  65500 ?

<<<<<< advertised back by spoke 2 ISP2 to HUB2 previously learnt via HUB1 iBGP

*                   198.51.100.100       1             0  65500 ?

<<<<<< advertised by spoke 2 ISP tunnel

*>                  198.51.100.40        1             0  65500 ?

<<<<<< advertised and preferred by spoke 1 ISP 1 tunnel

*  192.0.2.16/29    198.51.100.1       100             0  65500 ?
*                   198.51.100.1       100             0  65500 ?
*                   198.51.100.101       1             0  65500 ?
*>                  198.51.100.41        1             0  65500 ?
```

<#root>

```
HUB2# show bgp 192.0.2.8 <<<< this will display available paths in BGP for the network
```

```
BGP routing table entry for 192.0.2.8/29, version 4
Paths: (4 available, best #4, table default)
  Advertised to update-groups:
             1          2
  65500
    198.51.100.1 (inaccessible) from 198.51.100.41 (203.0.113.36)
```

<<<<<< advertised back by spoke 2 ISP1 to HUB2 previously learnt via HUB1 iBGP

```
      Origin incomplete, metric 100, localpref 100, valid, external
      Community:
```

202020

```
  65500
    198.51.100.1 (inaccessible) from 198.51.100.101 (203.0.113.36)
```

<<<<<< advertised back by spoke 2 ISP2 to HUB2 previously learnt via HUB1 iBGP

```
      Origin incomplete, metric 100, localpref 100, valid, external
      Community:
```

202020

```
  65500
    198.51.100.100 from 198.51.100.100 (203.0.113.35)
```

<<<<<< advertised by spoke 1 ISP 2 tunnel

```
      Origin incomplete, metric 1, localpref 100, valid, external
      Community:
```

202020

```
  65500
    198.51.100.40 from 198.51.100.40 (203.0.113.35)
```

<<<<<< advertised and preferred by spoke 1 ISP 1 tunnel

```
      Origin incomplete, metric 1, localpref 100, valid, external, best
      Community:
```

202020

<#root>

HUB2# show bgp ipv4 unicast neighbors 198.51.100.40 routes <<<<<< to check specific prefixes learnt via

```
BGP table version is 5, local router ID is 198.51.100.4
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale, m multipath
Origin codes: i - IGP, e - EGP, ? - incomplete

   Network          Next Hop        Metric LocPrf Weight  Path
*> 192.0.2.8/29     198.51.100.40        1             0  65500 ?
```

<<<< preferred

```
*  192.0.2.16/29    198.51.100.1       100             0  65500 ?

Total number of prefixes 2
```

<#root>

HUB2# show bgp ipv4 unicast neighbors 198.51.100.41 routes <<<<<< to check specific prefixes learnt via

```
BGP table version is 5, local router ID is 198.51.100.4
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale, m multipath
Origin codes: i - IGP, e - EGP, ? - incomplete

   Network          Next Hop        Metric LocPrf Weight  Path
*  192.0.2.8/29     198.51.100.1       100             0  65500 ?
```

<<<<<<

```
*> 192.0.2.16/29    198.51.100.41        1             0  65500 ?

Total number of prefixes 2
```

<#root>

HUB2# show bgp ipv4 unicast neighbors 198.51.100.100 routes <<<<<< to check specific prefixes learnt via

```
BGP table version is 5, local router ID is 198.51.100.4
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale, m multipath
Origin codes: i - IGP, e - EGP, ? - incomplete

   Network          Next Hop        Metric LocPrf Weight  Path
*  192.0.2.8/29     198.51.100.100       1             0  65500 ?
```

<<<<<<

```
*  192.0.2.16/29    198.51.100.1       100             0  65500 ?

Total number of prefixes 2
```

<#root>

HUB2# show bgp ipv4 unicast neighbors 198.51.100.101 routes <<<<<< to check specific prefixes learnt via

```
BGP table version is 5, local router ID is 198.51.100.4
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale, m multipath
Origin codes: i - IGP, e - EGP, ? - incomplete

   Network          Next Hop        Metric LocPrf Weight  Path
*  192.0.2.8/29     198.51.100.1       100             0  65500 ?
```

<<<<<<

```
*  192.0.2.16/29    198.51.100.101       1             0  65500 ?

Total number of prefixes 2
```

- Lo spoke 2 riceve la rete Spoke 1 192.0.2.8/29 da entrambi i tunnel HUB1 ISP1 e HUB1 ISP2 con metrica 1, mentre riceve la stessa rete dai tunnel HUB2 ISP1 e HUB2 ISP2 con hop successivo aggiornato di HUB1.

Raggio 2

<#root>

```
Spoke2# show bgp community 202020 exact-match <<<< this will confirm if receieved prefixes have communit


BGP table version is 8, local router ID is 203.0.113.36
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale, m multipath
Origin codes: i - IGP, e - EGP, ? - incomplete

   Network          Next Hop        Metric LocPrf Weight  Path
*mi192.0.2.8/29     198.51.100.3         1    100      0  ?
*

>

i                 198.51.100.1         1    100      0  ?

<<<< HUB1 ISP1 route preferred

*                   198.51.100.2       100             0  65510 65510 ?
*                   198.51.100.4       100             0  65510 65510 ?
*  192.0.2.16/29    198.51.100.4       100             0  65510 65510 ?
*                   198.51.100.2       100             0  65510 65510 ?
```

<#root>

```
route-map

FMC_VPN_RMAP_COMMUNITY_IN_8589939614

 permit 10
match community FMC_VPN_COMMUNITY_101010 exact-match

set community 202020

route-map

FMC_VPN_RMAP_COMMUNITY_IN_8589956263

 permit 20
match community FMC_VPN_COMMUNITY_202020 exact-match
```

- Spoke 2 annuncia anche le reti apprese dall'HUB1 all'HUB2, come definito dalla route-map in uscita configurata, con la metrica aggiornata.

<#root>

route-map

FMC_VPN_RMAP_COMMUNITY_OUT_8589939614

 permit 10
match community FMC_VPN_COMMUNITY_101010 exact-match
set metric 1

route-map

FMC_VPN_RMAP_COMMUNITY_OUT_8589939614

 permit 20
match community FMC_VPN_COMMUNITY_202020 exact-match
set metric 100

 <<<<<

route-map

FMC_VPN_RMAP_COMMUNITY_OUT_8589939614

 deny 100

<#root>

Spoke2# show bgp ipv4 unicast neighbors 198.51.100.2 advertised-routes <<<<< to check specific prefixes

BGP table version is 8, local router ID is 203.0.113.36
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale, m multipath
Origin codes: i - IGP, e - EGP, ? - incomplete

   Network   Next Hop   Metric LocPrf Weight Path
*>i192.0.2.8/29   198.51.100.1   1   100   0 ?

<<<<<<<

*> 192.0.2.16/29   0.0.0.0   0   32768 ?

Total number of prefixes 2

<#root>

Spoke2# show bgp ipv4 unicast neighbors 198.51.100.4 advertised-routes <<<<< to check specific prefixes

BGP table version is 8, local router ID is 203.0.113.36
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale, m multipath
Origin codes: i - IGP, e - EGP, ? - incomplete

   Network   Next Hop   Metric LocPrf Weight Path
*>i192.0.2.8/29   198.51.100.1   1   100   0 ?

<<<<<<<

*> 192.0.2.16/29   0.0.0.0   0   32768 ?

```
Total number of prefixes 2
```

Conclusioni

Lo scopo di questo documento è quello di fornire una procedura dettagliata per l'implementazione del routing back-end, con particolare attenzione ai controlli di routing implementati all'interno di BGP per garantire sia la contingenza che la ridondanza.

In sintesi, il spoke 2 e gli altri spoke nella topologia utilizzano lo stesso approccio quando pubblicizzano le loro reti nel dominio BGP. Il controllo più importante del routing in questo scenario è il filtro dell'elenco delle comunità, che assicura che solo le reti all'interno di questa topologia vengano pubblicizzate agli altri peer, evitando la propagazione involontaria della rete.

Inoltre, l'attributo MED Multi-exit Discriminator viene utilizzato per influenzare la selezione delle route per i peer eBGP, garantendo che le route apprese tramite il peer iBGP configurato come HUB primario siano preferite rispetto ai prefissi appresi dall'HUB secondario tramite eBGP.

Effettuando regolazioni della topologia, ad esempio configurando iBGP per l'HUB secondario, è possibile eliminare la necessità di manipolazione MED e route-map in ingresso che capovolgono i tag della community prima di pubblicizzare la stessa rete ad altri spoke.

Informazioni correlate

- Per ulteriore assistenza, contattare TAC. È necessario un contratto di supporto valido:Contatti del supporto Cisco internazionali.
- Puoi anche visitare la Cisco VPN Community per ulteriori informazioni e discussioni sui trend.

Informazioni su questa traduzione

Cisco ha tradotto questo documento utilizzando una combinazione di tecnologie automatiche e umane per offrire ai nostri utenti in tutto il mondo contenuti di supporto nella propria lingua. Si noti che anche la migliore traduzione automatica non sarà mai accurata come quella fornita da un traduttore professionista. Cisco Systems, Inc. non si assume alcuna responsabilità per l’accuratezza di queste traduzioni e consiglia di consultare sempre il documento originale in inglese (disponibile al link fornito).