

BGP の設定

©2021 いっとねっと。

Agenda

- ▶ BGP の基本設定
- ▶ BGP の timer 設定
- ▶ BGP の基本コマンド
 - ▶ update source I/F
 - ▶ next-hop-self

©2021 いっとねっと。

BGP の基本設定

©2021 いっとねっと。

<pre> Step1, BGP process を設定 RT1(config)#router bgp 10000 AS number は 1 - 4294967295 の範囲で指定できる。 </pre>	
<pre> Step2, BGP neighbor を指定する RT1(config)#router bgp 10000 RT1(config-router)#neighbor 192.168.12.2 remote-as 10000 </pre>	
<pre> Step3, BGP neighbor を確認する RT1#show ip bgp summary BGP router identifier 1.1.1.1, local AS number 10000 BGP table version is 3, main routing table version 3 Neighbor V AS MsgRcvd MsgSent TblVer InQ OutQ Up/Down State/PfxRcd 192.168.12.2 4 10000 6 7 3 0 0 00:02:59 0 </pre>	
<pre> Step4, BGP table に広報させたい経路を登録する RT1(config)#router bgp 10000 RT1(config-router)#network 1.1.1.1 mask 255.255.255.255 </pre>	
<pre> Step5, RT1 の BGP table を確認する RT1#show ip bgp BGP table version is 4, local router ID is 1.1.1.1 Status codes: s suppressed, d damped, h history, * valid, > best, i - internal, r RIB-failure, S Stale, m multipath, b backup-path, f RT-Filter, x best-external, a additional-path, c RIB-compressed, Origin codes: i - IGP, e - EGP, ? - incomplete RPKI validation codes: V valid, I invalid, N Not found Network Next Hop Metric LocPrf Weight Path * > 1.1.1.1/32 0.0.0.0 0 0 32768 i </pre>	

BGP は表記の Step で設定していく。
 基本設定は Process の設定、neighbor の指定、広報する経路の登録である。

はじめに Process を立ち上げるため、所属する AS number を指定する。
 今回は検証環境であるため自由な値として global AS の範囲から 10000 を設定している。

その後、neighbor 関係を構築したい BGP Router のアドレスと、その BGP Router が属する AS number を「neighbor command」で指定する。

同じ手順で RT2 側も設定すると、BGP neighbor が確立する。
 BGP neighbor の status は「show ip bgp summary」にて簡易的に確認が可能。

続いて、RT1 は自身の持つ「1.1.1.1/32」のネットワークを RT2 に広報するため、「network command」でこの経路情報を BGP table に登録する。
 設定後に BGP table を確認すると、「1.1.1.1/32」が best path として登録されていることがわかる。

Step6, RT2 の BGP table を確認する

```

RT2#show ip bgp
BGP table version is 4, local router ID is 2.2.2.2
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
               r RIB-failure, S Stale, m multipath, b backup-path, f RT-Filter,
               x best-external, a additional-path, c RIB-compressed,
Origin codes: i - IGP, e - EGP, ? - incomplete
RPKI validation codes: V valid, I invalid, N Not found

   Network          Next Hop          Metric LocPrf Weight Path
  *>i 1.1.1.1/32     192.168.12.1      0     100     0 i

```

Step7, RT2 の RIB を確認する

```

RT2#show ip route
Codes: L - local, C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       I - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
       ia - IS-IS inter area, * - candidate default, U - per-user static route
       o - ODR, P - periodic downloaded static route, H - NHRP, l - LISP
       a - application route
       + - replicated route, % - next hop override

Gateway of last resort is not set

 1.0.0.0/32 is subnetted, 1 subnets
 B   1.1.1.1 [200/0] via 192.168.12.1, 00:02:21
 2.0.0.0/32 is subnetted, 1 subnets
 C   2.2.2.2 is directly connected, Loopback0
 192.168.12.0/24 is variably subnetted, 2 subnets, 2 masks
 C   192.168.12.0/24 is directly connected, Ethernet0/0
 L   192.168.12.2/32 is directly connected, Ethernet0/0

```

AS10000

```

graph LR
    subgraph AS10000
        RT1[RT1  
1.1.1.1/32] ---|192.168.12.0/24| RT2[RT2  
2.2.2.2/32]
    end

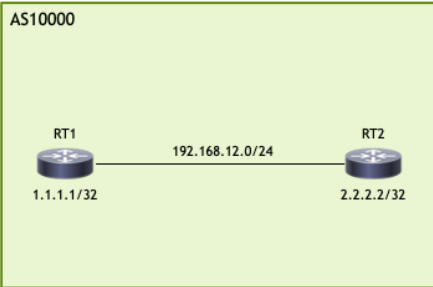
```

RT2 の BGP table を確認すると、RT1 から受信した経路情報が登録されている。
これは best path であるため、RIB へ登録されていることがわかる。

また、今回は iBGP neighbor から学習した経路であるため、AD 値は 200 で登録されている。

BGP の timer 設定

©2021 いっとねっと。

<pre> Step1, BGP timer を確認 RT1#show ip bgp neighbors 192.168.12.2 BGP neighbor is 192.168.12.2, remote AS 10000, internal link BGP version 4, remote router ID 2.2.2.2 BGP state = Established, up for 00:41:22 Last read 00:00:33, last write 00:00:38, hold time is 180, keepalive interval is 60 seconds ... </pre>	
<pre> Step2, BGP timer を変更 RT1(config)#router bgp 10000 RT1(config-router)#neighbor 192.168.12.1 timers 30 90 </pre>	
<pre> Step3, BGP timer を確認 RT1#show ip bgp neighbors 192.168.12.2 BGP neighbor is 192.168.12.2, remote AS 10000, internal link BGP version 4, remote router ID 2.2.2.2 BGP state = Established, up for 00:45:34 Last read 00:00:11, last write 00:00:29, hold time is 180, keepalive interval is 60 seconds Configured hold time is 90, keepalive interval is 30 seconds Minimum holdtime from neighbor is 0 seconds ... </pre>	
<pre> Step4, BGP neighbor を clear RT1#clear ip bgp * *May 29 06:48:28.940: %BGP-5-ADJCHANGE: neighbor 192.168.12.2 Down User reset *May 29 06:48:28.940: %BGP_SESSION-5-ADJCHANGE: neighbor 192.168.12.2 IPv4 Unicast topology base removed from session User reset *May 29 06:48:29.830: %BGP-5-ADJCHANGE: neighbor 192.168.12.2 Up </pre>	
<pre> Step5, BGP timer を確認 RT1#show ip bgp neighbors 192.168.12.2 BGP neighbor is 192.168.12.2, remote AS 10000, internal link BGP version 4, remote router ID 2.2.2.2 BGP state = Established, up for 00:08:14 Last read 00:00:25, last write 00:00:01, hold time is 90, keepalive interval is 30 seconds Configured hold time is 90, keepalive interval is 30 seconds Minimum holdtime from neighbor is 0 seconds ... </pre>	

BGP では default で 60秒ごとに keepalive packet を送信することで、neighbor の死活監視を行なっている。

そして、180秒間 keepalive packet を受信できなければ、neighbor を down させる。

これは「neighbor timer command」で neighbor ごとに timer 値を変更可能。

しかし、すでに up している neighbor に対して設定した場合、表記のように実動作は変更されない。すでに up している neighbor への timer 値を変更する場合は、1度 neighbor を clear し、再度 neighbor 確立 process のなかでこれらの値を negotiation する必要がある。

※これらの値は neighbor router 間で低い値が採用される。

BGP の基本コマンド update source I/F

©2021 いっとねっと。

<pre> Step1, BGP process を設定 RT1(config)#router bgp 10000 </pre>	
<pre> Step2, BGP neighbor を指定する RT1(config)#router bgp 10000 RT1(config-router)#neighbor 2.2.2.2 remote-as 10000 RT1(config-router)#neighbor 2.2.2.2 update-source loopback 0 </pre>	
<pre> Step3, BGP neighbor を確認する RT1#show ip bgp summary BGP router identifier 1.1.1.1, local AS number 10000 BGP table version is 1, main routing table version 1 Neighbor V AS MsgRcvd MsgSent TblVer InQ OutQ Up/Down State/PfxRcd 2.2.2.2 4 10000 5 5 1 0 0 00:01:20 0 </pre>	
<pre> Step4, BGP table に広報させたい経路を登録する RT2(config)#router bgp 10000 RT2(config-router)#network 22.22.22.22 mask 255.255.255.255 </pre>	
<pre> Step5, RT1 の BGP table を確認する RT1#show ip bgp BGP table version is 2, local router ID is 1.1.1.1 Status codes: s suppressed, d damped, h history, * valid, > best, i - internal, r RIB-failure, S Stale, m multipath, b backup-path, f RT-Filter, x best-external, a additional-path, c RIB-compressed, Origin codes: i - IGP, e - EGP, ? - incomplete RPKI validation codes: V valid, I invalid, N Not found Network Next Hop Metric LocPrf Weight Path *>i 22.22.22.22/32 2.2.2.2 0 100 0 i </pre>	

BGP は neighbor command を指定した際、 default では neighbor address に対する出力 I/F が自身の address として指定される。
 例えば、 RT2 は RT1 と neighbor を構築する際、「192.168.12.2」を自身の neighbor address として扱っている。

しかし、物理 I/F のアドレスを用いていると、該当 I/F の故障が発生した際などに疎通性が失われてしまう。
 これらを回避するために、 iBGP では Loopback I/F のアドレスを neighbor address に用いて冗長性を確保することが一般的な実装となっている。

そのため、「neighbor update-source command」を用いて Loopback I/F の address を neighbor address に指定する。
 ※iBGP は default で BGP packet の TTL が 255 であるため問題ないが、 eBGP で同様の実装をする場合は TTL を「neighbor ebgp-multihop command」で変更が必要。

こうして Loopback I/F の address を用いて bgp neighbor を形成すると、受信した経路の next-hop も Loopback I/F の address となっている。

※備考
 Loopback I/F の address を用いて BGP neighbor を形成するためには、 Loopback I/F の address 間で疎通が取れるよう、 static route や IGP (OSPF, EIGRP など) で事前設定が必要である。

BGP の基本コマンド next-hop-self

©2021 いっとねっと。

Step1, RT2 の BGP table を確認する

```
RT2#show ip bgp
BGP table version is 4, local router ID is 2.2.2.2
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
               r RIB-failure, S Stale, m multipath, b backup-path, f RT-Filter,
               x best-external, a additional-path, c RIB-compressed,
Origin codes: i - IGP, e - EGP, ? - incomplete
RPKI validation codes: V valid, I invalid, N Not found
```

Network	Next Hop	Metric	LocPrf	Weight	Path
*> 3.3.3.3/32	192.168.23.3	0		0	20000 1

Step2, RT1 の BGP table を確認する

```
RT1#show ip bgp
BGP table version is 3, local router ID is 1.1.1.1
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
               r RIB-failure, S Stale, m multipath, b backup-path, f RT-Filter,
               x best-external, a additional-path, c RIB-compressed,
Origin codes: i - IGP, e - EGP, ? - incomplete
RPKI validation codes: V valid, I invalid, N Not found
```

Network	Next Hop	Metric	LocPrf	Weight	Path
* i 3.3.3.3/32	192.168.23.3	0	100	0	20000 1

Step3, RT1 の Routing table を確認する

```
RT1#show ip route | begin Gateway
Gateway of last resort is not set
```

```

1.0.0.0/32 is subnetted, 1 subnets
C    1.1.1.1 is directly connected, Loopback0
2.0.0.0/32 is subnetted, 1 subnets
O    2.2.2.2 [110/11] via 192.168.12.2, 00:17:18, Ethernet0/0
192.168.12.0/24 is variably subnetted, 2 subnets, 2 masks
C    192.168.12.0/24 is directly connected, Ethernet0/0
L    192.168.12.1/32 is directly connected, Ethernet0/0

```

この例では、RT1 と RT2 が iBGP neighbor を。RT2 と RT3 が eBGP neighbor を構成している。そして、RT3 は 「3.3.3.3/32」 を RT2 に広報している。

また、iBGP の basic なデザインに従い、iBGP neighbor は Loopback I/F の address を用いて形成している (p9 参照)。

上記条件のもと、RT2 の BGP table を確認すると RT3 から受信した 「3.3.3.3/32」 が best path として登録されている。

BGP では best path である経路を他の neighbor に対して広報するため、RT2 は RT1 にこの経路を広報する。

※iBGP から受信した経路は default で他の iBGP neighbor へは広報しない。

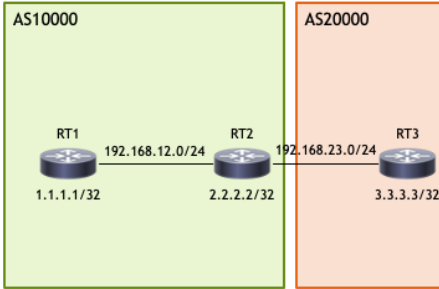
続いて RT1 の BGP table を確認すると、「3.3.3.3/32」 は受信しているものの best path となっていない。

これは何故かということ、Next Hop である 「192.168.23.3」 に対して RT1 が疎通性を持っていない (このアドレスに対する経路が Routing table に存在しない) からである。

このような状態を防ぐためには、以下2つの方法が存在する。

- (1) 「192.168.23.3」 への経路を何らかの手段で RIB に登録する。(Static route や IGP, BGP など)
- (2) 「next-hop-self command」 を使用する

一般的には AS 間のネットワークを AS 内に広報することはしないため、ここでは (2) を採用し動作を確認する。

<pre>Step4, RT2 に next-hop-self を設定する RT2(config)#router bgp 10000 RT2(config-router)#neighbor 1.1.1.1 next-hop-self</pre>	
<pre>Step5, RT1 の BGP table を確認する RT1#show ip bgp BGP table version is 6, local router ID is 1.1.1.1 Status codes: s suppressed, d damped, h history, * valid, > best, i - internal, r RIB-failure, S Stale, m multipath, b backup-path, f RT-Filter, x best-external, a additional-path, c RIB-compressed, Origin codes: i - IGP, e - EGP, ? - incomplete RPKI validation codes: V valid, I invalid, N Not found Network Next Hop Metric LocPrf Weight Path *>i 3.3.3.3/32 2.2.2.2 0 100 0 20000 i</pre>	
<pre>Step6, RT1 の Routing table を確認する RT1#show ip route begin Gateway Gateway of last resort is not set 1.0.0.0/32 is subnetted, 1 subnets C 1.1.1.1 is directly connected, Loopback0 2.0.0.0/32 is subnetted, 1 subnets O 2.2.2.2 [110/11] via 192.168.12.2, 00:26:33, Ethernet0/0 3.0.0.0/32 is subnetted, 1 subnets B 3.3.3.3 [200/0] via 2.2.2.2, 00:00:52 192.168.12.0/24 is variably subnetted, 2 subnets, 2 masks C 192.168.12.0/24 is directly connected, Ethernet0/0 L 192.168.12.1/32 is directly connected, Ethernet0/0</pre>	

「next-hop-self command」を設定すると、eBGP neighbor から受信した経路を対象の neighbor に対して経路広報する際、Next Hop を自身の neighbor address に書き換えることができる。

設定後、RT1 の bgp table を確認すると Next Hop が「192.168.23.3」から「2.2.2.2」に書き換わっていることが確認できた。

RT1 は「2.2.2.2/32」を OSPF で学習しているため、この Next Hop への到達性があることから「3.3.3.3/32」は best path として処理され、無事 RT1 の Routing table に登録されている。