Deploying
BGP Large Communities

Job Snijders
job@ntt.net
NTT Communications
Network Operators Use BGP Communities

- **RFC 1997** style communities have been available for the past 20 years
  - Encodes a 32-bit value displayed as: “16-bit ASN:16-bit value”
  - Designed to simplify Internet routing policies
  - Signals routing information between networks so that an action can be taken
- Broad support in BGP implementations
- Widely deployed and required by network operators for Internet routing

Source: [https://www.us.ntt.net/support/policy/routing.cfm](https://www.us.ntt.net/support/policy/routing.cfm) (AS 2914)

<table>
<thead>
<tr>
<th>Community</th>
<th>Local-pref</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>(default)</td>
<td>120</td>
<td>customer</td>
</tr>
<tr>
<td>65520:nnnn50</td>
<td></td>
<td>only within country &lt;nnnn&gt; (see country list below)</td>
</tr>
<tr>
<td>65530:nnnn50</td>
<td></td>
<td>only within region &lt;nnnn&gt; (see region list below)</td>
</tr>
<tr>
<td>2914:435</td>
<td>50</td>
<td>only beyond the connected country</td>
</tr>
<tr>
<td>2914:436</td>
<td>50</td>
<td>only beyond the connected region</td>
</tr>
<tr>
<td>2914:450</td>
<td>96</td>
<td>customer fallback</td>
</tr>
<tr>
<td>2914:460</td>
<td>98</td>
<td>peer backup</td>
</tr>
<tr>
<td>2914:470</td>
<td>100</td>
<td>peer</td>
</tr>
<tr>
<td>2914:480</td>
<td>110</td>
<td>customer backup</td>
</tr>
<tr>
<td>2914:490</td>
<td>120</td>
<td>customer default</td>
</tr>
<tr>
<td>2914:666</td>
<td></td>
<td>blackhole</td>
</tr>
</tbody>
</table>
Needed RFC 1997 Style Communities, but Larger

• We knew we’d run out of 16-bit ASNs eventually and came up with 32-bit ASNs
  – RIRs started allocating 32-bit ASNs by request in 2007, no distinction between 16-bit and 32-bit ASNs now

• However, you can’t fit a 32-bit value into a 16-bit field
  – Can’t use native 32-bit ASNs with RFC 1997 communities

• Needed an Internet routing communities solution for 32-bit ASNs for almost 10 years
  – Parity and fairness so everyone can use their globally unique ASN
The Solution: **RFC 8092**  
“BGP Large Communities Attribute”

- Idea progressed rapidly from inception in March 2016
- First I-D in September 2016 to RFC publication on February 16, 2017 in just seven months
- Final standard, plus a number of implementation and tools developed as well
- Network operators can test and deploy the new technology now

Cake and photo courtesy of the NTT Communications NOC.

5/22/17
Encoding and Usage

- A unique namespace for all 16-bit and 32-bit ASNs
  - No namespace collisions between ASNs
- Large communities are encoded as a 96-bit quantity and displayed as “32-bit ASN:32-bit value:32-bit value”
- Canonical representation is $Me:$Action:$You
Planning for Large Communities

- The entire network ecosystem needs to support large communities in order to provision, deploy and troubleshoot them
- Ask your vendors and implementers for software support
- Update your tools and provisioning software
- Extend your routing policies, and openly publish this information
- Train your technical staff

Image sources: https://www.sunet.se/blogg/all-i-want-for-christmas-is-large-bgp-communities/
“All i want for christmas is … Large BGP Communities” by Fredrik "Hugge" Korsbäck
Develop a Comprehensive Communities Policy

• Classic RFC 1997 communities will continue to be used together with large communities
  – There’s no flag day to convert, large communities simply provide an additional way to signal information
• Your existing routing policy with classic communities is still valid
• Well-known communities such as “no-advertise”, “no–export”, “blackhole”, etc. are still used
• Extend your policy with large communities that allow network operators to signal the same information as they can with classic communities
## BGP Large Community Examples

<table>
<thead>
<tr>
<th>RFC 1997 (Current)</th>
<th>BGP Large Communities</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>65400:peer-as</td>
<td>2914:65400:peer-as</td>
<td>Do not Advertise to peer-as in North America (NTT)</td>
</tr>
<tr>
<td>43760:peer-as</td>
<td>43760:1:peer-as</td>
<td>Announce a prefix to a certain peer (INEX)</td>
</tr>
<tr>
<td>0:43760</td>
<td>43760:0:peer-as</td>
<td>Prevent announcement of a prefix to a certain peer (INEX)</td>
</tr>
<tr>
<td>65520:nnn</td>
<td>2914:65520:nnn</td>
<td>Lower Local Preference in Country nnn (NTT)</td>
</tr>
<tr>
<td>2914:410</td>
<td>2914:400:10</td>
<td>Route Received From a Peering Partner (NTT)</td>
</tr>
<tr>
<td>2914:420</td>
<td>2914:400:20</td>
<td>Route Received From a Customer (NTT)</td>
</tr>
</tbody>
</table>

- No namespace collisions or use of reserved ASNs
- Enables operators to use 32-bit ASNs in $Me$ and $You$ values
Communities Policy Development

• Provides examples and inspiration for network operators to use large communities
• Also provides many examples on how to develop a communities policy
  – Informational communities
  – Action communities
Informational Communities

• An informational label to mark a route with
  – Its origin: ISO 3166-1 numeric country ID and UM M.49 geographic region
  – Relation or propagation: internal, customer, peer, transit
• Provides information for debugging or capacity planning
• The Global Administrator field is set to the ASN that labels the routes
• Most useful for downstream networks and the Global Administrator itself
### Information Communities Example

<table>
<thead>
<tr>
<th>ISO 3166-1 Country ID</th>
<th>Large Community</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>64497:1:528</td>
<td>64497:1:528</td>
<td>Netherlands</td>
</tr>
<tr>
<td>64497:1:392</td>
<td>64497:2:2</td>
<td>Africa</td>
</tr>
<tr>
<td>64497:1:840</td>
<td>64497:2:9</td>
<td>Oceania</td>
</tr>
<tr>
<td></td>
<td>64497:2:145</td>
<td>Western Asia</td>
</tr>
<tr>
<td></td>
<td>64497:2:150</td>
<td>Europe</td>
</tr>
<tr>
<td></td>
<td>64497:3:1</td>
<td>Internal</td>
</tr>
<tr>
<td></td>
<td>64497:3:2</td>
<td>Customer</td>
</tr>
<tr>
<td></td>
<td>64497:3:3</td>
<td>Peering</td>
</tr>
<tr>
<td></td>
<td>64497:3:4</td>
<td>Transit</td>
</tr>
</tbody>
</table>

- For example, a communities value of “64497:1:528 64497:2:150 64497:3:2” would indicated that is was learned in the Netherlands, in Europe, from a customer.

---

5/22/17

SINOG 4.0, Ljubljana, Slovenia
CDN / Eyeball Example – You do a lot with 32 bits!

<table>
<thead>
<tr>
<th>British Postal Codes (~31 Bits)</th>
<th>or</th>
<th>GPS Coordinates</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Large Community</strong></td>
<td><strong>Postal Code</strong></td>
<td><strong>Large Community</strong></td>
</tr>
<tr>
<td>64497:9:849701135</td>
<td>E1W 1LB (London)</td>
<td>64497:10:1281024</td>
</tr>
<tr>
<td>64497:9:1345374681</td>
<td>M90 1QX (Manchester)</td>
<td></td>
</tr>
</tbody>
</table>

- Location encoding can be used to provide very accurate location information attached to more-specific routes announced to CDN caches
- British postal codes can be encoded by stripping the whitespace and doing a simple base36 to base10 conversion
- GPS coordinates can be encoded with Geohash
  - For example 52.37783, 4.87995 (Amsterdam) encoded with 600 meter precision
  - Python: `import Geohash; Geohash.encode(52.37783, 4.87995, precision=6)`
  - Geohash result: “u173zp”
  - Convert “u173zp” from base32 to base10 = 1281024
Action Communities

• An action label to request that a route be treated in a particular way within an AS
  – Propagation characteristics: export, selective export, no export
  – Local preference: influence ingress traffic within the AS
  – AS Path: influence traffic from outside the AS

• The Global Administrator field is set to the ASN which has defined the functionality of the community
  – Also is the AS that is expected to perform the action

• Most useful for transit providers taking action on behalf of a customer or the Global Administrator
Action Communities Example

- Selective no export
  - ASN based selective no export
  - Location based selective no export
- Selective AS path prepending
  - ASN based selective AS path prepending
  - Location based selective AS path
- Local preference
  - Global local preference
  - Region based local preference

### ASN Based No Export

<table>
<thead>
<tr>
<th>Large Community</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>64497:4:64498</td>
<td>AS 64498</td>
</tr>
<tr>
<td>64497:4:64499</td>
<td>AS 64499</td>
</tr>
<tr>
<td>64497:4:65551</td>
<td>AS 65551</td>
</tr>
</tbody>
</table>

### Location Based No Export

<table>
<thead>
<tr>
<th>Large Community</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>64497:5:528</td>
<td>Netherlands</td>
</tr>
<tr>
<td>64497:5:392</td>
<td>Japan</td>
</tr>
<tr>
<td>64497:5:840</td>
<td>USA</td>
</tr>
</tbody>
</table>
Getting Started With Large Communities

• 2018 is the year of large BGP communities
  – Preparation, testing, training and deployment can take weeks, months or even over a year
  – Start the work now, so you are ready when customers want to use large communities

• Lots of resources are available to help network operators learn about large communities
  – BGP speaker implementations
  – Analysis and ecosystem tools
  – Presentations (http://largebgpcommunities.net/talks/)
  – Documentation for each implementation
  – Configuration examples (http://largebgpcommunities.net/examples/)
Large Communities Beacon Prefixes

- The following prefixes are announced with AS path 2914_15562$
  - 192.147.168.0/24 (looking glass)
  - 2001:67c:208c::/48 (looking glass)
  - BGP Large Community: 15562:1:1

### Cisco IOS Output (Without Large Communities Support)

route-views>show ip bgp 192.147.168.0
BGP routing table entry for 192.147.168.0/24, version 98399100
Paths: (39 available, best #30, table default)
  Not advertised to any peer
  Refresh Epoch 1
  701 2914 15562
  137.39.3.55 from 137.39.3.55 (137.39.3.55)
    Origin IGP, localpref 100, valid, external
    unknown transitive attribute: flag 0xE0 type 0x20 length 0xC
    value 0000 3CCA 0000 0001 0000 0001
    rx pathid: 0, tx pathid: 0

### BIRD Output (With Large Communities Support)

COLOCLUE1 11:06:17 from 94.142.247.3] (100/-) [AS15562i]
Type: BGP unicast univ
BGP.origin: IGP
BGP.as_path: 8283 2914 15562
BGP.next_hop: 94.142.247.3
BGP.med: 0
BGP.local_pref: 100
BGP.community: (2914,410) (2914,1206) (2914,2203) (8283,1)
BGP.large_community: (15562, 1, 1)
# BGP Speaker Implementation Status

<table>
<thead>
<tr>
<th>Implementation</th>
<th>Software</th>
<th>Status</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Arista</td>
<td>EOS</td>
<td>Planned</td>
<td>Feature Requested BUG169446</td>
</tr>
<tr>
<td>Cisco</td>
<td>IOS XE</td>
<td>Planned</td>
<td>16.9.1 (FCS July 2018) source</td>
</tr>
<tr>
<td>Cisco</td>
<td>IOS XR</td>
<td>✔️ Done!</td>
<td>Beta (perhaps in 6.3.2 for real?)</td>
</tr>
<tr>
<td>cz.nic</td>
<td>BIRD</td>
<td>✔️ Done!</td>
<td>BIRD 1.6.3 (commit)</td>
</tr>
<tr>
<td>ExaBGP</td>
<td>ExaBGP</td>
<td>✔️ Done!</td>
<td>PR482</td>
</tr>
<tr>
<td>FreeRangeRouting</td>
<td>frr</td>
<td>✔️ Done!</td>
<td>Issue 46 (commit)</td>
</tr>
<tr>
<td>Juniper</td>
<td>Junos OS</td>
<td>Planned</td>
<td>Second Half 2017 (perhaps 17.3R1?)</td>
</tr>
<tr>
<td>MikroTik</td>
<td>RouterOS</td>
<td>Won’t Implement Until RFC</td>
<td>Feature Requested 2016090522001073</td>
</tr>
<tr>
<td>Nokia</td>
<td>SR OS</td>
<td>Planned</td>
<td>Third Quarter 2017</td>
</tr>
<tr>
<td>nop.hu</td>
<td>freeRouter</td>
<td>✔️ Done!</td>
<td></td>
</tr>
<tr>
<td>OpenBSD</td>
<td>OpenBGPD</td>
<td>✔️ Done!</td>
<td>OpenBSD 6.1 (commit)</td>
</tr>
<tr>
<td>OSGR</td>
<td>GoBGP</td>
<td>✔️ Done!</td>
<td>PR1094</td>
</tr>
<tr>
<td>rtbrick</td>
<td>Fullstack</td>
<td>✔️ Done!</td>
<td>FullStack 17.1</td>
</tr>
<tr>
<td>Quagga</td>
<td>Quagga</td>
<td>✔️ Done!</td>
<td>Quagga 1.2.0 875</td>
</tr>
<tr>
<td>Ubiquiti</td>
<td>EdgeOS</td>
<td>Planned</td>
<td>Internal Enhancement Requested</td>
</tr>
<tr>
<td>VyOS</td>
<td>VyOS</td>
<td>Requested</td>
<td>Feature Requested T143</td>
</tr>
</tbody>
</table>
# Tools and Ecosystem Implementation Status

<table>
<thead>
<tr>
<th>Implementation</th>
<th>Software</th>
<th>Status</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>DE-CIX</td>
<td>pbgpp</td>
<td>✔ Done!</td>
<td>PR16</td>
</tr>
<tr>
<td>FreeBSD</td>
<td>tcpdump</td>
<td>✔ Done!</td>
<td>PR213423</td>
</tr>
<tr>
<td>Marco d’Itri</td>
<td>zebra-dump-parser</td>
<td>✔ Done!</td>
<td>PR3</td>
</tr>
<tr>
<td>OpenBSD</td>
<td>tcpdump</td>
<td>✔ Done!</td>
<td>OpenBSD 6.1 (patch)</td>
</tr>
<tr>
<td>pmacct.net</td>
<td>pmacct</td>
<td>✔ Done!</td>
<td>PR61</td>
</tr>
<tr>
<td>RIPE NCC</td>
<td>bgpdump</td>
<td>✔ Done!</td>
<td>Issue 41 (commit)</td>
</tr>
<tr>
<td>tcpdump.org</td>
<td>tcpdump</td>
<td>✔ Done!</td>
<td>PR543 (commit)</td>
</tr>
<tr>
<td>Yoshiyuki Yamauchi</td>
<td>mrtparse</td>
<td>✔ Done!</td>
<td>PR13</td>
</tr>
<tr>
<td>Wireshark</td>
<td>Dissector</td>
<td>✔ Done!</td>
<td>18172 (patch)</td>
</tr>
</tbody>
</table>

Visit [http://largebgpcommunities.net/implementations/](http://largebgpcommunities.net/implementations/) for the Latest Status
Testing Large Communities

• The BGP Large Communities Playground provides an easy way run several implementations together in a lab environment
• Supports BIRD, ExaBGP, GoBGP, Quagga and pmacct
• Docker images are available
• Use the playground to
  – Become familiar with large communities
  – Test interoperability with your vendor’s BGP implementations
  – Design, configure and verify your new community policies

BGP Large Communities Playground: https://github.com/pierky/bgp-large-communities-playground
Questions?

Presentation created by:

Greg Hankins
Nokia
greg.hankins@nokia.com
@greg_hankins

Job Snijders
NTT Communications
job@ntt.net
@JobSnijders

Visit http://LargeBGPCommunities.net/ for the Latest Info

Reuse of this slide deck is permitted and encouraged!
Configuration and Output Examples
BIRD Configuration

# match
if (8283, 1, 2) ~ bgp_large_community then return true;

# scrub / delete
bgp_large_community.delete([(8283, *, *)]);
bgp_large_community.delete([(8283, 0, 1)]);

# set
bgp_large_community.add((8283, 0, 100));
bgp_large_community.add([(8283, 0, 100), (8283, 2, 333)]);
# match
route-policy set-something
  if large-community matches-any (8283:4:3) then
    set local-preference 120
  endif
end-policy

# scrub / delete
route-policy set-something
  delete large-community in (8283:*::*)
  delete large-community in (8283:4:3)
end-policy

# set
route-policy set-something
  set large-community (8283:45:29) additive
end-policy
Nokia SR OS Configuration

policy-options
  community "set" members "8283:45:29"
  community "match" members "8283:4:3"
  community "delete" members "8283:4:3"

policy-statement "set-something"
  entry 10
    description "match"
    from
      community "match"
    exit
    action accept
    local-preference 120
    exit
  exit
  entry 20
    description "scrub / delete"
    action accept
    community remove "delete"
    exit
  exit
  entry 30
    description "set"
    action accept
    community add "set"
    exit
  exit
  exit
OpenBGPD Configuration

# match
allow from any large-community 8283:1:2
match from any large-community 8283:1:2 set localpref 300
deny to any peer-as neighbor-as \
   large-community 8283:6:neighbor-as

# scrub / delete
match from any set { large-community delete 8283:*:* }
match from any set { large-community delete 8283:1:2 }

# set
match from any set { large-community 8283:1:2 }
match from any set { large-community 8283:1:2 \
   large-community 8283:4034:24824 }
tcpdump 4.9.0 Packet Capture

# ./tcpdump -i eth3 -n -v -c 1 src port 179

tcpdump: listening on eth3, link-type EN10MB (Ethernet), capture size 262144 bytes
16:22:08.992920 IP (tos 0xc0, ttl 64, id 41807, offset 0, flags [DF], proto TCP (6), length 181)
94.142.247.3.179 > 94.142.247.6.33785: Flags [P.], cksum 0xabce (incorrect -> 0x1e40), seq 58743671:58743800, ack 2012368616, win 2270, options [nop,nop,TS val 857977378 ecr 149127175], length 129: BGP
Update Message (2), length: 129
  Origin (1), length: 1, Flags [T]: IGP
  AS Path (2), length: 34, Flags [T]: 38930 1299 3910 721 27065 1554 1555 1501
  Next Hop (3), length: 4, Flags [T]: 94.142.247.3
  Multi Exit Discriminator (4), length: 4, Flags [O]: 0
  Local Preference (5), length: 4, Flags [T]: 100
  Atomic Aggregate (6), length: 0, Flags [T]:
  Aggregator (7), length: 8, Flags [OT]: AS #1501, origin 144.105.202.0
  Community (8), length: 8, Flags [OT]: 1299:20000, 8283:14
  Large Community (32), length: 12, Flags [OTP]:
    8283:6:14
Updated routes:
  136.210.249.0/24
Wireshark 2.3.0 (Prerelease) Packet Capture