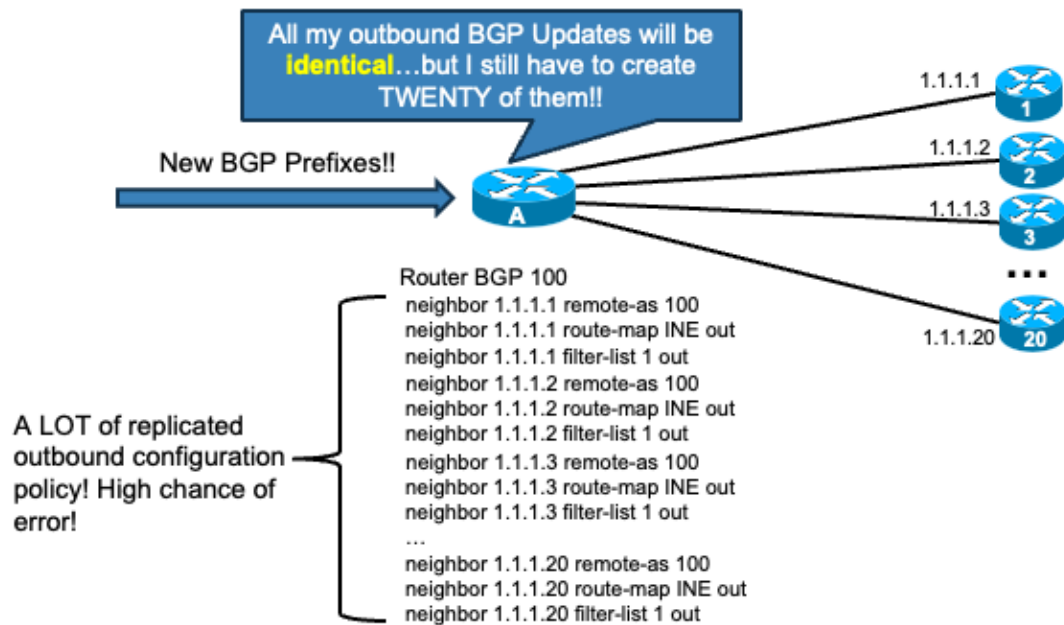




BGP Peer-Groups & Peer-Templates



The Problem To Be Solved

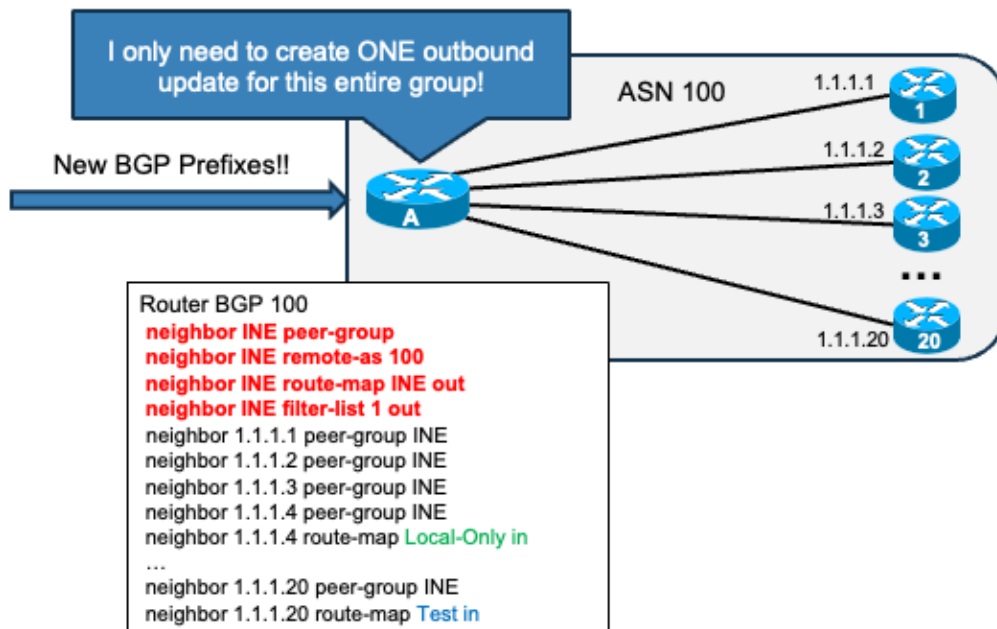


Peer-Group Overview

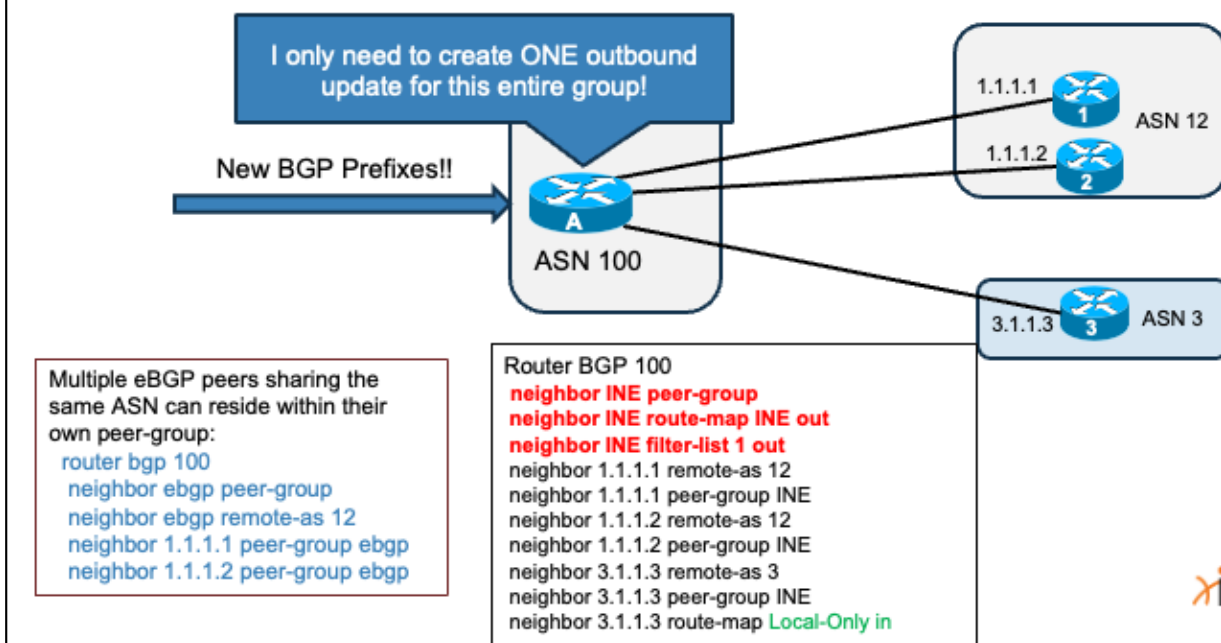
- + BGP neighbors with common **outbound** policy can be placed into the same Peer-Group
- + All members within the same Peer-Group MUST share a common outbound policy
- + Members of a peer group may have different inbound policies (applied on a per-neighbor basis)
- + Members within a shared peer-group may all be iBGP peers or eBGP peers...but not both.
- + "Remote-AS" for iBGP peers configured within the Peer-Group
- + "Remote-AS" for eBGP peers (in different ASNs) configured per peer



iBGP BGP Peer-Groups



eBGP BGP Peer-Groups



- When the “remote-as” keyword will be configured OUTSIDE of the Peer-Group, then the neighbor statement (with remote-as) must be configured BEFORE that neighbor is added to the Peer-Group.

The Problem With Peer-Groups

Hmm...it's 6-months later.
We should change our BGP
passwords as well as our
local-as statements. How
many BGP statements am I
going to need to delete and
rewrite?

I wish these commands were
only in ONE place!

Router BGP 3

```
neighbor Payroll peer-group
neighbor Payroll remote-as 64513
neighbor Payroll password INE111
neighbor Payroll local-as 333
neighbor Marketing peer-group
neighbor Marketing remote-as 64514
neighbor Marketing password INE111
neighbor Marketing local-as 333
neighbor Engineering peer-group
neighbor Engineering remote-as 64515
neighbor Engineering password INE111
neighbor Engineering local-as 333
neighbor Execs peer-group
neighbor Execs remote-as 64516
neighbor Execs password INE111
neighbor Execs local-as 333
neighbor 1.1.1.1 peer-group Payroll
neighbor 1.1.1.2 peer-group Payroll
neighbor 1.1.1.3 peer-group Payroll
neighbor 1.1.1.4 peer-group Marketing
neighbor 1.1.1.4 route-map Local-Only in
```



Another Problem With Peer-Groups

Each of my Peer-Groups has the same three lines of outbound BGP policy.

I need to create TEN more Peer-groups. That's another 30-LINES of Policy!!

I wish these commands were only in ONE place!

Router BGP 3

```
neighbor Payroll peer-group
neighbor Payroll remote-as 64513
neighbor Payroll password INE111
neighbor Payroll filter-list 1 out
neighbor Payroll distribute-list 1 out
neighbor Payroll route-map INE out
neighbor Marketing peer-group
neighbor Marketing remote-as 64514
neighbor Marketing password INE111
neighbor Marketing filter-list 1 out
neighbor Marketing distribute-list 1 out
neighbor Marketing route-map INE out
neighbor Engineering peer-group
neighbor Engineering remote-as 64515
neighbor Engineering password INE111
neighbor Engineering filter-list 1 out
neighbor Engineering distribute-list 1 out
neighbor Engineering route-map INE out
```



BGP Template Overview

- + Cisco routers provide two kinds of BGP Templates
 - + Session Templates
 - + Policy Templates

```
R3(config-router)#template ?  
peer-policy  Template configuration for policy parameters  
peer-session Template configuration for session parameters
```

- + Session templates provide a single place to configure BGP session characteristics that will be applied across a group of peers.
- + Policy templates provide a single place to configure BGP routing policy characteristics that will be applied across a group of peers.
- + Like Peer-Groups, both types of Templates are identified by a meaningful, descriptive name.



BGP Session Templates

```
R3(config-router)#template peer-session Payroll-Sessions
R3(config-router-stmp)#?
BGP peer-policy configuration commands:
  bmp-activate      Activate the BMP monitoring for a BGP peer
  cluster-id        Configure Route-Reflector Cluster-id (peers may
                    reset)
  default           Set a command to its defaults
  description       Neighbor specific description
  disable-connected-check one-hop away EBGP peer using loopback address
  ebgp-multihop     Allow EBGP neighbors not on directly connected
                    networks
  exit-peer-session Exit from template configuration mode
  fall-over         session fall on peer route lost
  ha-mode           high availability mode
  inherit           Inherit a template
  local-as          Specify a local-as number
  log-neighbor-changes Log neighbor up/down and reset reason
  no               Negate a command or set its defaults
  password          Set a password
  path-attribute    BGP optional attribute filtering
  remote-as         Specify a BGP neighbor
  shutdown          Administratively shut down this neighbor
  timers            BGP per neighbor timers
  transport         Transport options
  ttl-security      BGP ttl security check
  update-source     Source of routing updates
  version           Set the BGP version to match a neighbor

R3(config-router-stmp)#
```

Descriptive Name

One template can
"inherit"
characteristics from a
different template.

Session
Characteristics



BGP Policy Templates

One template can
"inherit"
characteristics from a
different template.

```
R3(config-router)#template peer-policy My-Great-Policy
R3(config-router-ptmp)#?
BGP peer-policy configuration commands:
accept-route-legacy-rt Accept Legacy Route Target
additional-paths Negotiate additional paths capabilities with this
neighbor
advertise Advertise to this neighbor
advertise-map specify route-map for conditional advertisement
advertisement-interval Minimum interval between sending BGP routing updates
aigp Enable a AIGP on neighbor
allow-policy Enable the policy support for this IBGP Neighbor
allowas-in Accept as-path with my AS present in it
as-override Override matching AS-number while sending update
capability Advertise capability to the peer
default Set a command to its defaults
default-originate Originate default route to this neighbor
distribute-list Filter updates to/from this neighbor
dmzlink-bw Propagate the DMZ link bandwidth
exit-peer-policy Exit from template configuration mode
filter-list Establish BGP filters
inherit Inherit a template
inter-as-hybrid Inter AS Hybrid mode
internal-vpn-client Stack IBGP-CE Neighbor Path in ATTR_SET for vpn
update
maximum-prefix Maximum number of prefixes accepted from this peer
next-hop-self Disable the next hop calculation for this neighbor
next-hop-unchanged Propagate next hop unchanged for IBGP paths to this
neighbor
no Negate a command or set its defaults
prefix-length-size Packet Level storage size for Prefixes
prefix-list Filter updates to/from this neighbor
remove-private-as Remove private AS number from outbound updates
route-map Apply route map to neighbor
route-reflector-client Configure a neighbor as Route Reflector client
route-server-client Configure a neighbor as Route Server client
send-community Send Community attribute to this neighbor
send-label Send NLRI + MPLS label to this peer
slow-peer Configure slow-peer
soft-reconfiguration Per neighbor soft reconfiguration
soo Site-of-Origin extended community
translate-topology Translate topology id
unsuppress-map Route-map to selectively unsuppress suppressed routes
validation Validation of flowspec paths
weight Set default weight for routes from this neighbor
```

Descriptive Name

Policy
Characteristics



Applying Templates To Peers

- + BGP peers are associated with templates using the “inherit” command;
Neighbor <ip-addr> inherit <peer-policy | peer-session> <name>
- + Peer-groups can also utilize templates because peer-groups support the “inherit” command.



Comparison

Router BGP 3

```

neighbor Payroll peer-group
neighbor Payroll remote-as 64513
neighbor Payroll password INE111
neighbor Payroll filter-list 1 out
neighbor Payroll distribute-list 1 out
neighbor Payroll route-map INE out
neighbor Marketing peer-group
neighbor Marketing remote-as 64514
neighbor Marketing password INE111
neighbor Marketing filter-list 1 out
neighbor Marketing distribute-list 1 out
neighbor Marketing route-map INE out
neighbor Engineering peer-group
neighbor Engineering remote-as 64515
neighbor Engineering password INE111
neighbor Engineering filter-list 1 out
neighbor Engineering distribute-list 1 out
neighbor Engineering route-map INE out

```

```

router bgp 3
!
template peer-policy Common-Policy
  distribute-list 1 out
  route-map INE out
  filter-list 1 out
exit-peer-policy
!
template peer-session Common-Sessions
  password INE111
exit-peer-session
!
template peer-session Payroll
  remote-as 64513
  inherit peer-session Common-Sessions
exit-peer-session
!
template peer-session Marketing
  remote-as 64514
  inherit peer-session Common-Sessions
exit-peer-session
!
template peer-session Engineering
  remote-as 64515
  inherit peer-session Common-Sessions
exit-peer-session
!
bgp log-neighbor-changes
neighbor 1.1.1.1 inherit peer-session Payroll
neighbor 1.1.1.1 inherit peer-policy Common-Policy
neighbor 1.1.1.2 inherit peer-session Marketing
neighbor 1.1.1.2 inherit peer-policy Common-Policy
neighbor 1.1.1.3 inherit peer-session Engineering
neighbor 1.1.1.3 inherit peer-policy Common-Policy

```



- At first glance it doesn't appear that you've saved yourself much configuration work here, but consider the following benefits:
 - This sample configuration is only showing three Peer-Groups...what if there were 23 groups?
 - This sample configuration is only showing three lines of BGP routing policy per Peer-group...what if there were SEVEN lines of policy?
- Also considering that if you need to change something in the future (like the BGP Password) now you only have to do it in ONE place (template peer-session Common Sessions).

Verifying BGP Session-Templates

```
R3#show ip bgp template peer-session
Template:Common-Sessions, index:1
Local policies:0x10, Inherited polices:0x0
  *Inherited by Template Payroll, index= 2
  *Inherited by Template Marketing, index= 3
  *Inherited by Template Engineering, index= 4
Locally configured session commands:
  password is configured
Inherited session commands:

Template:Payroll, index:2
Local policies:0x1, Inherited polices:0x10
This template inherits:
  Common-Sessions index:1 flags:0x0
Locally configured session commands:
  remote-as 64513
Inherited session commands:
  password is configured
```



Verifying BGP Policy-Templates

```
R3#show ip bgp template peer-policy
Template:Common-Policy, index:1.
Local policies:0x2A, Inherited polices:0x0
Local disable policies:0x0, Inherited disable policies:0x0
Locally configured policies:
  distribute-list 1 out
  filter-list 1 out
  route-map INE out
Inherited policies:
```





BGP Dynamic Neighbors



What Problem Is Solved?

- + Under some circumstances, having static BGP neighbor statements can prove problematic;
 - + BGP neighbors with dynamic addresses (DHCP or PPP/IPCPC)
 - + A router with a great quantity of neighbors
 - + A hub router for which additional spokes will be added in the future.
- + The BGP Dynamic Neighbor feature alleviates the need to configure multiple neighbor statements
- + Only configured on ONE side of the BGP connection
 - + The opposite side still requires a static neighbor command.



BGP Dynamic Neighbor Use-Case

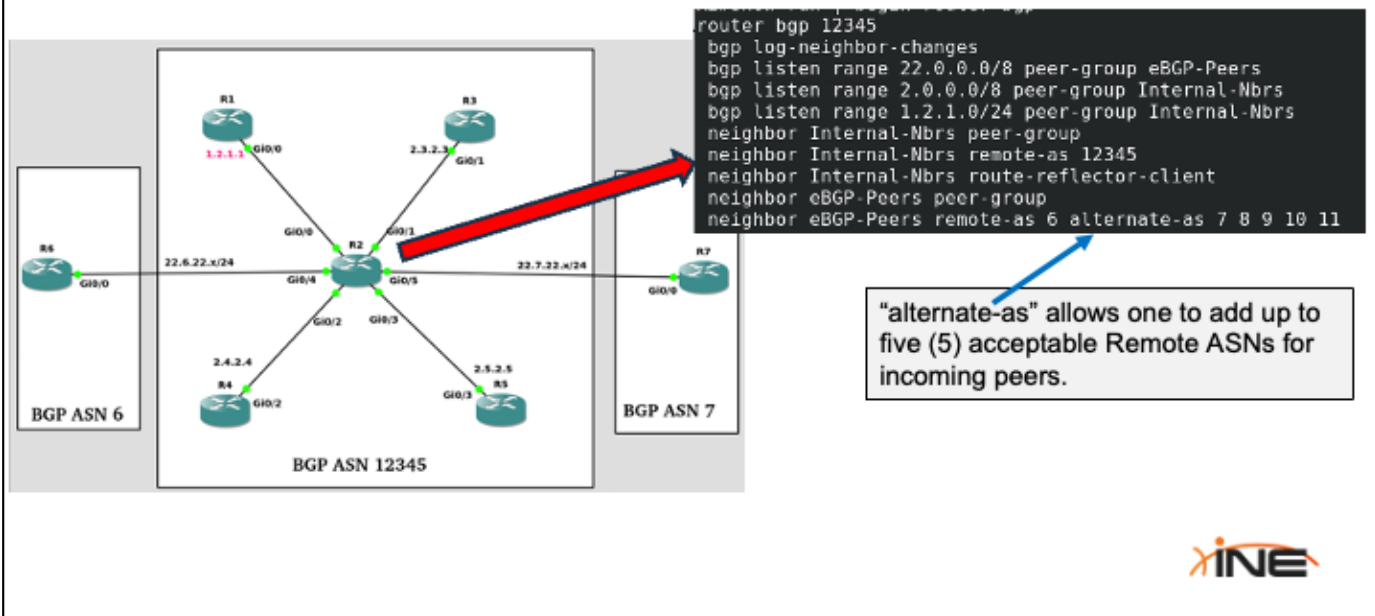


General Principles of BGP Dynamic Neighbors

- + Dynamic neighbors can be eBGP or iBGP peers
- + Local router configured to “listen” to incoming BGP TCP “SYN” messages only from allowed subnet ranges
- + Approved, incoming BGP session requests are paired with a Peer-Group
- + One may exclude certain IP addresses from the subnet range when static peering is required
- + One may limit the quantity of dynamic neighbors allowed



BGP Dynamic Neighbors Configuration Example



- Although in the output of "show running-config" the command "bgp listen range" appears first, it's probably advisable to configure your BGP Peer-Groups first before configuring the "listen range" commands.
- The "alternate-as" keyword can be especially useful when peering to neighbors in different Confederations.

Verification of BGP Dynamic Neighbors

```
*Oct 11 15:48:40.971: %BGP-5-ADJCHANGE: neighbor *2.4.2.4 Up
*Oct 11 15:48:41.082: %BGP-5-ADJCHANGE: neighbor *1.2.1.1 Up
*Oct 11 15:48:41.336: %BGP-5-ADJCHANGE: neighbor *2.5.2.5 Up
*Oct 11 15:48:47.009: %BGP-5-ADJCHANGE: neighbor *2.3.2.3 Up
```

```
R2#show ip bgp neighbor
BGP neighbor is *1.2.1.1, remote AS 12345, internal link
Member of peer-group Internal-Nbrs for session parameters
Belongs to the subnet range group: 1.2.1.0/24
BGP version 4, remote router ID 1.2.1.1
BGP state = Established, up for 00:06:11
```

Dynamic Neighbor Caveats

- + Create static BGP peers prior to implementing bgp "listen" command.
- + Once a peer has been learned dynamically, BGP will not allow you to configure it as a static BGP peer.
- + Once a dynamic neighbor has been formed, one cannot apply most of the BGP "neighbor" keywords against that neighbor.

```
R2(config-router)#neighbor 1.2.1.1 additional-paths send  
% Cannot configure for dynamically created neighbor
```

```
R2(config-router)#neighbor 1.2.1.1 allowas-in  
% Cannot configure for dynamically created neighbor
```



- Some flavors of IOS (such as IOS-XE) provide a "bgp listen block" command so that (prior to the creation of dynamic neighbors) you can exclude certain IP addresses from the listen range...thus allowing you to create static peering statements for those addresses. However mainline IOS does not provide that command.

