

Deploying 32-bit ASNs



ISP Workshops

32-bit ASNs

- Standards documents
 - Description of 32-bit ASNs
 - www.rfc-editor.org/rfc/rfc6793.txt
 - Textual representation
 - www.rfc-editor.org/rfc/rfc5396.txt
 - New extended community
 - www.rfc-editor.org/rfc/rfc5668.txt
- AS 23456 is reserved as interface between 16-bit and 32-bit ASN world

32-bit ASNs – terminology

- 16-bit ASNs
 - Refers to the range 0 to 65535
- 32-bit ASNs
 - Refers to the range 65536 to 4294967295
 - (or the extended range)
- 32-bit ASN pool
 - Refers to the range 0 to 4294967295

Getting a 32-bit ASN

- Nowadays:
 - Standard application process to the RIRs
 - Or via upstream provider
 - Sample RIR policy
 - www.apnic.net/docs/policy/asn-policy.html
- Bootstrap phase from 2007-2010
 - From 1st January 2007
 - 32-bit ASNs were available on request
 - From 1st January 2009
 - 32-bit ASNs were assigned by default
 - 16-bit ASNs were only available on request
 - From 1st January 2010
 - No distinction – ASNs assigned from the 32-bit pool

Representation (1)

- Initially three formats proposed for the 0-4294967295 ASN range :
 - asplain
 - asdot
 - asdot+
- In reality:
 - **Most operators favour traditional plain format**
 - A few prefer dot notation (X.Y):
 - asdot for 65536-4294967295, e.g 2.4
 - asdot+ for 0-4294967295, e.g 0.64513
 - But regular expressions will have to be completely rewritten for asdot and asdot+ !!!

Representation (2)

- ❑ Rewriting regular expressions for asdot/asdot+ notation
- ❑ Example:
 - `^[0-9]+$` matches any ASN (16-bit and asplain)
 - This and equivalents extensively used in BGP multihoming configurations for traffic engineering
- ❑ Equivalent regexp for asdot is:
 - `^([0-9]+)|([0-9]+\.[0-9]+)$`
- ❑ Equivalent regexp for asdot+ is:
 - `^[0-9]+\.[0-9]+$`

Changes

- ❑ 32-bit ASNs are backward compatible with 16-bit ASNs
- ❑ **There is no flag day**
- ❑ You do NOT need to:
 - Throw out your old routers
 - Replace your 16-bit ASN with a 32-bit ASN
- ❑ You do need to be aware that:
 - Your customers will come with 32-bit ASNs
 - ASN 23456 is not a bogon!
 - You will need a router supporting 32-bit ASNs to use a 32-bit ASN locally
- ❑ If you have a proper BGP implementation, 32-bit ASNs will be transported silently across your network

How does it work?

- If local router and remote router support configuration of 32-bit ASNs
 - BGP peering is configured as normal using the 32-bit ASN
- If local router and remote router do not support configuration of 32-bit ASNs
 - BGP peering can only use a 16-bit ASN
- If local router only supports 16-bit ASN and remote router/network has a 32-bit ASN
 - Compatibility mode is initiated...

Compatibility Mode (1)

- ❑ Local router only supports 16-bit ASN and remote router uses 32-bit ASN
- ❑ BGP peering initiated:
 - Remote asks local if 32-bit supported (BGP capability negotiation)
 - When local says “no”, remote then presents AS23456
 - Local needs to be configured to peer with remote using AS23456
- ❑ ⇒ Operator of local router has to configure BGP peering with AS23456

Compatibility Mode (2)

- Local router supports only 16-bit ASNs, peering with router supporting 32-bit ASNs
 - Peering set up with AS23456 (transition AS)

```
router bgp 64510
  neighbor 192.168.2.1 remote-as 23456
  neighbor 192.168.2.1 description eBGP with AS 131076
  neighbor 192.168.2.1 prefix-list AS131076-in in
  neighbor 192.168.2.1 prefix-list AS131076-out out
!
```

- Remote router configures normal BGP peering:

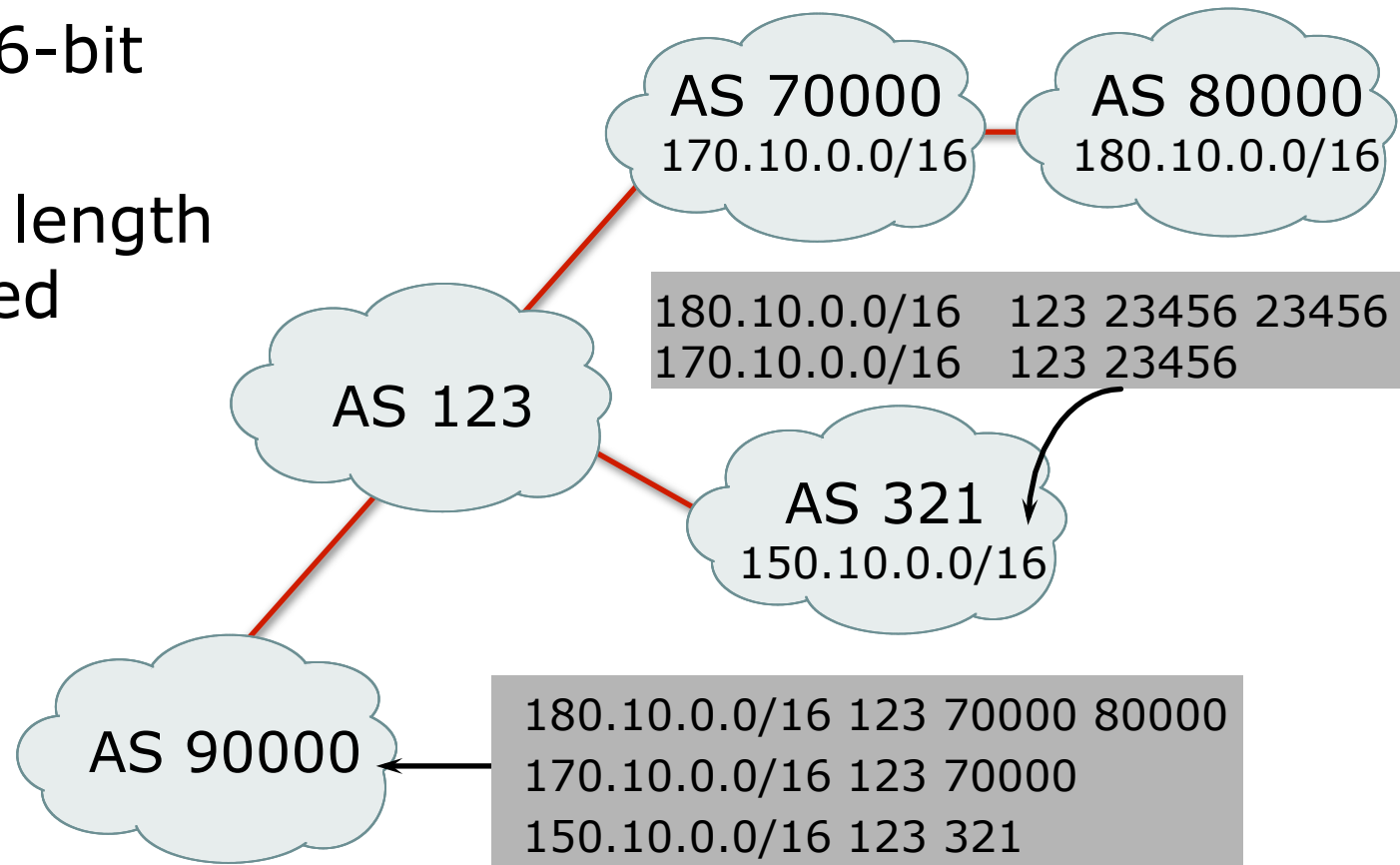
```
router bgp 131076
  neighbor 192.168.2.2 remote-as 64510
  neighbor 192.168.2.2 description eBGP with AS 64510
  neighbor 192.168.2.2 prefix-list AS64510-in in
  neighbor 192.168.2.2 prefix-list AS64510-out out
!
```

Compatibility Mode (3)

- BGP peering initiated (cont):
 - BGP session established using AS23456
 - 32-bit ASN included in a new BGP attribute called AS4_PATH
 - (as opposed to AS_PATH for 16-bit ASNs)
- Result:
 - 16-bit ASN world sees 16-bit ASNs and 23456 standing in for each 32-bit ASN
 - 32-bit ASN world sees 16 and 32-bit ASNs

Example:

- ❑ Internet with 32-bit and 16-bit ASNs
- ❑ AS-PATH length maintained



What has changed?

- Two new BGP attributes:
 - AS4_PATH
 - Carries 32-bit ASN path info
 - AS4_AGGREGATOR
 - Carries 32-bit ASN aggregator info
 - Well-behaved BGP implementations will simply pass these along if they don't understand them
- AS23456 (AS_TRANS)

What do they look like?

- IPv4 prefix originated by AS196613

```
as4-7200#sh ip bgp 145.125.0.0/20
```

```
BGP routing table entry for 145.125.0.0/20, version  
58734
```

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

asplain
format

```
131072 12654 196613
```

```
204.69.200.25 from 204.69.200.25 (204.69.200.25)
```

```
Origin IGP, localpref 100, valid, internal, best
```

- IPv4 prefix originated by AS3.5

```
as4-7200#sh ip bgp 145.125.0.0/20
```

```
BGP routing table entry for 145.125.0.0/20, version  
58734
```

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

asdot
format

```
2.0 12654 3.5
```

```
204.69.200.25 from 204.69.200.25 (204.69.200.25) 14
```

```
Origin IGP, localpref 100, valid, internal, best
```

What do they look like?

- IPv4 prefix originated by AS196613
 - But 16-bit AS world view:

```
BGP-view1>sh ip bgp 145.125.0.0/20
```

```
BGP routing table entry for 145.125.0.0/20, version  
113382
```

```
Paths: (1 available, best #1, table Default-IP-Routing-  
Table)
```

```
23456 12654 23456
```

```
204.69.200.25 from 204.69.200.25 (204.69.200.25)
```

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

Transition
AS

If 32-bit ASN not supported:

- Inability to distinguish between peer ASes using 32-bit ASNs
 - They will all be represented by AS23456
 - Could be problematic for transit provider's policy
 - Workaround: use BGP communities instead
- Inability to distinguish prefix's origin AS
 - How to tell whether origin is real or fake?
 - The real and fake both represented by AS23456
 - **(There should be a better solution here!)**

If 32-bit ASN not supported:

- ❑ Incorrect NetFlow summaries:
 - Prefixes from 32-bit ASNs will all be summarised under AS23456
 - Traffic statistics need to be measured per prefix and aggregated
 - Makes it hard to determine peerability of a neighbouring network
- ❑ Unintended filtering by peers and upstreams:
 - Even if IRR supports 32-bit ASNs, not all tools in use can support
 - ISP may not support 32-bit ASNs which are in the IRR – and don't realise that AS23456 is the transition AS

Implementations (May 2011)

- ❑ Cisco IOS-XR 3.4 onwards
- ❑ Cisco IOS-XE 2.3 onwards
- ❑ Cisco IOS 12.0(32)S12, 12.4(24)T, 12.2SRE, 12.2(33)SXI1 onwards
- ❑ Cisco NX-OS 4.0(1) onwards
- ❑ Quagga 0.99.10 (patches for 0.99.6)
- ❑ OpenBGPd 4.2 (patches for 3.9 & 4.0)
- ❑ Juniper JunOSe 4.1.0 & JunOS 9.1 onwards
- ❑ Redback SEOS
- ❑ Force10 FTOS7.7.1 onwards

- ❑ http://as4.cluepon.net/index.php/Software_Support used to have a complete list

Cisco Routers Supporting 4-byte ASNs

- CRS
 - IOS-XR 3.4 onwards
- GSR
 - IOS-XR 3.4 onwards
 - IOS 12.0(32)S12, 12.0(33)S and 12.0(32)SY8 onwards
- ASR1000
 - IOS-XE 2.3 onwards
- Nexus Switches
 - NX-OS 4.0(1) onwards

Cisco Routers Supporting 4-byte ASNs

- Catalyst 6500
 - IOS 12.2(33)SXI1 onwards
- 7600
 - IOS 12.2(33)SRE1 onwards
- 7200 series
 - IOS 12.0(32)S12, 12.0(33)S, 12.2(33)SRE1, 12.4(24)T, 15.0 onwards
- 7301
 - IOS 12.2(33)SRE1, 12.4(24)T, 15.0 onwards

Cisco Routers Supporting 4-byte ASNs

- 3900/2900/1900 series
 - IOS 15.0 onwards
- 3800/2800/1800/800 series
 - IOS 12.4(24)T and IOS 15.0 onwards
- 3745/3725
 - IOS 12.4(24)T
- AS5350/5400
 - IOS 12.4(24)T and IOS 15.0 onwards

Cisco Routers NOT supporting 4-byte ASNs

- Routers which will never support 4-byte ASNs include:
 - 2500 series
 - 2600 series
 - 3600 series
 - AS5300
 - 7304

Deployment Tips



How to deploy 32-bit ASNs in
the backbone network

Deployment Scenarios

- Typical ISP design is thus:
 - ISIS/OSPF for IGP, carrying loopback and point to point link addresses
 - iBGP mesh (full/RR/Confederation) to carry customer and Internet prefixes
- All routers support 4-byte ASNs:
 - Proceed with iBGP design as normal
- Not all routers support 4-byte ASNs:
 - Three viable options

iBGP options

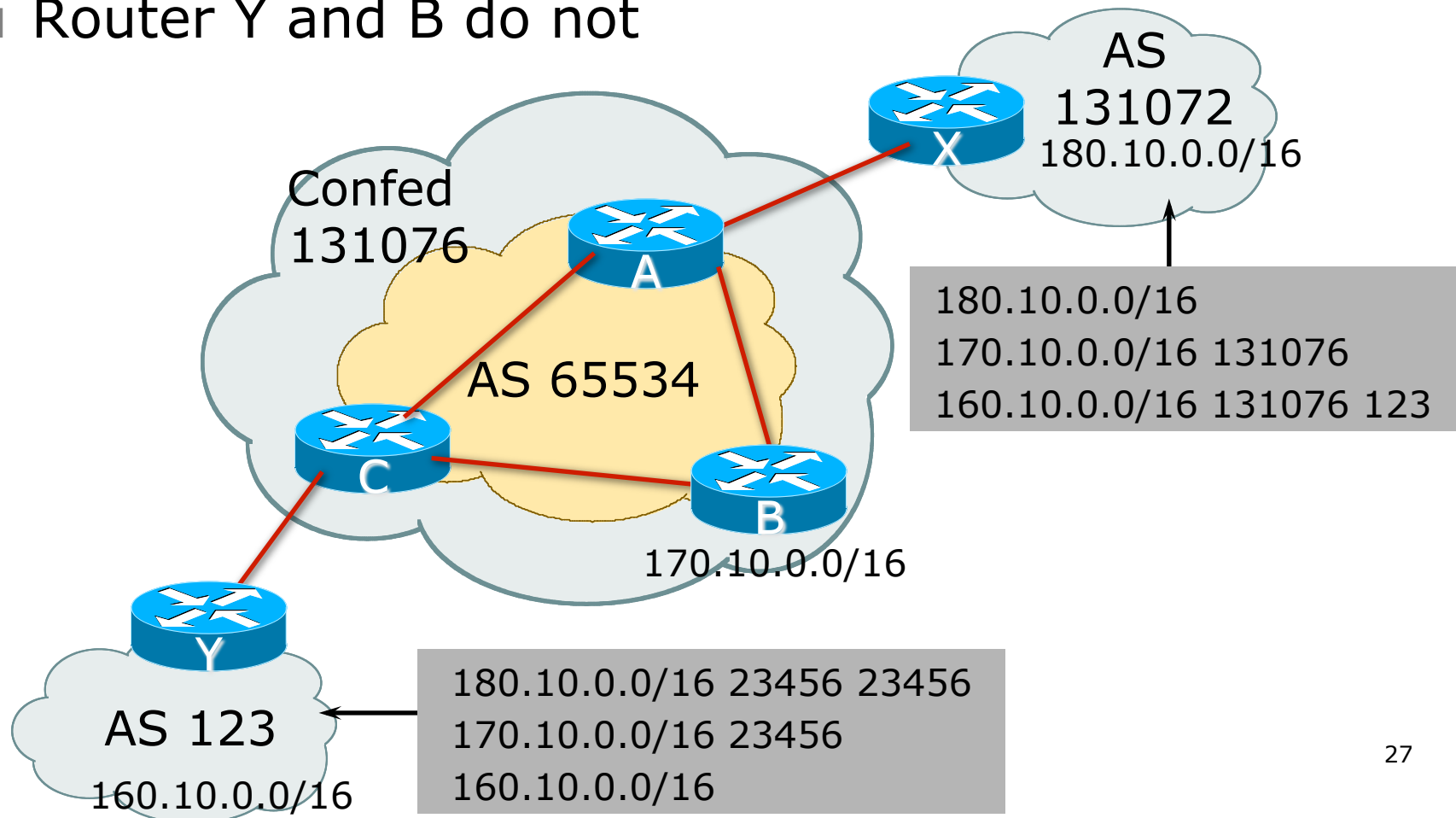
1. Return 4-byte ASN to the RIR and request 2-byte ASN instead
 - Works if RIR is willing to do so
 - Works as long as there are 2-byte ASNs remaining
2. Routers which support 4-byte ASNs run iBGP mesh
 - Routers which do not support 4-byte ASNs either run in private ASN (as a pseudo-customer) or do not run BGP at all
3. The BGP Confederation “hack”

BGP Confederation “hack”

- Useful if only border routers can support 4-byte ASNs
 - Remaining backbone and aggregation routers cannot support 4-byte ASNs
- How?
 - The entire network runs within one private AS
 - The border routers declare to their eBGP neighbours that they are really in 4-byte ASN confederation

Example:

- ❑ Routers X, A and C support 4-byte ASNs
- ❑ Router Y and B do not



The Rules

- ❑ All routers with eBGP neighbours (customer, peer, upstream) must support 4-byte ASNs
- ❑ Remaining routers within the network do not have to support 4-byte ASNs
- ❑ Entire backbone operates in AS65534
 - Or any one private ASN from 64512 to 65534
- ❑ Only the eBGP speaking routers are confederation aware

Router X Configuration

- Router X is in AS131072
 - Supports 4-byte ASNs

```
interface FastEthernet 0/0
  description Link to RouterA
  ip address 192.168.1.1 255.255.255.252
!
router bgp 131072
  neighbor 192.168.1.2 remote 131076
  neighbor 192.168.1.2 eBGP with RouterA
  network 180.10.0.0 mask 255.255.0.0
!
ip route 180.10.0.0 255.255.0.0 null0
```

Router A Configuration

```
interface Loopback 0
  ip address 192.168.2.1 255.255.255.255
!
interface FastEthernet 0/0
  description Link to RouterX
  ip address 192.168.1.2 255.255.255.252
!
router bgp 65534
  bgp confederation identifier 131076
  neighbor 192.168.1.1 remote 131072
  neighbor 192.168.1.1 eBGP with RouterX
  neighbor 192.168.2.2 remote 65534
  neighbor 192.168.2.2 iBGP with RouterB
  neighbor 192.168.2.2 next-hop-self
  neighbor 192.168.2.3 remote 65534
  neighbor 192.168.2.3 iBGP with RouterC
  neighbor 192.168.2.3 next-hop-self
!
```

Router B Configuration

- Router B is in AS65534 (Confederation 131076)
 - Does not support configuration of 4-byte ASNs

```
interface Loopback 0
  ip address 192.168.2.2 255.255.255.255
!
router bgp 65534
  neighbor 192.168.2.1 remote 65534
  neighbor 192.168.2.1 iBGP with RouterA
  neighbor 192.168.2.1 next-hop-self
  neighbor 192.168.2.3 remote 65534
  neighbor 192.168.2.3 iBGP with RouterC
  neighbor 192.168.2.3 next-hop-self
  network 170.10.0.0 mask 255.255.0.0
!
ip route 170.10.0.0 255.255.0.0 null0
```

Router C Configuration

```
interface Loopback 0
  ip address 192.168.2.3 255.255.255.255
!
interface FastEthernet 0/0
  description Link to RouterY
  ip address 192.168.3.1 255.255.255.252
!
router bgp 65534
  bgp confederation identifier 131076
  neighbor 192.168.3.2 remote 123
  neighbor 192.168.3.2 eBGP with RouterY
  neighbor 192.168.2.1 remote 65534
  neighbor 192.168.2.1 iBGP with RouterA
  neighbor 192.168.2.1 next-hop-self
  neighbor 192.168.2.2 remote 65534
  neighbor 192.168.2.2 iBGP with RouterB
  neighbor 192.168.2.2 next-hop-self
!
```


Router Y Configuration

- Router Y is in AS123
 - Does not support configuration of 4-byte ASNs

```
interface FastEthernet 0/0
  description Link to RouterC
  ip address 192.168.3.2 255.255.255.252
!
router bgp 123
  neighbor 192.168.3.1 remote 23456
  neighbor 192.168.3.1 eBGP with RouterC in AS131076
  network 160.10.0.0 mask 255.255.0.0
!
ip route 160.10.0.0 255.255.0.0 null0
```

Commentary

- Only the edge routers, Router A and C, need to know about the confederation and carry the confederation configuration
 - Router B (and any other router participating in the iBGP) believe they are running in AS65534
 - The edge routers will remove the internal AS and present the confederation AS to eBGP neighbours

BGP on Router X

- Router X supports 4-byte ASNs
 - Sees AS131076 and AS123 transit

```
RouterX>sh ip bgp
```

```
BGP table version is 4, local router ID is 192.168.1.1
```

```
Status codes: s suppressed, d damped, h history, * valid, > best, i -  
internal, r RIB-failure, S Stale
```

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

Network	Next Hop	Metric	LocPrf	Weight	Path
*> 160.10.0.0	192.168.1.2			0	131076 123 i
*> 170.10.0.0	192.168.1.2			0	131076 i
*> 180.10.0.0	0.0.0.0	0		32768	i

BGP on Router A

- Router A supports 4-byte ASNs
 - iBGP with B and C, eBGP with X

```
RouterA>sh ip bgp
```

```
BGP table version is 4, local router ID is 192.168.2.1
```

```
Status codes: s suppressed, d damped, h history, * valid, > best, i -  
internal, r RIB-failure, S Stale
```

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

Network	Next Hop	Metric	LocPrf	Weight	Path
*>i160.10.0.0	192.168.2.3	0	100	0	123 i
*>i170.10.0.0	192.168.2.2	0	100	0	i
*> 180.10.0.0	192.168.1.1	0		0	131072 i

BGP on Router B

- ❑ Router B does not support 4-byte ASNs
 - iBGP with B and C; 4-byte ASNs seen as AS23456

```
RouterB>sh ip bgp
```

```
BGP table version is 4, local router ID is 192.168.2.2
```

```
Status codes: s suppressed, d damped, h history, * valid, > best, i -  
internal, r RIB-failure, S Stale
```

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

Network	Next Hop	Metric	LocPrf	Weight	Path
*>i160.10.0.0	192.168.2.3	0	100	0	123 i
*> 170.10.0.0	0.0.0.0	0		32768	i
*>i180.10.0.0	192.168.2.1	0	100	0	23456 i

BGP on Router C

- Router C supports 4-byte ASNs
 - iBGP with A and B, eBGP with Y

```
RouterC>sh ip bgp
```

```
BGP table version is 4, local router ID is 192.168.2.3
```

```
Status codes: s suppressed, d damped, h history, * valid, > best, i -  
internal, r RIB-failure, S Stale
```

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

Network	Next Hop	Metric	LocPrf	Weight	Path
*> 160.10.0.0	192.168.3.2	0		0	123 i
*>i170.10.0.0	192.168.2.2	0	100	0	i
*>i180.10.0.0	192.168.2.1	0	100	0	131072 i

BGP on Router Y

- Router Y does not support 4-byte ASNs
 - eBGP with C; 4-byte ASNs seen as AS23456

```
RouterY>sh ip bgp
```

```
BGP table version is 4, local router ID is 192.168.3.2
```

```
Status codes: s suppressed, d damped, h history, * valid, > best, i -  
internal, r RIB-failure, S Stale
```

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

Network	Next Hop	Metric	LocPrf	Weight	Path
*> 160.10.0.0	0.0.0.0	0		32768	i
*> 170.10.0.0	192.168.3.1			0	23456 i
*> 180.10.0.0	192.168.3.1			0	23456 23456 i

BGP Confederation “hack”

- ❑ Not really a hack, but a workaround so that non-eBGP speaking backbone routers can participate in iBGP using 4-byte ASNs
- ❑ Important:
 - eBGP routers (border and aggregation edge) must support 4-byte ASNs
 - Multiple internal ASNs can work provided that internal AS edge routers (eiBGP speakers) support 4-byte ASNs too; they require:
`bgp confederation identifier <4-byte-ASN>`

Summary

- Deploying 4-byte ASNs can be done three ways:
 - Entire iBGP mesh (upgrading software and/or routers as appropriate)
 - Omit non-4-byte ASN routers from iBGP mesh, or treat them as pseudo BGP customers (like RFC2270)
 - Using the BGP Confederation “hack”
- Or return 4-byte ASN to RIR in exchange for 2-byte ASN (if possible)

Deploying 32-bit ASNs



ISP Workshops