4 Byte AS support in JUNOS

Markku Rantanen
markku@juniper.net

Agenda

- Introduction
- JUNOS implementation details
- Interoperability

Introduction

- 2 byte AS space is getting exhausted pretty soon
  - Unallocated AS Numbers will be exhausted sometime in the period between 2010 and 2016
- IANA has dead line after which AS are always allocated in 4 bytes (Jan 1st 2009)
  - http://www.ripe.net/ripe/policies/proposals/2005-12.html
- IETF standard defines the method of smooth transition maintaining backward compatibility, AS_TRANS etc…

RIRs and 4-byte AS Numbers

- On 1 January 2007 the registry will process applications that specifically request 4-byte only AS Numbers and allocate such AS Numbers as requested by the applicant. In the absence of any specific request for a 4-byte only AS Number, a 2-byte only AS Number will be allocated by the registry

- On 1 January 2009 the registry will process applications that specifically request 2-byte only AS Numbers and allocate such AS Numbers as requested by the applicant. In the absence of any specific request for a 2-byte only AS Number, a 4-byte only AS Number will be allocated by the registry

- On 1 January 2010 the registry will cease to make any distinction between 2-byte only AS Numbers and 4-byte only AS Numbers, and will operate AS Number allocations from an undifferentiated 4- byte AS Number allocation pool

Terminology AS-DOT

- It is proposed to identify 4-byte AS Numbers using a syntax of <high order 16 bit value in decimal>.<low order 16 bit value in decimal>. Accordingly, a 4-byte AS Number of value 65546 (decimal) would be identified as "1.10".
- "2-byte only AS Numbers" refers to AS Numbers in the range 0 – 65535
- ”4-byte only AS Numbers" refers to AS Numbers in the range 1.0 - 65535.65535 (decimal range 65,536 - 4,294,967,295)
- "4-byte AS Numbers" refers to AS Numbers in the range 0.0 - 65535.65535 (decimal range 0 - 4,294,967,295)

Terminology ASPLAIN

- The RIPE NCC assigns and registers 4-byte AS Numbers in ASPLAIN format. ASPLAIN defines the 4-byte AS Number as a basic 32-bit integer.
- It is the current format used to represent 2-byte AS Numbers. Accordingly, the following representation will be used:

- Original 2-byte AS pool : 0 – 65535
  New 4-byte AS pool: 0 – 4294967295

- Terminology
  - "2-byte only AS Numbers" refers to AS Numbers in the range 0 – 65535
  - "4-byte only AS Numbers" refers to AS Numbers in the range 65536 - 4294967295”
  - 4-byte AS Numbers" refers to AS Numbers in the range 0 - 4294967295

Supporting ASPLAIN vs ASDOT (from RIPE)

- A list of advantages of ASPLAIN are as follows:

- ASPLAIN has wide support in the operator community
- IETF is progressing as-representation-01.txt as a standard
- ASPLAIN does not break AS-PATH REGEX
- APNIC reached consensus to adopt ASPLAIN for assignment and representation of 4-byte AS Numbers
- Routers vendors appear to be supporting ASPLAIN, which will require no conversion from allocation to configuration

- Arguments Opposing the ProposalASDOT is more easily remembered
- All existing 4-byte only assignments have been made in ASDOT

4 Byte AS Approach

- As little as possible change in the BGP spec, well…;)
- Backward compatible with 2 Byte BGP speakers
  - Negotiate 4 byte capabilities
  - Automatic adjust behavior with 2 Byte AS speakers
- No ‘Flag Day’ transition
  - Allow 2 Byte “Old Speakers” to continue in mixed 2 byte and 4 byte world
- 4 Byte speakers are New speakers
- 2 Byte speakers are Old speakers
- New AS_TRANS AS 23456
- Two new attributes; AS4_AGGREGATOR and AS4_PATH

Agenda

- Introduction
- **JUNOS implementation details**
- Interoperability

JUNOS and 4 Byte ASN

- JUNOS supports 4 byte AS **officially** from Software release 9.1 onwards
- However JUNOS support for 4 Byte AS since ~8.3 regards in the AS_PATH

- 4 Byte AS is not just simple extension of the AS_PATH fied:
  - Policies, Aggregation etc need to handle 4 Byte AS
  - ASfull vd Asdot
  - Non-routing related, example Sampling and 4 byte
  - CLI…

What all has changed now ?

- AS definition changed globally from 2 byte unsigned short to 4 byte unsigned long
- Within JUNOS, AS numbers are always Generated/ Stored in 4 byte form
- However, in order to be backward compatible, appropriate corrections are made to BGP protocol messages as defined in the 4 byte AS extension RFC
- A hidden command “disable-4byte-as” that makes the router not to declare itself as 4 byte AS capable to a peer. Helpful mostly for testing purposes only
- All other modules/features now use 4 byte AS internally

User Interface Implications

- There are no significant changes in any of the CLI show commands
- The AS numbers are shown in plain 4 byte decimal number format
- “show route receive-protocol bgp” clearly shows both 2 byte AS paths, 4 byte AS paths and the final merged AS path for a particular BGP route
- “show route advertising-protocol shows the AS Paths advertised only in 4byte AS format

User Interface Implications continued…

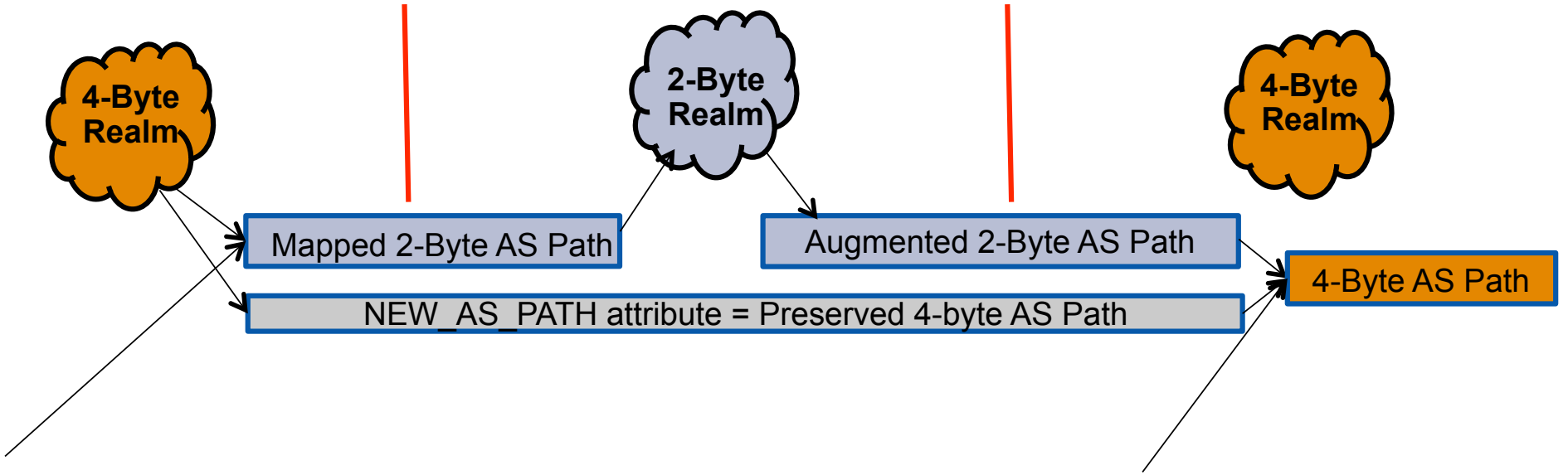
- 4 byte AS extended BGP community is configured by specifying ‘L’ token after the 4 byte AS number
  - target:345578L:12 --> represents 4 byte AS 345578 extended route-target community
  - target:1234:201.1.1.2 --> represents 2 byte AS 1234 extended route-target community
- BGP update trace messages show both 2 byte and 4 byte AS encoded messages

Interaction with the Old BGP speaker (RibIn)

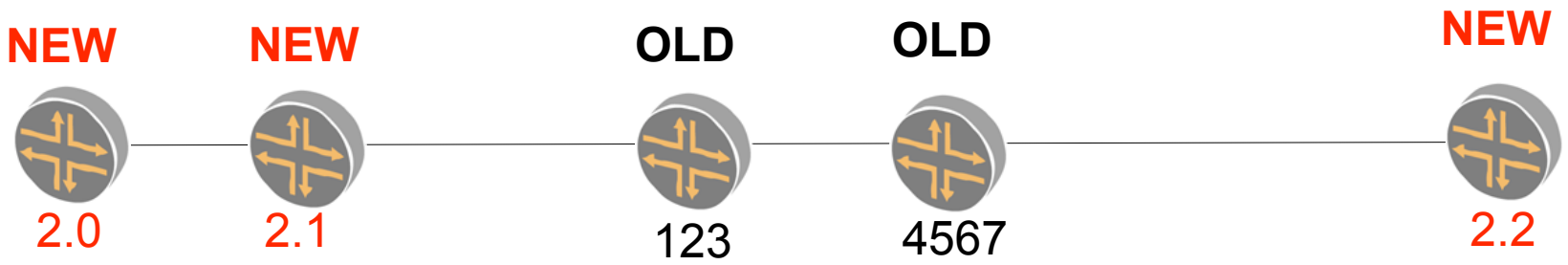
- When messages are received from an Old Speaker, AS Path is converted into 4 byte AS form
- If AS 4 byte path is also present, it is merged with the AS 2 byte path, as defined in the RFC
- The merging process also considers the Aggregator and Aggregator4 attributes received, if any
- For the rest of the modules, received AS_Path is always the merged path. Rest of the details are practically hidden inside the as-path structure

4-Byte AS Transition

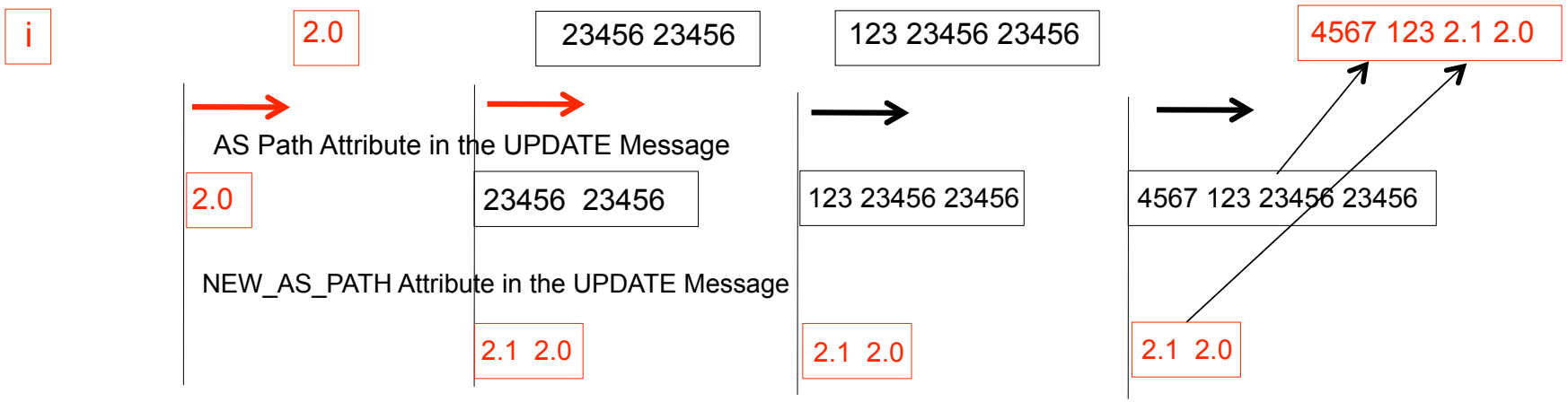
- **Define the NEW / OLD and the OLD / NEW transitions**
- **Preserve all BGP information at the transition interfaces**
  - **Translate** 4-Byte AS Path information into a 2-Byte representation
  - **Tunnel** 4-Byte AS Path information through 2-Byte AS domain as an update attribute



4-Byte AS Example



AS Path in the RIB



Interaction with Old BGP Speaker - RibOut

- No separate peer-group for New and Old BGP speakers for simplicity as well as update generation performance implications
- Update message reformatted just before being sent out to Old BGP speakers as described in the RFC
- None/Minimal copy of messages during reformatting, messages are sent out in a list of fragments, using the writev() socket call
- Mini caching support is added in order to reuse reformatted packets when sending data for members of a peer-group at the same time (In sync peers)

Scaling/Performance implications

- There are some amount of memory implications due to the size of the AS being doubled
  - In a typical Internet scenario, the number of unique AS paths are typically not too a large a number
- There are some amount of CPU implications due to message reformatting required on the BGP outbound side, for sessions with Old BGP speakers (AS_TRANS operation)
- BGP NSR scale implications due to more data processing during RIB exchange in initial rsync part

Implications to other applications

- **Traceroute**
  - The As-path display support is extended to show 4 byte Ases
- **Tcpdump**
  - All 4 byte AS related protocol message decipher is supported
- **Sampled**
  - Currently sampled exports records using Netflow V5 and Netflow V8 format, both of which does not support 4 byte ASes. However, infrastructure has been upgraded to send 4 byte ASes from RPD to Sampled already

Lab Examples

- 4 Byte Supported AS peering
- Policies, Attributes
- Interop 2<>4 Byte AS
- ASfull vs ASdot peering, the translation
- REG_EXP
- Misc…

4 Byte Supported AS peering

- 4 Byte AS router to peer with 2 Byte AS

```
user@chaser# show logical-systems cos_vr routing-options | find auton
autonomous-system 1000000000; <-- Using whole 32bit AS space

[edit]
user@chaser# run show bgp neighbor logical-system cos_vr
Peer: 1.1.1.54+179 AS 1111     Local: 1.1.1.53+49854 AS 1000000000 <-- Peer is 2 byte but supports 4 byte
  Type: External    State: Established    Flags: <ImportEval Sync>
  Last State: OpenConfirm   Last Event: RecvKeepAlive
  Last Error: None
  Export: [ test ]
  Options: <Preference LocalAddress AddressFamily PeerAS Refresh>
  Address families configured: inet-unicast
  Local Address: 1.1.1.53 Holdtime: 90 Preference: 170
  Number of flaps: 0
  Peer ID: 1.1.1.111         Local ID: 110.110.0.1      Active Holdtime: 90
  Keepalive Interval: 30         Peer index: 0
  BFD: disabled, down
  Local Interface: at-0/3/1.0
  NLRI for restart configured on peer: inet-unicast
  NLRI advertised by peer: inet-unicast
  NLRI for this session: inet-unicast
  Peer supports Refresh capability (2)
  Restart time configured on the peer: 120
  Stale routes from peer are kept for: 300
  Restart time requested by this peer: 120
  NLRI that peer supports restart for: inet-unicast
  NLRI that restart is negotiated for: inet-unicast
  NLRI of received end-of-rib markers: inet-unicast
  NLRI of all end-of-rib markers sent: inet-unicast
  Peer supports 4 byte AS extension (peer-as 1111) <-- Negotiation successfully regards 4 Byte support
[…]
```

4 Byte Supported AS peering…

- Peer router, 2 Byte AS…

```
[edit protocols bgp group cos]
user@chaser# show
type external;
traceoptions {
    file cos;
    flag packets detail;
}
local-address 1.1.1.54;
family inet {
    unicast;
}
peer-as 1000000000; <-- 2 Byte As can peer with 4 Byte with no hazzle
neighbor 1.1.1.53;
[…]

[edit protocols bgp group cos]
user@chaser# Aug 29 17:28:04.539850 advertising receiving-speaker only capabilty to neighbor 1.1.1.53
(External AS 1000000000)
Aug 29 17:28:04.539968 bgp_send: sending 59 bytes to 1.1.1.53 (External AS 1000000000)
Aug 29 17:28:04.539998
Aug 29 17:28:04.539998 BGP SEND 1.1.1.54+179 -> 1.1.1.53+49742
Aug 29 17:28:04.540035 BGP SEND message type 1 (Open) length 59
Aug 29 17:28:04.540082 BGP SEND version 4 as 1111 holdtime 90 id 1.1.1.111 parmlen 30
Aug 29 17:28:04.540108 BGP SEND MP capability AFI=1, SAFI=1
Aug 29 17:28:04.540131 BGP SEND Refresh capability, code=128
Aug 29 17:28:04.540345 BGP SEND Refresh capability, code=2
Aug 29 17:28:04.540372 BGP SEND Restart capability, code=64, time=120, flags=
Aug 29 17:28:04.540397 BGP SEND 4 Byte AS-Path capability (65), as_num 1111
Aug 29 17:28:04.540583 bgp_send: sending 19 bytes to 1.1.1.53 (External AS 1000000000)
Aug 29 17:28:04.540614
```

Policies and attributes

- No differences compared with 2 Byte AS

```
[edit logical-routers cos]
user@chaser# show routing-options rib inet.0
aggregate {
    route 110.110.0.0/16 {
        as-path {
            origin igp;
            aggregator 1000000000 1.1.1.4; <-- Policies and other features been modified to handle 4 byte AS
        }
    }
[...]

[edit logical-routers cos]
user@chaser# show policy-options policy-statement test
term 1 {
    from {
        protocol aggregate;
        route-filter 110.110.0.0/16 exact;
    }
    then {
        as-path-prepend "1000000000 1000000000 1000000000";
        accept;
    }
}
```

Policies and attributes

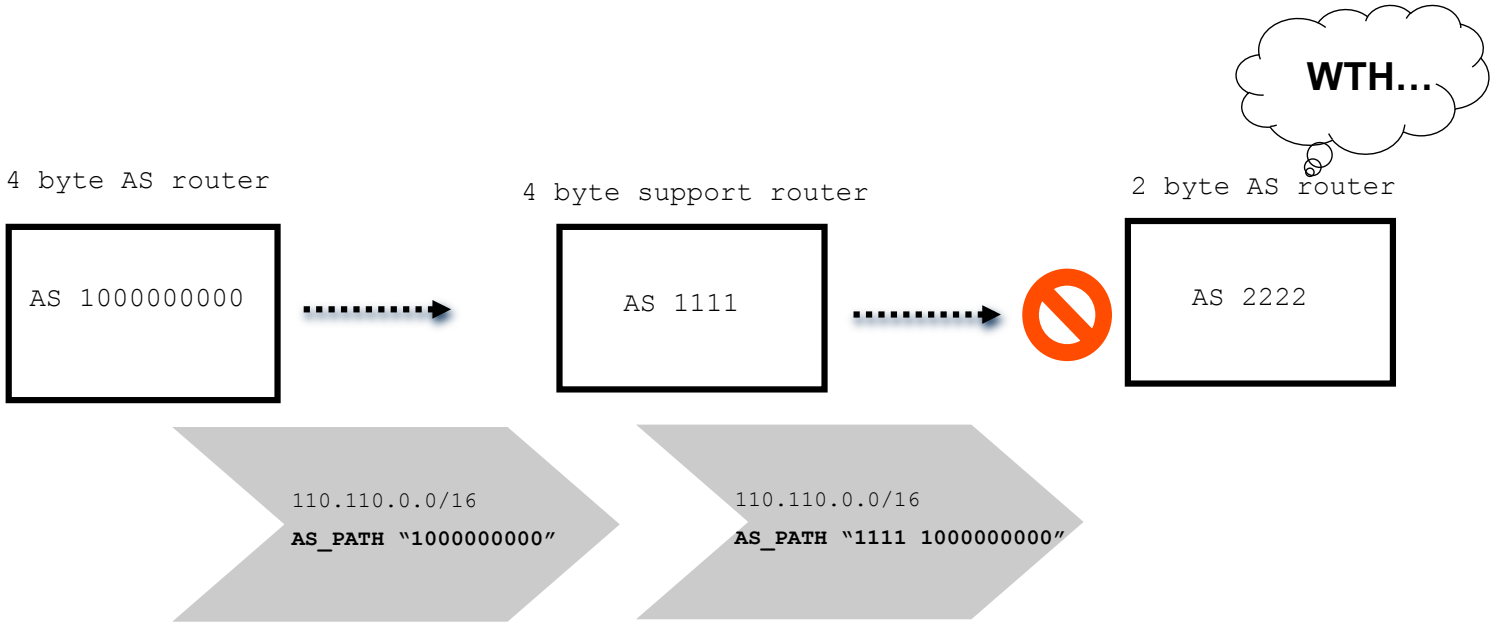
- No differences compared with 2 Byte AS

```
<-- Adj-Rib-In/out can handle 4 byte AS

[edit logical-routers cos]
user@chaser# run show route advertising-protocol bgp 1.1.1.54 logical-router cos

inet.0: 31 destinations, 31 routes (31 active, 0 holddown, 0 hidden)
  Prefix                  Nexthop              MED     Lclpref    AS path
* 110.110.0.0/16          Self                                    1000000000 1000000000 1000000000
[1000000000] I


<-- Receiving router can use regular-axpressions with 4 byte AS

[edit]
user@chaser# run show route table inet.0 aspath-regex "1000000000.*"

inet.0: 49 destinations, 50 routes (46 active, 0 holddown, 3 hidden)
+ = Active Route, - = Last Active, * = Both

110.110.0.0/16     *[BGP/170] 00:13:40, localpref 100
                      AS path: 1000000000 1000000000 1000000000 1000000000 I
                    > to 1.1.1.53 via at-0/3/0.0
```

Interop 2<>4 Byte AS

- But If indirect router does not support 4 byte AS, this value needs to be converted in the AS_PATH field

- The AS_TRANS (23456) is used which is the reserved AS number can handle this this.



Interop 2<>4 Byte AS

- Peer router clearly notifies NO support for 4 byte AS

```
user@chaser# run show bgp neighbor 1.1.1.10

Peer: 1.1.1.10+57007 AS 1111   Local: 1.1.1.111+179 AS 1111
  Type: Internal    State: Established    Flags: <Sync>
  Last State: OpenConfirm   Last Event: RecvKeepAlive
  Last Error: Cease
  Export: [ direct statics ]
  Options: <Preference LocalAddress AddressFamily Multipath Rib-group Refresh>
  Options: <MtuDiscovery>
  Address families configured: inet-unicast inet-multicast inet-vpn-unicast inet-vpn-multicast l2vpn-signaling
inet-labeled-unicast inet-mvpn
  Local Address: 1.1.1.111 Holdtime: 90 Preference: 170
  Number of flaps: 1
  Last flap event: Stop
  Error: 'Cease' Sent: 1 Recv: 0
  Peer ID: 1.1.1.10         Local ID: 1.1.1.111        Active Holdtime: 90
  Keepalive Interval: 30         Peer index: 0
  BFD: disabled, down
  NLRI advertised by peer: inet-unicast inet-multicast inet-vpn-unicast inet-vpn-multicast l2vpn-signaling inet-
labeled-unicast inet6-vpn-unicast inet-mvpn
  NLRI for this session: inet-unicast inet-multicast inet-vpn-unicast inet-vpn-multicast l2vpn-signaling inet-
labeled-unicast inet-mvpn
  Peer supports Refresh capability (2)
  Peer does not support 4 byte AS extension <--
[...]
```

Interop 2<>4 Byte AS…

- Router below 110.110/16… The 4 byte AS information needs to be translated.

```
user@chaser# run show route 110.110/16 detail

inet.0: 49 destinations, 50 routes (46 active, 0 holddown, 3 hidden)
110.110.0.0/16 (1 entry, 1 announced)
        *BGP    Preference: 170/-101
                Next hop type: Router, Next hop index: 686
                Next-hop reference count: 10
                Source: 1.1.1.53
                Next hop: 1.1.1.53 via at-0/3/0.0, selected
                State: <Active Ext>
                Local AS:  1111 Peer AS: 1000000000
                Age: 15:25
                Task: BGP_1000000000.1.1.1.53+64018
                Announcement bits (4): 0-KRT 8-BGP RT Background 9-Resolve tree 2 10-Resolve tree 3
                AS path: AS2 PA[4]: AS_TRANS AS_TRANS AS_TRANS AS_TRANS
                AS path: AS4 PA[4]: 1000000000 1000000000 1000000000 1000000000
                AS path: Merged[4]: 1000000000 1000000000 1000000000 1000000000 I Aggregator (Ignored):
AS_TRANS 1.1.1.4
                AS path:  Aggregator4: 1000000000 1.1.1.4
                Localpref: 100
                Router ID: 1.1.1.4
[...]
```

Interop 2<>4 Byte AS…

- The result below shown for 110.110/16 on the 2 byte AS only router

```
user@junos_distribution# run show route 110/8 detail

inet.0: 59 destinations, 60 routes (55 active, 0 holddown, 4 hidden)
Restart Complete
110.110.0.0/16 (1 entry, 1 announced)
        *BGP    Preference: 170/-101
                Next hop type: Indirect
                Next-hop reference count: 4
                Source: 1.1.1.111
                Next hop type: Router, Next hop index: 774
                Next hop: 1.1.2.189 via so-0/0/2.0, selected
                Protocol next hop: 1.1.1.53
                Indirect next hop: 8ca21d4 262177
                State: <Active Int Ext>
                Local AS:  1111 Peer AS:  1111
                Age: 22:02      Metric2: 20
                Task: BGP_1111.1.1.1.111+179
                Announcement bits (5): 0-KRT 2-RT 10-BGP RT Background 11-Resolve tree 2 12-Resolve tree 3
                AS path: 23456 23456 23456 23456 I () Aggregator: 23456 1.1.1.4 <--
                AS path:  Unrecognized Attributes: 32 bytes
                AS path:  Attr flags e0 code 11: 02 04 3b 9a ca 00 3b 9a ca 00 3b 9a ca 00 3b 9a ca 00
                AS path:  Attr flags e0 code 12: 3b 9a ca 00 01 01 01 04
                Localpref: 100
                Router ID: 1.1.1.111
[...]c]
```

AS full vs AS dot peering, the translation

- **According to the RFC, several ways to display/set the AS**
  - AS Full (65536)
  - AS Dot (1.1)
- **JUNOS supports both - both for configuration and display**

```
user@chaser> show route protocol bgp table inet.0
[...]
110.110.0.0/16     *[BGP/170] 00:00:47, localpref 100
                      AS path: 1000000000 1000000000 1000000000 1000000000 I <--
                    > to 1.1.1.53 via at-0/3/0.0
[...]


user@chaser# set routing-options autonomous-system ?
Possible completions:
  <as_number>           Autonomous system number in plain number or 'higher 16bits'.'Lower 16 bits' (asdot notation) format
  asdot-notation        Use AS-Dot notation to display true 4 byte AS numbers
  loops                 Maximum number of times this AS can be in an AS path


[edit]
user@chaser# set routing-options autonomous-system asdot-notation

[edit]
user@chaser# commit
commit complete


[edit]
user@chaser# run show route protocol bgp table inet.0
[...]
110.110.0.0/16     *[BGP/170] 00:02:57, localpref 100
                      AS path: 15258.51712 15258.51712 15258.51712 15258.51712 I <--
                    > to 1.1.1.53 via at-0/3/0.0
[...
```

ASfull vs ASdot…

- You can however mix ASdot or ASfull for the config…
- The knob earlier shown is just for show commands output

```
[edit protocols bgp group cos]
user@chaser# show
type external;
traceoptions {
    file cos;
    flag packets detail;
}
local-address 1.1.1.54;
family inet {
    unicast;
}
export 4byte;
peer-as 1000000000; <--
neighbor 1.1.1.53
```

```
[edit protocols bgp group cos]
user@chaser# show
type external;
traceoptions {
    file cos;
    flag packets detail;
}
local-address 1.1.1.54;
family inet {
    unicast;
}
export 4byte;
peer-as 15258.51712; <--
neighbor 1.1.1.53
```

ASfull vs ASdot…

- Regular-expressions do not have to be ASDOT format
- The \. is a pest, trust me… people who have been forced to use that will soon change their mind and ask for AS full :)

```
user@junos_distribution> ... protocol bgp aspath-regex "15258\.51712"

inet.0: 51 destinations, 59 routes (46 active, 0 holddown, 5 hidden)
Restart Complete
+ = Active Route, - = Last Active, * = Both

110.110.0.0/16     *[BGP/170] 14:45:16, localpref 100, from 1.1.1.111
                      AS path: 15258.51712 I
                    > to 1.1.2.189 via so-0/0/2.0

[edit]
user@junos_distribution# run show route table inet.0 aspath-regex "1000000000.*"

inet.0: 51 destinations, 67 routes (46 active, 15 holddown, 5 hidden)
Restart Pending: BGP IS-IS(TED done) RSVP LDP
+ = Active Route, - = Last Active, * = Both

110.110.0.0/16     *[BGP/170] 00:00:58, localpref 100, from 1.1.1.111
                      AS path: 15258.51712 I
                    > to 1.1.2.189 via so-0/0/2.0
```

Misc

- Example v5/v8 Sampling

```
[edit]
user@chaser# show forwarding-options sampling
input {
    family inet {
        rate 1;
        run-length 0;
        max-packets-per-second 1000;
    }
}
output {
    file filename sampling files 10 size 1m world-readable stamp;
    cflowd 192.168.0.200 {
        port 2000;
        source-address 192.168.0.102;
        version 8;
        local-dump;
        autonomous-system-type peer;
        aggregation {
            source-destination-prefix;
        }
    }
}
```

Misc

- Example v5/v8 Sampling …

```
<-- Peer NLRI generated to fake an AS_PATH

[…]
term 4 {
    from {
        protocol static;
        route-filter 110.110.3.0/24 exact;
    }
    then {
        as-path-expand "10.7 71 100 171 111"; <== 655367 71 100 171 111
        accept;
    }
}
[…]

<-- Seen in the Local-RIB

user@chaser# run show route 110.110.3.0

inet.0: 41 destinations, 42 routes (39 active, 0 holddown, 2 hidden)
+ = Active Route, - = Last Active, * = Both

110.110.3.0/24     *[BGP/170] 00:02:16, localpref 100
                      AS path: 1000000000 655367 71 100 171 111 I
                    > to 1.1.1.53 via at-0/3/0.0
```

Misc

- Example v5/v8 Sampling …

```
<--Hmmm

[edit logical-systems cos_vr policy-options policy-statement test]
user@chaser# Sep  9 16:22:47 v5 flow entry
Sep  9 16:22:47    Src addr: 110.110.3.1
Sep  9 16:22:47    Dst addr: 1.1.1.10
Sep  9 16:22:47    Nhop addr: 1.1.2.188
Sep  9 16:22:47    Input interface: 0
Sep  9 16:22:47    Output interface: 0
Sep  9 16:22:47    Pkts in flow: 59
Sep  9 16:22:47    Bytes in flow: 4956
Sep  9 16:22:47    Start time of flow: 8583425
Sep  9 16:22:47    End time of flow: 8641447
Sep  9 16:22:47    Src port: 0
Sep  9 16:22:47    Dst port: 0
Sep  9 16:22:47    TCP flags: 0x0
Sep  9 16:22:47    IP proto num: 1
Sep  9 16:22:47    TOS: 0x0
Sep  9 16:22:47    Src AS: 0
Sep  9 16:22:47    Dst AS: 1111
Sep  9 16:22:47    Src netmask len: 24
Sep  9 16:22:47    Dst netmask len: 32
```

```
edit]
user@chaser#
*** sampled ***
Sep  9 14:09:31 v5 flow entry
Sep  9 14:09:31    Src addr: 110.110.2.1
Sep  9 14:09:31    Dst addr: 1.1.1.10
Sep  9 14:09:31    Nhop addr: 1.1.2.188
Sep  9 14:09:31    Input interface: 0
Sep  9 14:09:31    Output interface: 0
Sep  9 14:09:31    Pkts in flow: 148
Sep  9 14:09:31    Bytes in flow: 12432
Sep  9 14:09:31    Start time of flow: 645989
Sep  9 14:09:31    End time of flow: 646254
Sep  9 14:09:31    Src port: 0
Sep  9 14:09:31    Dst port: 0
Sep  9 14:09:31    TCP flags: 0x0
Sep  9 14:09:31    IP proto num: 1
Sep  9 14:09:31    TOS: 0x0
Sep  9 14:09:31    Src AS: 23456
Sep  9 14:09:31    Dst AS: 1111
Sep  9 14:09:31    Src netmask len: 24
Sep  9 14:09:31    Dst netmask len: 32
```

Pls note this is fixed,

its just an example of complications along the 2<>4 Byte AS road…

Agenda

- Introduction
- RFC/IETF Standards
- JUNOS implementation details
- **Interoperability**

What about company who needs to interoperate with C ?

```
NOTE:
ASPLAIN 65547 -> ASDOT 1.11
ASPLAIN 65546 -> ASDOT 1.10
```

<-- IOSX

```
router bgp 1.10
 address-family ipv4 unicast
 !
 neighbor 193.10.255.102
  remote-as 1.11 <--
  update-source
TenGigE0/2/0/0
  address-family ipv4 unicast
  !
 !
!
```

<-- JUNOS

```
[edit]
user@t320b_re1# show routing-
options
[...]
router-id 193.10.255.5;
autonomous-system 65547;


[edit]
user@t320b_re1# show
protocols bgp
mtu-discovery;
log-updown;
tcp-mss 4096;
[...]
group iosz {
    type external;
    peer-as 65546; <--
    neighbor 193.10.255.101;
}
```

interop with C…

<-- show commands IOSX

```
RP/0/9/CPU0:gsr1#sh ip bgp sum
BGP router identifier 193.10.255.1, local AS number 1.10
BGP generic scan interval 60 secs
BGP table state: Active
Table ID: 0xe0000000
BGP main routing table version 1
BGP scan interval 60 secs

BGP is operating in STANDALONE mode.

Process          RecvTblVer    bRIB/RIB  LabelVer  ImportVer  SendTblVer
Speaker                   1           1         1          1           1
Neighbor        Spk    AS MsgRcvd MsgSent   TblVer  InQ OutQ Up/Down  St/PfxRcd
193.10.255.102    0  1.11      16      14        1    0    0 00:06:00        0!

RP/0/9/CPU0:gsr1#sh ip bgp nei 193.10.255.102
BGP neighbor is 193.10.255.102
 Remote AS 1.11, local AS 1.10, external link
 Remote router ID 193.10.255.5
  BGP state = Established, up for 00:06:26
  Last read 00:00:00, hold time is 90, keepalive interval is 30 seconds
  Precedence: internet
  Neighbor capabilities:
    Route refresh: advertised and received
    4-byte AS: advertised and received
    Address family IPv4 Unicast: advertised and received
  Received 17 messages, 0 notifications, 0 in queue
  Sent 15 messages, 0 notifications, 0 in queue
  Minimum time between advertisement runs is 30 seconds

 For Address Family: IPv4 Unicast
  BGP neighbor version 1
  Update group: 0.2
  eBGP neighbor with no inbound or outbound policy; defaults to 'drop'
  Route refresh request: received 0, sent 0
  0 accepted prefixes, 0 are bestpaths
  Prefix advertised 0, suppressed 0, withdrawn 0, maximum limit 524288
  Threshold for warning message 75%
  An EoR was received during read-only mode

  Connections established 1; dropped 0
  Last reset 00:00:00
```

Summary

Something to remember

- ASPLAIN is Your friend ☺

Something to read

- RFC 4893

http://tools.ietf.org/html/rfc4893

- Geoff Huston has a good presentation

http://www.nanog.org/mtg-0702/presentations/huston.pdf

- Check the 2 Byte AS Number exhaustion time

http://www.potaroo.net/tools/asns