BGPSEC Protcol Error Handling

IETF 87

Berlin, Germany

Wednesday, 31 Jul 2013

Friday, 2 Aug 2013

BGPSEC-Protocol Errors Noted

- MUST/MUST NOT in various places
  – Error handing mentioned in some of them
- Section 5.2: Validation Algorithm
  – "properly formed" – steps 1-5
  – dealing with signatures and validity

BGPSEC Protocol Error Response

- Section 5.2:
  - If any of these checks identify an error in the BGPSEC_Path attribute, then the implementation should notify the operator that an error has occurred and treat the update in a manner consistent with other BGP errors (i.e., following RFC 4271[2] or any future updates to that document).
- Section 4.3:
  - Such an error is treated in exactly the same way as receipt of a non-BGPSEC update message containing an AS_CONFED_SEQUENCE from a peer that is not a member of the same AS confederation.

BGP Error Handling

- RFC4271 – Usually NOTIFICATION message is sent with code/subcode and the BGP connection is closed
- RFC5065 - error handling for confederations (AS_CONFED_SEQUENCE presence from non-confed member and vice versa) – another NOTIFICATION subcode
- IDR draft draft-ietf-idr-error-handling-04.txt
  - Three possible responses
  - "session reset”
  - "treat-as-withdraw"
  - "attribute discard”

What to Do?

- Should response be more specific?
- Response Choice:
  - Follow RFC4271 (Notification with code/subcode and close session)?
  - Follow idr error handling draft?
    - ***If so, which errors get which response?***

BGPSEC-Protocol Draft Error Handling

- “Properly formed” checks in Section 5.2
  1. check syntactic correctness
  2. each Signature_Block has one Signature for each Secure Path segment
  3. check that AS_PATH not present
  4. for non-confed-member neighbor, ensure Confed_Sequence flag is not set
  5. pcount=0 but peer is not configured to use pcount=0
- “treat the update in a manner consistent with other BGP errors”

BGPSEC-Protocol Draft Error Handling

- Section 5.2: unable to find key – mark Signature_Block Not Valid
- Section 5.2: no supported signature – consider unsigned
- Section 5.2: no matching covering ROA for AS: mark route Not Valid
- Section 5.2: signature fails, mark Signature_Block Not Valid
- Section 5.2: no valid Signature_Block, mark route Not Valid
- Section 4.3 (forward ref to 5.2)
  - Checks if confed bit set when neighbor not in confed
  - No text for vice versa case: i.e., confed bit not set from confed member

BGPSEC-Protocol Draft Error Handling

- Error handling for MUST NOT?
- E.g., Section 4.2

If a BGPSEC router has received only a non-BGPSEC update message (without the BGPSEC_Path attribute), …. then it MUST NOT attach any BGPSEC_Path attribute to the corresponding update being propagated.

- If neighbor messes up and produces a BGPSEC_PATH attribute anyway, and strips the AS_PATH, will that be caught?