draft-huston-sidr-roa-validation-00

Geoff Huston
George Michaelson
SIDR WG, IETF 71
March 2008
Ooh yummy! A -zero Draft!

- Early thoughts about validation
  - During the transitional phases
    - Any security mechanism will be partially deployed
    - Looking for reasonable behaviours which will permit validation of origination in a ‘mixed’ world with useful properties
      - Minimum change to BGP (or none!)
      - Minimum disruption of the non-security-aware world

- More work required..
  - Is the basic model heading in a useful direction?
What does ‘validation’ really mean at this time?

- Older thinking/language
  
  IF <no ROA> || <ROA ‘fails’> THEN
  
  <its bogus, get rid of it>

- In early deployment, its not entirely black-and-white state
  
  - What if this is just one of those ‘not yet’ networks?
    - More specific flag in ROA adds complications
    - Validation failure can be for a number of reasons
      - Don’t we have to try and take account of this?

- (re)define application of ROA to take account of
  
  - Missing origination authority possibilities
  
  - transitional state issues
  
  - existing BGP route selection processes
The Good, The Bad and the Ugly

Possible outcomes when matching a collection of ROAs to a route object:

- **Good**
  - **Exact match** (same prefix, same origin AS, valid ROA)
  - **Covering match** (covering prefix, same origin AS, “more specifics permitted” ROA Flag ON, valid ROA)

- **Bad**
  - **Exact mismatch** (same prefix, different origin AS, valid ROA)

- **Ugly** (Not clearly bad)
  - **ROA missing** (partial deployment case)
  - **Covering mismatch** (covering prefix, mismatch on origin AS, “more specifics permitted”, valid ROA – could be related to partial deployment case)
  - **Covering failure** (covering prefix, same origin AS, “more specifics permitted” ROA Flag ON, invalid ROA - could be related to partial deployment case)
  - **Exact Failure** (same prefix, same origin AS, invalid ROA – expired authority or DOS attack?)
Apply Outcomes to BGP localpref

- Follow RFC4271 sec 9.1.1
  - “calculation of degree of preference”
  - Reject unacceptables, but RANK everything else by ROA preference order
- More specific ROAs apply highest localpref
- Un-secured routes apply lower localpref
Prefer the best...

.. But take the least-worst?

- Never take something (actively) revoked
  - On a CRL
- Never take something patently bogus
  - Bad ASN.1, bad signature
- What about provably good crypto state?
  - Useful to take things which aren’t *quite* as good as an exact match, but aren’t *evil*
- Do not reject originations with no authorization
  - Not (yet) demonstrably bad
And After the Transition?

- Can make the ‘intermediate’ states map to the same preference and treat as **EVIL**
- Can begin to apply ROA-based rejection more widely
  - Actively decline non-secured routes
Open Issues

- Is validation before, during or after RFC4271 9.1.1 Adj-RIB-In?
  - And what about state change of ROA info even when no AS change?

- Lifetimes of ROA validity state?

- Can lessons of flap-damping be applied?

- ROA validation per-AS?

- Possible DoS:
  - make someone reject routes based a detectably bad ROA for a valid AS/pfx..