Overview of draft-ietf-sidr-roa-format-01.txt

Matt Lepinski

BBN Technologies

Presentation Outline

- Review of route origination security
- Review of high level ROA design
- Changes from -00
- Open Issues
  - Matching of ROAs to EE certs
  - Matching of ROAs to route advertisements
- Questions

Route Origination Security

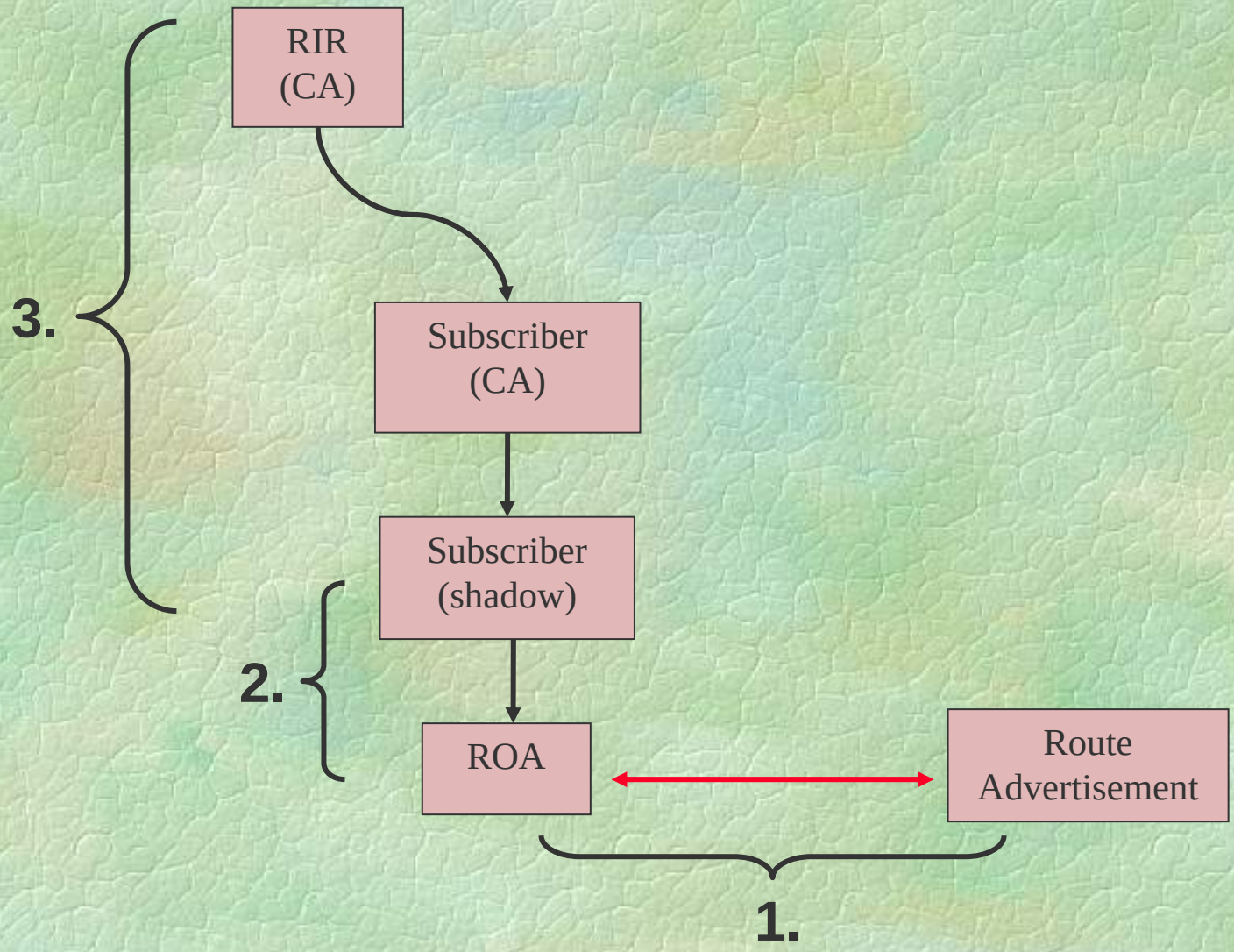
- One goal of this PKI is to enable ISPs to verify route origination assertions in BGP UPDATE messages
- To support this goal, each address space holder needs to digitally sign one or more objects that identify each AS authorized to advertise routes on behalf of the address space holder
- We call the object a route origination authorization (ROA)
- An address space holder issues a distinct ROA to each ISP he wants to advertise all or a portion of his address space
- Since each ISP is an address space holder, it would sign one or more ROAs (one per AS number) authorizing itself to advertise the addresses it holds

Validity of a Route Origination

A route origination is valid if:

1. The route advertisement “matches” a ROA
   - AS number “matches”
   - IP address prefix “matches” the NLRI
2. The ROA “matches” an EE certificate
   - Signature is valid
   - IP addresses “match”
3. The EE certificate is valid as described in:
   draft-ietf-sidr-res-certs

Route Origination Validation



ROA Design

- A ROA has four major data elements, encapsulated in a CMS signed data object
  - A version number
  - One of more address prefixes, corresponding to the NLRI that the ROA signer authorizes for origination by one or more ISPs
  - A flag indicating the semantics for matching the NLRI to the prefixes in the ROA
  - An AS number of an ISP authorized to originate routes to the above list of prefixes
- We use the CMS format to represent a signed ROA, as this format is well supported in open source software

Changes Since -00

- OID bug fix in the CMS profile
- ROAs now include only IP address prefixes and not IP address ranges (as in RFC 3779)
- Added a Boolean *ExactMatch* flag to the ROA to indicate semantics for NLRI to ROA "matching"
  - TRUE means the AS may ***only*** advertise the prefixes that appear in the ROA
  - FALSE means the AS may advertise the prefixes in the ROA or any more specific prefixes
- Added a section describing how a ROA is validated

ROA Format

```
RouteOriginAttestation ::= SEQUENCE {
  version [0] INTEGER DEFAULT 0,
  -- this is the ROA version #
  asID   ASID,
  exactMatch BOOLEAN,
  ipAddrBlocks ROAIPAddrBlocks }

ASID ::= INTEGER

ROAIPAddrBlocks ::= SEQUENCE of ROAIPAddressFamily

ROAIPAddressFamily ::= SEQUENCE {
  addressFamily OCTET STRING (SIZE (2..3)),
  addresses SEQUENCE OF IPAddress }
-- Only two address families: IPv4 and IPv6

IPAddress ::= BIT STRING
```

Issue: Matching ROA to EE Cert

- EE Certificates use IP address ranges for compact representation of multiple prefixes
- ROAs include only prefixes and not ranges
- For example:

  ROA includes: 11.0.0.0/8 and 12.0.0.0/8

  EE Certificate: 11.0.0.0 - 12.255.255.255

- Proposed Solution:

  Add text that clarifies this point and provides more detailed instructions for performing the comparison

Issue: Matching ROA to NLRI (1/2)

- The -00 version of the draft specified that the NLRI in an advertisement must exactly match a prefix in the ROA
- Feedback on the list was that this is too restrictive
- The suggestion was made to introduce the following four options (taken from RPSL):
  - Exact Match
  - Any more specific prefix
  - Any more specific prefix of length exactly X
  - Any more specific prefix of length between X and Y

Issue: Matching ROA to NLRI (2/2)

- Not possible to use the RPSL syntax given the current usage of RFC 3779 ASN.1
- Analysis of the RIPE IRR indicates that only two of the RPSL semantics are widely used:

  Exact Match and *Any More Specific Prefix*
  - *Any Prefix of Length X* was used by fewer than 5 ASes
  - *Prefixes between Length X and Y* was used in situations where *Any More Specific Prefix* would also work

- Proposed Solution:

  Keep the current flag to express the *Exact Match* and *Any More Specific Prefix* semantics

Thank You