ROA Contents & Format Proposal

Brian Weis
Overview

• An informal study was conducted considering
  – ROA Contents
    • Based on Steve Kent’s earlier presentations
  – ROA Format
    • Design Considerations
    • Three possible formats studied
ROA Contents

- Data necessary to have a fully specified ROA:
  - **Object type** (i.e., “ROA”)
    - Plan ahead for other object types (e.g., signed AS policy)
  - **Object version** (i.e., “1”)
  - **Address prefix(es)**
    - May be a subset of addresses in the EE set?
  - **AS number(s)** authorized to advertise the address prefixes in the ROA
  - **Validity interval** (i.e., start/stop times)
    - May be shorter than the EE validity period in an emergency?
  - **Signature list**
    - Including certificate pointers and other necessary parameters
ROA Design Considerations

- **Design Considerations**
  - **Size.** Distribution through a network protocol may be advantageous in some cases
  - **Extensibility.** Format should allow standards-track additions to the format.
  - **Open source tool availability.** Tool availability is crucial to adoption.
  - **Clearly defined canonicalization rules.** Needed to support reliable digital signatures
ROA Format

• Three data formats considered
  – Simple TLVs
    • Header + Type-Length-Value attributes representation of the data
  – ASN.1
  – XML
TLV Format

- **Header**
  - Object Type
  - Version
  - Object Length

- **Attributes**

<table>
<thead>
<tr>
<th>Type</th>
<th>Len</th>
<th>Auth AS</th>
</tr>
</thead>
<tbody>
<tr>
<td>Type</td>
<td>Len</td>
<td>Validity</td>
</tr>
<tr>
<td>Type</td>
<td>Len</td>
<td>Signature</td>
</tr>
</tbody>
</table>
ASN.1 Format

• Imports many definitions from RFC 3280 and RFC 3779
  – No reason to re-specify common fields
  – ASN.1 open source tools already contain support for these definitions

• New ASN.1 definitions create an ROA framework for imported definitions.
ASN.1 Format (Abridged)

so OBJECT IDENTIFIER ::= { joint-iso-ccitt(2) ds(5) 40 }
so-roa OBJECT IDENTIFIER ::= { so 1 }

SO ::= SEQUENCE {
    sObject                 SObject,
    signatures              SEQUENCE OF Signatures }

SObject ::= SEQUENCE {
    signedObjectType        Type,
    version                 [0] EXPLICIT SOVersion DEFAULT v1,
    validity                Validity,
    ipAddrBlocks            SEQUENCE OF IPAddressFamily,
    asIdentifiers           SEQUENCE OF ASIdentifiers }

Type ::= INTEGER { roa(1) }
SOVersion ::= INTEGER { v1(0) }

Signatures ::= SEQUENCE {
    certificatePointer      AuthorityKeyIdentifier,
    authorityInfo           AuthorityInfoAccessSyntax,
    signatureAlgorithm      AlgorithmIdentifier,
    signatureValue          BIT STRING }
XML Format

• Basic ROA Document Type Definition (.dtd file) is simple

• The digital signature specification is taken from RFC 3275
  – Signature XML elements are added during the signature process
XML ROA

<!ELEMENT SO (sObject)>  
<!ELEMENT sObject (signedObjectType, version, validity,  
ipAddrBlocks*, asIdentifiers*)>  
<!ELEMENT signedObjectType (#PCDATA)>  
<!ELEMENT version (#PCDATA)>  
<!ELEMENT validity (notBefore,notAfter)>  
<!ELEMENT notBefore (uctTime)>  
<!ELEMENT notAfter (uctTime)>  
<!ELEMENT uctTime (#PCDATA)>  
<!ELEMENT ipAddrBlocks (IPAddressFamily,addressPrefix)>  
<!ELEMENT IPAddressFamily (addressFamily)>  
<!ELEMENT addressFamily (#PCDATA)>  
<!ELEMENT addressPrefix (#PCDATA)>  
<!ELEMENT asIdentifiers (id*)>  
<!ELEMENT id (#PCDATA)>
Sample ROA

• Comparison of an ROA in the three formats
  – Type: ROA
  – Version: 1
  – Two prefixes
  – Two authorized ASes
  – One signature (RSA 1024-bit)
# Design Considerations

<table>
<thead>
<tr>
<th></th>
<th>TLV</th>
<th>ASN.1</th>
<th>XML</th>
</tr>
</thead>
<tbody>
<tr>
<td>Size (bytes) of sample ROA</td>
<td>286</td>
<td>445</td>
<td>1654</td>
</tr>
<tr>
<td>Extensible</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Open Source Tools?</td>
<td>No</td>
<td>Yes (asn1c)</td>
<td>Yes (XMLSec)</td>
</tr>
<tr>
<td>Canonicalization?</td>
<td>TBD</td>
<td>Yes (DER)</td>
<td>Yes (RFC 3275)</td>
</tr>
</tbody>
</table>
Conclusion: ASN.1 is the best compromise

• While DER is substantially larger than a simple TLV format (35% larger) it remains manageable.
• ASN.1 is easily extensible.
• Canonicalization rules are well defined.
• Use of ASN.1 has some synergy with Resource Certificates.
• Open source ASN.1 compiler tools appear to hide much of ASN.1 required knowledge from tools developers.
Next steps

• Get consensus on the content & format
• Generate a -00 draft describing the ROA prior to IETF 68