Individual Submission
Internet-Draft
Intended status: Informational
Expires: October 24, 2008

G. Huston
T. Manderson
G. Michaelson
APNIC

April 22, 2008

A Profile for Bogon Origin Attestations (BOAs)
draft-huston-sidr-bogons-01.txt

Status of this Memo

Copyright Notice

Abstract

This document defines a standard profile for Bogon Origin Attestations (BOAs). A BOA is a digitally signed object that provides a means of verifying that an IP address block holder has not authorized any Autonomous System (AS) to originate routes that are equivalent to any of the addresses listed in the BOA, and also provides a means of verifying that BGP speaker is not using an AS as a BGP speaker without appropriate authority to use that AS. The

proposed application of BOAs is intended to fit within the requirements for adding security measures to inter-domain routing, including the ability to support incremental and piecemeal deployment of such measures, and does not require any changes to the specification of BGP.

Table of Contents

1. Introduction 3
2. Basic Format 3
2.1. Signed-Data Content Type 4
2.1.1. version 4
2.1.2. digestAlgorithms 4
2.1.3. encapContentInfo 4
2.1.4. certificates 6
2.1.5. crls 6
2.1.6. signerInfo 6
3. BOA Validation 9
4. BOA Use Practices 11
5. BOA Interpretation 11
6. Security Considerations 12
7. IANA Considerations 12
8. Acknowledgments 12
9. Normative References 12
Authors' Addresses 13
Intellectual Property and Copyright Statements 15

1. Introduction

This document defines an application of the Resource Public Key Infrastructure (RPKI) to validate the attestations of Internet Registries that certain addresses are currently neither allocated nor assigned to any party, and any appearance of such addresses or ASes in a routing advertisement in the Border Gateway Protocol (BGP) [RFC4271] should be considered an invalid use of such addresses or ASes.

The RPKI is based on Resource Certificates. Resource Certificates are X.509 certificates that conform to the PKIX profile [RFC3280], and to the extensions for IP addresses and AS identifiers [RFC3779]. A Resource Certificate describes an action by an Issuer that binds a list of IP address blocks and Autonomous System (AS) numbers to the Subject of a certificate, identified by the unique association of the Subject's private key with the public key contained in the Resource Certificate. The PKI is structured such that each current Resource Certificate matches a current resource allocation or assignment. This is described in [ID.ietf-sidr-arch].

BOAs can be regarded as a logical opposite of a Route Origin Authorization (ROA) [ID.ietf-sidr-roa-format], and allows a resource holder to explicitly list those IP addresses and ASes that are denoted by the holder as not validly appearing in any routing advertisement, and to make this attestation in a manner that a relying party can validate under the framework of the RPKI.

A BOA is a digitally signed object that makes use of Cryptographic Message Syntax (CMS) [RFC3852] as a standard encapsulation format. CMS was chosen to take advantage of existing open source software available for processing messages in this format.

2. Basic Format

Using CMS syntax, a BOA is a type of signed-data object. The general format of a CMS object is:

```
ContentInfo ::= SEQUENCE {
  contentType ContentType,
  content [0] EXPLICIT ANY DEFINED BY contentType }

ContentType ::= OBJECT IDENTIFIER
```

2.1. Signed-Data Content Type

According to the CMS specification, The signed-data content type shall have ASN.1 type SignedData:

```
SignedData ::= SEQUENCE {
  version CMSVersion,
  digestAlgorithms DigestAlgorithmIdentifiers,
  encapContentInfo EncapsulatedContentInfo,
  certificates [0] IMPLICIT CertificateSet OPTIONAL,
  crls [1] IMPLICIT RevocationInfoChoices OPTIONAL,
  signerInfos SignerInfos }

DigestAlgorithmIdentifiers ::= SET OF DigestAlgorithmIdentifier

SignerInfos ::= SET OF SignerInfo
```

2.1.1. version

The version is the syntax version number. It MUST be 3, corresponding to the signerInfo structure having version number 3.

2.1.2. digestAlgorithms

The digestAlgorithms set MUST include only SHA-256, the OID for which is 2.16.840.1.101.3.4.2.1. [RFC4055] It MUST NOT contain any other algorithms.

2.1.3. encapContentInfo

encapContentInfo is the signed content, consisting of a content type identifier and the content itself.

```
EncapsulatedContentInfo ::= SEQUENCE {
  eContentType ContentType,
  eContent [0] EXPLICIT OCTET STRING OPTIONAL }

ContentType ::= OBJECT IDENTIFIER
```

2.1.3.1. eContentType

The ContentType for a BOA is defined as id-ct-rpkiBOA, and has the numerical value of 1.2.840.113549.1.9.16.1.[TBS]. [This value has to be assigned via an OID registration.]

```
id-smime OBJECT IDENTIFIER ::= { iso(1) member-body(2) us(840)
    rsadsi(113549) pkcs(1) pkcs9(9) 16 }

id-ct OBJECT INDENTIFIER ::= { id-smime 1 }

id-ct-rpkiBOA OBJECT IDENTIFIER ::= { id-ct [TBS] }
```

2.1.3.2. eContent

The content of a BOA identifies a list of one or more ASes and a list of one or more IP address prefixes that are asserted to be "bogons" and, accordingly, BOAs are intended to act as a constraint on the routing system to signal that no route object that that relates to these ASes or IP addresses should be interpreted as representing a valid routing attestation. A BOA is formally defined as:

```
id-ct-rpkiBOA ::= {
   version [0] INTEGER DEFAULT 0,
   asIDs        SEQUENCE OF asIdsOrRange,
   ipAddrBlocks SEQUENCE OF BOAIPAddressFamily }

ASIdOrRange   ::= CHOICE {
   id               ASId,
   range            ASRange }

ASRange       ::= SEQUENCE {
   min              ASId,
   max              ASId }

ASId          ::= INTEGER

BOAIPAddressFamily ::= SEQUENCE {
   addressFamily OCTET STRING (SIZE (2..3)),
   addresses SEQUENCE OF IPAddress }

IPAddress ::= BIT STRING
```

2.1.3.2.1. version

The version number of the BogonOriginAttestation MUST be 0.

2.1.3.2.2. asIDs

The asIDs field contains the AS numbers that are to be regarded as Bogon ASes. The set of AS numbers may be explicitly listed, or specified as a continuous range of values. (See [RFC3779] for more details.)

2.1.3.2.3. BOAIPAddressFamily

The BOAIPAddressFamily field encodes the set of IP address prefixes that are to be regarded as Bogon IP addresses that are to be constrained from appearing in any routing advertisement. The intended semantics is that any route object that has the same address prefix as that listed as a Bogon IP address, or is a more specific prefix of a Bogon IP address can be regarded as a Bogon route object.

Note that the syntax here is more restrictive than that used in the IP Address Delegation extension defined in RFC 3779. That extension can represent arbitrary address ranges, whereas BOAs contain only prefixes.

Within the BOAIPAddressFamily structure, addressFamily contains the Address Family Identifier (AFI) of an IP address family. This specification only supports IPv4 and IPv6. Therefore, addressFamily MUST be either 0001 or 0002. The addresses field represents prefixes as a sequence of type IPAddress. (See [RFC3779] for more details.)

2.1.4. certificates

The certificates field MAY be included. If so, it MUST contain only the end entity (EE) certificate needed to validate this BOA. In the use context of BOAs being made available to relying parties via publication in a repository system, there is no a priori requirement to include the EE certificate in the BOA.

2.1.5. crls

The crls field MUST be omitted.

2.1.6. signerInfo

SignerInfo is defined under CMS as:

```
SignerInfo ::= SEQUENCE {
  version CMSVersion,
  sid SignerIdentifier,
  digestAlgorithm DigestAlgorithmIdentifier,
  signedAttrs [0] IMPLICIT SignedAttributes OPTIONAL,
  signatureAlgorithm SignatureAlgorithmIdentifier,
  signature SignatureValue,
  unsignedAttrs [1] IMPLICIT UnsignedAttributes OPTIONAL }
```

2.1.6.1. version

The version number MUST be 3, corresponding with the choice of SubjectKeyIdentifier for the sid.

2.1.6.2. sid

The sid is defined as:

```
SignerIdentifier ::= CHOICE {
  issuerAndSerialNumber IssuerAndSerialNumber,
  subjectKeyIdentifier [0] SubjectKeyIdentifier }
```

For a BOA, the sid MUST be a SubjectKeyIdentifier.

2.1.6.3. digestAlgorithm

The digestAlgorithm MUST be SHA-256, the OID for which is 2.16.840.1.101.3.4.2.1. [RFC4055]

2.1.6.4. signedAttrs

Signed Attributes are defined as:

```
SignedAttributes ::= SET SIZE (1..MAX) OF Attribute

UnsignedAttributes ::= SET SIZE (1..MAX) OF Attribute
Attribute ::= SEQUENCE {
  attrType OBJECT IDENTIFIER,
  attrValues SET OF AttributeValue }
AttributeValue ::= ANY
```

The signer MUST digitally sign a collection of attributes along with the content payload. Each attribute in the collection MUST be DER-encoded. The syntax for attributes is defined in [X.501], and the X.500 Directory provides a rich attribute syntax. A very simple subset of this syntax is used extensively in [RFC3852], where ATTRIBUTE.Type and ATTRIBUTE.id are the only parts of the ATTRIBUTE class that are employed.

Each of the attributes used with this CMS profile has a single attribute value. Even though the syntax is defined as a SET OF AttributeValue, there MUST be exactly one instance of AttributeValue present.

The SignedAttributes syntax within signerInfo is defined as a SET OF Attribute. The SignedAttributes MUST include only one instance of any particular attribute.

The signer MUST include the content-type and message-digest attributes. The signer MAY also include the signing-time signed attribute, the binary-signing-time signed attribute, or both signed attributes. Other signed attributes that are deemed appropriate MAY also be included. The intent is to allow additional signed attributes to be included if a future need is identified. This does not cause an interoperability concern because unrecognized signed attributes are ignored at verification.

2.1.6.4.1. Content-Type Attribute

```
id-contentType OBJECT IDENTIFIER ::= { iso(1) member-body(2)
    us(840) rsadsi(113549) pkcs(1) pkcs9(9) 3 }

ContentType ::= OBJECT IDENTIFIER
```

A content-type attribute is required to contain the same object identifier as the content type contained in the EncapsulatedContentInfo. The signer MUST include a content-type attribute containing the appropriate content type. Section 11.1 of the CMS Specification [RFC3852] defines the content-type attribute.

2.1.6.4.2. Message-Digest Attribute

```
id-messageDigest OBJECT IDENTIFIER ::= { iso(1) member-body(2)
    us(840) rsadsi(113549) pkcs(1) pkcs9(9) 4 }
MessageDigest ::= OCTET STRING
```

The signer MUST include a message-digest attribute, having as its value the output of a one-way hash function computed on the content that is being signed. Section 11.2 of the CMS Specification [RFC3852] defines the message-digest attribute.

2.1.6.4.3. Signing-Time Attribute

```
id-signingTime OBJECT IDENTIFIER ::= { iso(1) member-body(2)
    us(840) rsadsi(113549) pkcs(1) pkcs9(9) 5 }
SigningTime ::= Time
Time ::= CHOICE {
  utcTime UTCTime,
  generalizedTime GeneralizedTime }
```

The signing-time attribute MAY be present in a BOA.

The signing-time attribute specifies the time, based on the local system clock, at which the digital signature was applied to the content. If both signing-time and binary-signing-time are present,

the time that is represented in both attributes MUST represent the same time value. Section 11.3 of the CMS Specification [RFC3852] defines the content-type attribute.

2.1.6.4.4. Binary-Signing-Time Attribute

```
id-aa-binarySigningTime OBJECT IDENTIFIER ::= { iso(1)
    member-body(2) us(840) rsadsi(113549) pkcs(1) pkcs9(9)
    smime(16) aa(2) 46 }

BinarySigningTime ::= BinaryTime

BinaryTime ::= INTEGER (0..MAX)
```

The signer MAY include a binary-signing-time attribute, specifying the time at which the digital signature was applied to the content. If both signing-time and binary-signing-time are present, the time that is represented in both attributes MUST represent the same time value. The binary-signing-time attribute is defined in [RFC4049].

2.1.6.5. signatureAlgorithm

The signatureAlgorithm MUST be RSA (rsaEncryption), the OID for which is 1.2.840.113549.1.1.1.

2.1.6.6. signature

The signature value is defined as:

```
SignatureValue ::= OCTET STRING
```

The signature characteristics are defined by the digest and signature algorithms.

2.1.6.7. unsignedAttrs

unsignedAttrs MUST be omitted.

3. BOA Validation

Before a relying party can use a BOA as a constrictor of a routing announcement, the relying party must use the RPKI to validate the BOA. To do this the relying party performs the following steps:

1. Verify that the BOA syntax complies with this specification. In particular, verify the following:

A. The eContentType of the CMS object is id-ct-rpkiBOA (OID 1.2.840.113549.1.9.16.1.[TBS])

B. The version of the SignedData object is 3.

C. The digestAlgorithm in the SignedData object is SHA-256 (OID 2.16.840.1.101.3.4.2.1).

D. The crls field in the SignedData object is omitted.

E. The version of the BOA is 0.

F. The addressFamily in the BOAIPAddressFamily is either IPv4 or IPv6 (0001 and 0002, respectively).

G. The version of the SignerInfo is 3.

H. The digestAlgorithm in the SignerInfo object is SHA-256 (OID 2.16.840.1.101.3.4.2.1).

I. The signatureAlgorithm in the SignerInfo object is RSA (OID 1.2.840.113549.1.1.1).

J. The signedAttrs field in the SignerInfo object is included.

K. The unsignedAttrs field in the SignerInfo object is omitted.

2. Obtain an EE certificate that has a Subject Key Identifier (SKI) that matches the sid field of the SignerInfo object. This certificate may be obtained from the certificates field of the SignedData object (if present), the RPKI repository system, or a local cache.

3. Use the public key in the EE certificate to verify the signature on the BOA.

4. Verify that the EE certificate has an IP Address Delegation extension [RFC3779] and that the IP address prefix(es) in that extension exactly matches the IP address prefix(es) in the BOA, and the AS numbers in that extension exactly match the AS numbers in the BOA.

5. Verify that the EE certificate is a valid end-entity certificate in the resource PKI by constructing a valid certificate path to a trust anchor. (See [ID.ietf-sidr-res-certs] for more details.)

Note that requiring an exact match between the IP address prefixes and ASes in a BOA and the IP address prefixes and ASes in the

corresponding EE certificate does not place any limitations on BOA use. Since each EE certificate in the RPKI architecture is used to verify only a single BOA, it is natural to have the IP address prefixes in the certificate match those in the corresponding BOA.

4. BOA Use Practices

BOAs are intended to allow relying parties a means of validating whether route origination information as described in a route advertisement refers to an IP address or AS number that has not been validly allocated for use in the routing system.

Any party with a validly assigned Internet resource set and a CA certificate that described this delegation can publish a BOA, independently of the actions of the actions of the party that assigned the resource set. BOAs are not hierarchically related.

An Internet Registry SHOULD maintain a single BOA in relation to each parent registry that has assigned resources to this registry.

An Internet Registry SHOULD maintain a regular issuance cycle for BOAs.

For registries that operate on a day-to-day basis in terms of resource transactions, it is suggested that a local BOA management practice would be that a new BOA should be issued on a regular 24 hour basis. The corresponding EE certificate should have a validity period of no more than 72 hours from the time of issuance. Each time a new EE certificate for a BOA is issued the previous BOA's EE certificate should be revoked and the previous BOA removed from the publication repository.

Parties that operate a local cache of RPKI objects should ensure that they refresh BOA objects at intervals 24 hours to ensure that they have the current BOA in the local cache.

5. BOA Interpretation

A BOA can be used to check a route object to determine if the origination information in the route object refers to invalid IP addresses or an invalid AS number.

If a route object has an AS origination that refers to an AS number that is included in a valid BOA then the route object can be regarded as a Bogon object, and local policies that apply to Bogon ASes can be applied to the object. This holds whether or not the address prefix

of the route object is described by a valid ROA or not.

If a route object has an address prefix that is equal to, or is a more specific prefix of an IP address that is included in a valid BOA then the route object can be regarded as a Bogon object, and local policies that apply to Bogon ASes can be applied to the object, unless the address prefix and AS origination of the route object is also described by a valid ROA, in which case the BOA is to be disregarded.

6. Security Considerations

The purpose of a BOA is to convey an attestation by an address holder that there is no authority for the generation of a route object that refers to specified addresses or origination from specified ASes. The integrity of a BOA must be established in order to validate the authority of the Bogon Attestation. The BOA makes use of the CMS signed message format for integrity, and thus inherits the security considerations associated with that data structure. The right of the BOA signer to authorize the attestation of specified IP addresses and ASes as Bogons is established through use of the address space and AS number PKI described in [ID.ietf-sidr-arch]. Specifically, a relying party must verify the signature on the BOA using an X.509 certificate issued under this PKI, and check that the prefix(es) in the BOA match those in the address space extension in the certificate.

7. IANA Considerations

[None]

8. Acknowledgments

The authors are indebted to the authors of Route Origin Authorization (ROA) [ID.ietf-sidr-roa-format], M. Lepinski, S. Kent and D. Kong, as much of the text used to define a BOA has been borrowed from the ROA format specification, and Russ Housley for clarification on the CMS profile.

9. Normative References

[ID.ietf-sidr-arch]
Lepinski, M., Kent, S., and R. Barnes, "An Infrastructure to Support Secure Internet Routing", draft-ietf-sidr-arch (work in progress), November 2007.

[ID.ietf-sidr-res-certs]
Huston, G., Michaleson, G., and R. Loomans, "A Profile for X.509 PKIX Resource Certificates", Work in progress: Internet Drafts draft-ietf-sidr-res-certs-09.txt, November 2007.

[ID.ietf-sidr-roa-format]
Lepinski, M., Kent, S., and D. Kong, "An Infrastructure to Support Secure Internet Routing", draft-ietf-sidr-roa-format (work in progress), July 2007.

[RFC3280] Housley, R., Polk, W., Ford, W., and D. Solo, "Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile", RFC 3280, April 2002.

[RFC3779] Lynn, C., Kent, S., and K. Seo, "X.509 Extensions for IP Addresses and AS Identifiers", RFC 3779, June 2004.

[RFC3852] Housley, R., "Cryptographic Message Syntax (CMS)", RFC 3852, July 2004.

[RFC4049] Housley, R., "BinaryTime: An Alternate Format for Representing Date and Time in ASN.1", RFC 4049, April 2005.

[RFC4055] Schaad, J., Kaliski, B., and R. Housley, "Additional Algorithms and Identifiers for RSA Cryptography for use in the Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile", RFC 4055, June 2005.

[RFC4271] Rekhter, Y., Li, T., and S. Hares, "A Border Gateway Protocol 4 (BGP-4)", RFC 4271, January 2006.

[X.501] ITU-T, "ITU-T Recommendation X.501: Information Technology - Open Systems interconnection - The Director Models", 1993.

Authors' Addresses

Geoff Huston
Asia Pacific Network Information Centre

Email: gih@apnic.net
URI: http://www.apnic.net

Terry Manderson
Asia Pacific Network Information Centre

Email: terry@apnic.net
URI: http://www.apnic.net

George Michaelson
Asia Pacific Network Information Centre

Email: ggm@apnic.net
URI: http://www.apnic.net

Full Copyright Statement

Intellectual Property

Acknowledgment

Funding for the RFC Editor function is provided by the IETF Administrative Support Activity (IASA).