Network Working Group
Internet-Draft
Intended status: Standards Track
Expires: 21 October 2026

J. Snijders
BSD
A. Azimov
Yandex
E. Uskov
JetLend
R. Bush
Internet Initiative Japan
R. Housley
Vigil Security
B. Maddison
Workonline

19 April 2026

A Profile for Autonomous System Provider Authorization
draft-ietf-sidrops-aspa-profile-26

Abstract

This document defines a Cryptographic Message Syntax (CMS) protected content type for Autonomous System Provider Authorization (ASPA) objects for use with the Resource Public Key Infrastructure (RPKI). An ASPA is a digitally signed object through which the issuer (the holder of an Autonomous System identifier), can authorize one or more other Autonomous Systems (ASes) as its transit providers. When validated, an ASPA's eContent can be used for detection and mitigation of route leaks.

Requirements Language

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "NOT RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in BCP 14 [RFC2119] [RFC8174] when, and only when, they appear in all capitals, as shown here.

Status of This Memo

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at https://datatracker.ietf.org/drafts/current/.

Copyright Notice

Table of Contents

1. Introduction

The primary purpose of the Resource Public Key Infrastructure (RPKI) [RFC6480] is to improve security in the global Internet routing system. As part of this infrastructure, a mechanism is needed for Autonomous Systems (AS) operators, in their capacity as customers, to designate and authorize other ASes as their Provider(s). A Provider AS (PAS) is a network providing connectivity between networks - it provides transit services to the customer. That is:

a. The provider may propagate Border Gateway Protocol (BGP) routes received from any direction. For example, routes the provider learned from its own providers, lateral peers, and other customers. The provider may also announce a default route to their customers.

b. The provider may propagate BGP routes received from the customer in any direction. For example, to the provider’s providers, lateral peers, and other customers.

This document specifies a digitally signed Autonomous System Provider Authorization (ASPA) object profile. An ASPA object is a cryptographically verifiable attestation signed by the holder of an Autonomous System identifier, hereafter called the "Customer AS", or CAS. An ASPA object contains a list of one or more Provider ASes each of which is authorized to be a Provider network for the CAS.

This profile provides the authorization mechanism mentioned above and can be used to facilitate detection and mitigation of route leaks. The procedures for verifying AS_PATHs in BGP UPDATE messages using ASPAs are described in [I-D.ietf-sidrops-aspa-verification].

When the Customer AS makes use of multiple providers, all Provider ASes are to be listed in the ASPA object, including any non-transparent Internet Exchange Point (IXP) Route Server (RS) ASes. Note that the common case for RS ASes at IXPs is to operate transparently (see Section 2.2.2.1 [RFC7947]), and transparent IXP Route Servers do not need to be listed as PAS in ASPAs.

This CMS [RFC5652] protected content type definition conforms to the [RFC6488] template for RPKI signed objects. In accordance with Section 4 of [RFC6488], this document defines:

1. The object identifier (OID) that identifies the ASPA signed object. This OID appears in the eContentType field of the encapContentInfo object as well as the content-type signed attribute within the signerInfo structure.

2. The ASN.1 syntax for the ASPA content, which is the payload signed by the CAS. The ASPA content is encoded using the ASN.1 [X.680] Distinguished Encoding Rules (DER) [X.690].

3. The steps required to validate an ASPA beyond the validation steps specified in [RFC6488].

This document also provides implementation guidance in Section 5.

2. ASPA Content Type

The content-type for an ASPA is defined as id-ct-ASPA, which has the numerical value of 1.2.840.113549.1.9.16.1.49. This OID MUST appear both within the eContentType in the encapContentInfo structure as well as the content-type signed attribute within the signerInfo structure (see [RFC6488]).

3. ASPA eContent

The content of an ASPA identifies the Customer AS (CAS) as well as the Set of Provider ASes (SPAS) that are authorized by the CAS to be its Providers.

The eContent of an ASPA is an instance of ASProviderAttestation, formally defined by the following ASN.1 [X.680] module:

```
RPKI-ASPA-2023
  { iso(1) member-body(2) us(840) rsadsi(113549) pkcs(1)
     pkcs-9(9) smime(16) modules(0) id-mod-rpki-aspa-2023(TBD) }

DEFINITIONS EXPLICIT TAGS ::=
BEGIN

IMPORTS
  CONTENT-TYPE
  FROM CryptographicMessageSyntax-2010 -- From RFC 6268
    { iso(1) member-body(2) us(840) rsadsi(113549) pkcs(1)
       pkcs-9(9) smime(16) modules(0) id-mod-cms-2009(58) } ;

id-ct-ASPA OBJECT IDENTIFIER ::=
  { iso(1) member-body(2) us(840) rsadsi(113549) pkcs(1)
    pkcs-9(9) id-smime(16) id-ct(1) aspa(49) }

ct-ASPA CONTENT-TYPE ::=
  { TYPE ASProviderAttestation IDENTIFIED BY id-ct-ASPA }

ASProviderAttestation ::= SEQUENCE {
  version       [0] INTEGER DEFAULT 0,
  customerASID      CAS,
  providers         ProviderASSet }

CAS ::= INTEGER (1..4294967295)

ProviderASSet ::= SEQUENCE (SIZE(1..MAX)) OF PAS

PAS ::= INTEGER (0..4294967295)

END
```

This content appears as the eContent within the encapContentInfo as specified in [RFC6488].

3.1. version

The version number of the ASProviderAttestation that complies with this specification MUST be 1 and MUST be explicitly encoded.

3.2. customerASID

The customerASID field contains a positive integer that represents the AS number of the Customer Autonomous System that is the authorizing entity.

3.3. providers

The providers field contains the listing of ASes that are authorized as providers.

Each element contained in the providers field is an instance of PAS. Each PAS element contains the AS number of an AS that has been authorized by the customer AS as its provider or non-transparent RS.

In addition to the constraints described by the formal ASN.1 definition, the contents of the providers field MUST satisfy the following constraints:

* The CustomerASID value MUST NOT appear in any PAS in the providers field.

* The elements of providers MUST be ordered in ascending numerical order.

* Each value of PAS MUST be unique (with respect to the other elements of providers).

* An PAS value of 0 can only be encoded in the providers field as a single item list, i.e., an element for AS 0 MUST NOT appear alongside any other elements.

4. ASPA Validation

Before a relying party can use an ASPA to validate a routing announcement, the relying party MUST first validate the ASPA object itself. To validate an ASPA, the relying party MUST perform all the validation checks specified in [RFC6488] as well as the following additional ASPA-specific validation steps:

* The Autonomous System Identifier Delegation Extension [RFC3779] MUST be present in the end-entity (EE) certificate (contained within the ASPA), and the Customer ASID in the ASPA eContent MUST match the ASId specified by the EE certificate’s Autonomous System Identifier Delegation Extension.

* The Autonomous System Identifier Delegation Extension MUST contain exactly one "id" element (Section 3.2.3.6 of [RFC3779]) and MUST NOT contain any "inherit" elements (Section 3.2.3.3 of [RFC3779]) or "range" elements (Section 3.2.3.7 of [RFC3779]).

* The IP Address Delegation Extension [RFC3779] MUST be absent.

5. Implementation Considerations

5.1. One ASPA Object per Customer AS

For any given Customer AS, only a single ASPA object SHOULD be maintained which contains all providers (including any non-transparent RS ASes). The practice of maintaining a single object per Customer AS avoids race conditions during ASPA updates that might impact BGP route propagation. When an ASPA record is being migrated between different CA registries, the authorization contents of its instances at those registries SHOULD be identical (remain unchanged) for the duration of the migration. The CA software that maintains ASPA records SHOULD support enforcement of this recommendation. See Section 4 of [I-D.ietf-sidrops-aspa-verification] for all ASPA registration recommendations.

5.2. Use of One-Time Use End Entity Certificates

CAs are RECOMMENDED to generate a new key pair for each new ASPA and only sign one ASPA with each EE certificate. This type of EE certificate is termed a "one-time-use" EE certificate (see Section 3 of [RFC6487]).

5.3. ASPA Object Filenames

CAs are RECOMMENDED to follow the guidelines for naming ASPA objects based on Section 2.2 of [RFC6481], i.e., convert the 160-bit hash of the EE’s public key value into a 27-character string using Base 64 Encoding with the URL and Filename Safe Alphabet (see Section 5 of [RFC4648]). See Section 7.7 of [I-D.ietf-sidrops-publication-server-bcp] for more information and considerations.

5.4. Upper Bound on the Number of Providers

While the ASN.1 profile specified in Section 3 imposes no limit on the number of Provider ASes that can be listed for a given CAS, consideration will need to be given to limitations existing in validators and elsewhere in the RPKI ecosystem. For example, the number of Provider ASes that can be listed in a single RPKI-To-Router protocol ASPA PDU following the Length field constraints in Section 5.1 of [I-D.ietf-sidrops-8210bis] is 16,380 providers. In addition to protocol limitations in the ecosystem, locally defined restrictions could exist for the maximum file size of signed objects a Relying Party implementation is willing to accept.

Relying Party implementations are RECOMMENDED to impose an upper bound on the number of Provider ASes for a given CAS. An upper bound value between 4,000 and 10,000 Provider ASes is suggested. If this threshold is exceeded, Relying Party implementations SHOULD treat all ASPA objects related to the CAS invalid; e.g. not emit a partial list of Provider ASes. Additionally, an error SHOULD be logged in the local system, indicating the CAS for which the threshold was exceeded. Implementers and operators SHOULD periodically review whether imposed upper bounds still are reasonable in context of the global Internet routing system.

6. Security Considerations

The security considerations of [RFC6481], [RFC6485], and [RFC6488] also apply to ASPAs.

7. IANA Considerations

7.1. SMI Security for S/MIME Module Identifier registry

IANA is requested to allocate for id-mod-rpki-aspa-2023 in the "SMI Security for S/MIME Module Identifier (1.2.840.113549.1.9.16.0)" registry as follows:

Decimal	Description	Specification
TBD2	id-mod-rpki-aspa-2023	[RFC-to-be]

Table 1

7.2. SMI Security for S/MIME CMS Content Type registry

IANA is requested to make permanent in the "SMI Security for S/MIME CMS Content Type (1.2.840.113549.1.9.16.1)" registry as follows:

Decimal	Description	Specification
49	id-ct-ASPA	[RFC-to-be]

Table 2

7.3. RPKI Signed Object registry

IANA is requested to make permanent in the "RPKI Signed Object" registry as follows:

Name	OID	Specification
Autonomous System Provider Authorization	1.2.840.113549.1.9.16.1.49	[RFC-to-be]

Table 3

7.4. RPKI Repository Name Scheme registry

IANA is requested to make permanent in the "RPKI Repository Name Scheme" registry [RFC6481] as follows:

Filename Extension	RPKI Object	Reference
.asa	Autonomous System Provider Authorization	[RFC-to-be]

Table 4

7.5. Media Type registry

The IANA is requested to register the media type application/rpki-aspa in the "Media Type" registry as follows:

```
Type name: application
Subtype name: rpki-aspa
Required parameters: N/A
Optional parameters: N/A
Encoding considerations: binary
Security considerations: Carries an RPKI ASPA [RFC-to-be].
    This media type contains no active content. See
    Section 4 of [RFC-to-be] for further information.
Interoperability considerations: None
Published specification: [RFC-to-be]
Applications that use this media type: RPKI operators
Additional information:
  Content: This media type is a signed object, as defined
      in [RFC6488], which contains a payload of a list of
      AS identifers as defined in [RFC-to-be].
  Magic number(s): None
  File extension(s): .asa
  Macintosh file type code(s):
Person & email address to contact for further information:
  Job Snijders <job@bsd.nl>
Intended usage: COMMON
Restrictions on usage: None
Change controller: IETF
```

8. Implementation status

This section is to be removed before publishing as an RFC.

This section records the status of known implementations of the protocol defined by this specification at the time of posting of this Internet-Draft, and is based on a proposal described in RFC 7942. The description of implementations in this section is intended to assist the IETF in its decision processes in progressing drafts to RFCs. Please note that the listing of any individual implementation here does not imply endorsement by the IETF. Furthermore, no effort has been spent to verify the information presented here that was supplied by IETF contributors. This is not intended as, and must not be construed to be, a catalog of available implementations or their features. Readers are advised to note that other implementations may exist.

According to RFC 7942, "this will allow reviewers and working groups to assign due consideration to documents that have the benefit of running code, which may serve as evidence of valuable experimentation and feedback that have made the implemented protocols more mature. It is up to the individual working groups to use this information as they see fit".

* A validator implementation [rpki-client] written in C was provided by Job Snijders.

* A validator implementation [routinator] written in Rust was provided by Martin Hoffman from NLnet Labs.

* A validator implementation [rpki-prover] written in Haskell was provided by Mikhail Puzanov.

* A signer implementation [rpki-aspa-demo] written in Perl was provided by Tom Harrison from APNIC.

* A signer implementation [rpki-commons] in Java was reported on by Ties de Kock from RIPE NCC.

* A signer implementation [krill] in Rust was reported on by Tim Bruijnzeels.

9. Acknowledgments

The authors would like to thank Keyur Patel for helping kick-start the ASPA profile project, Ties de Kock & Tim Bruijnzeels for suggesting that the ProviderASSet be in a canonical form, and Claudio Jeker, Martin Hoffman, Lancheng Qin, and Jeff Haas for review and several suggestions for improvements.

Contributors

The following people made significant contributions to this document:

Kotikalapudi Sriram
USA National Institute of Standards and Technology
Email: ksriram@nist.gov

References

Normative References

[I-D.ietf-sidrops-8210bis]
Bush, R., Austein, R., and T. Harrison, "The Resource Public Key Infrastructure (RPKI) to Router Protocol, Version 2", Work in Progress, Internet-Draft, draft-ietf-sidrops-8210bis-25, 2 March 2026, <https://datatracker.ietf.org/doc/html/draft-ietf-sidrops-8210bis-25>.

[I-D.ietf-sidrops-aspa-verification]
Azimov, A., Bogomazov, E., Bush, R., Patel, K., Snijders, J., and K. Sriram, "BGP AS_PATH Verification Based on Autonomous System Provider Authorization (ASPA) Objects", Work in Progress, Internet-Draft, draft-ietf-sidrops-aspa-verification-24, 19 October 2025, <https://datatracker.ietf.org/doc/html/draft-ietf-sidrops-aspa-verification-24>.

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997, <https://www.rfc-editor.org/info/rfc2119>.

[RFC3779] Lynn, C., Kent, S., and K. Seo, "X.509 Extensions for IP Addresses and AS Identifiers", RFC 3779, DOI 10.17487/RFC3779, June 2004, <https://www.rfc-editor.org/info/rfc3779>.

[RFC5652] Housley, R., "Cryptographic Message Syntax (CMS)", STD 70, RFC 5652, DOI 10.17487/RFC5652, September 2009, <https://www.rfc-editor.org/info/rfc5652>.

[RFC6481] Huston, G., Loomans, R., and G. Michaelson, "A Profile for Resource Certificate Repository Structure", RFC 6481, DOI 10.17487/RFC6481, February 2012, <https://www.rfc-editor.org/info/rfc6481>.

[RFC6485] Huston, G., "The Profile for Algorithms and Key Sizes for Use in the Resource Public Key Infrastructure (RPKI)", RFC 6485, DOI 10.17487/RFC6485, February 2012, <https://www.rfc-editor.org/info/rfc6485>.

[RFC6488] Lepinski, M., Chi, A., and S. Kent, "Signed Object Template for the Resource Public Key Infrastructure (RPKI)", RFC 6488, DOI 10.17487/RFC6488, February 2012, <https://www.rfc-editor.org/info/rfc6488>.

[RFC8174] Leiba, B., "Ambiguity of Uppercase vs Lowercase in RFC 2119 Key Words", BCP 14, RFC 8174, DOI 10.17487/RFC8174, May 2017, <https://www.rfc-editor.org/info/rfc8174>.

[X.680] ITU-T, "Information technology - Abstract Syntax Notation One (ASN.1): Specification of basic notation", ITU-T Recommendation X.680, 2021.

[X.690] ITU-T, "Information Technology - ASN.1 encoding rules: Specification of Basic Encoding Rules (BER), Canonical Encoding Rules (CER) and Distinguished Encoding Rules (DER)", ITU-T Recommendation X.690, 2021.

Informative References

[I-D.ietf-sidrops-publication-server-bcp]
Bruijnzeels, T., de Kock, T., Hill, F., Harrison, T., and J. Snijders, "Best Practises for Operating Resource Public Key Infrastructure (RPKI) Publication Services", Work in Progress, Internet-Draft, draft-ietf-sidrops-publication-server-bcp-07, 21 March 2026, <https://datatracker.ietf.org/doc/html/draft-ietf-sidrops-publication-server-bcp-07>.

[krill] Bruijnzeels, T., "krill", 2023, <https://mailarchive.ietf.org/arch/msg/sidrops/RrHCYTmevxDHgebdLC_adRlKH-o/>.

[RFC4648] Josefsson, S., "The Base16, Base32, and Base64 Data Encodings", RFC 4648, DOI 10.17487/RFC4648, October 2006, <https://www.rfc-editor.org/info/rfc4648>.

[RFC6480] Lepinski, M. and S. Kent, "An Infrastructure to Support Secure Internet Routing", RFC 6480, DOI 10.17487/RFC6480, February 2012, <https://www.rfc-editor.org/info/rfc6480>.

[RFC6487] Huston, G., Michaelson, G., and R. Loomans, "A Profile for X.509 PKIX Resource Certificates", RFC 6487, DOI 10.17487/RFC6487, February 2012, <https://www.rfc-editor.org/info/rfc6487>.

[RFC7947] Jasinska, E., Hilliard, N., Raszuk, R., and N. Bakker, "Internet Exchange BGP Route Server", RFC 7947, DOI 10.17487/RFC7947, September 2016, <https://www.rfc-editor.org/info/rfc7947>.

[routinator]
Hoffman, M., "routinator", 2023, <https://github.com/NLnetLabs/rpki-rs/pull/264>.

[rpki-aspa-demo]
Harrison, T., "rpki-aspa-demo", 2023, <https://github.com/APNIC-net/rpki-aspa-demo>.

[rpki-client]
Jeker, C., Snijders, J., Dzonsons, K., and T. Buehler, "OpenBSD rpki-client", 2023, <https://www.rpki-client.org/>.

[rpki-commons]
de Kock, T., "rpki-commons", 2023, <https://mailarchive.ietf.org/arch/msg/sidrops/nNAmZMrr7t9NMzm12jRXU03ABN4/>.

[rpki-prover]
Puzanov, M., "rpki-prover", 2023, <https://github.com/lolepezy/rpki-prover/compare/master...aspa-profile-16>.

Appendix A. Example ASPA eContent Payload

Below an example of a DER encoded ASPA eContent is provided with annotation following the '#' character.

```
 $ echo 301DA003020101020300FE633011020300FC00020301000F020500FA56EA00 \
   | xxd -r -ps | openssl asn1parse -inform DER -dump -i
   0:d=0  hl=2 l=  29 cons: SEQUENCE
   2:d=1  hl=2 l=   3 cons:  cont [ 0 ]
   4:d=2  hl=2 l=   1 prim:   INTEGER :01       # version
   7:d=1  hl=2 l=   3 prim:  INTEGER  :FE63     # CAS 65123
  12:d=1  hl=2 l=  17 cons:  SEQUENCE           # ProviderASSet
  14:d=2  hl=2 l=   3 prim:   INTEGER :FC00       # PAS 64512
  19:d=2  hl=2 l=   3 prim:   INTEGER :01000F     # PAS 65551
  24:d=2  hl=2 l=   5 prim:   INTEGER :FA56EA00   # PAS 4200000000
```

Below is a complete Base64 [RFC4648] encoded RPKI ASPA Signed Object.

```
MIIGLAYJKoZIhvcNAQcCoIIGHTCCBhkCAQMxDTALBglghkgBZQMEAgEwMAYLKoZI
hvcNAQkQATGgIQQfMB2gAwIBAQIDAP5jMBECAwD8AAIDAQAPAgUA+lbqAKCCBCMw
ggQfMIIDB6ADAgECAgEEMA0GCSqGSIb3DQEBCwUAMA8xDTALBgNVBAMTBHJvb3Qw
HhcNMjUwMTA2MTAyNjQ4WhcNMjYwMTA2MTAyNjQ4WjAPMQ0wCwYDVQQDEwRyb290
MIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIBCgKCAQEAmgjRVMrjafIK81cxs8GB
ehcjlP97o3CdlhceX8ogtUE19C1v3VZQttdzaBG1ViI2+TJiyueHm2M6nq7tl7A9
HfkPWGdvwjTUeF4ynpxNlqqoG5pGPMLuuk6HdVaKbuEeX3ZAYD2daG8qm3zkmmF3
F7e+xPvr6nCTIQX7nr0WQyX74FmLGwlr6TV+6MGdCnp24A5aCXJo3m1vGlQUKdd/
bMszpbfzaX1rxBKA0GSdWx/QEOxFb1BPObjF+P/GpDXOq3i0DoZiDB+omgdUAWDZ
dmskG0l+2EyQaCS/pVCES50Y5zAagTcFabCLKnPvr3qh9tvjD/KlolYc35gNIfoX
7wIDAQABo4IBhDCCAYAwDgYDVR0PAQH/BAQDAgeAMB0GA1UdDgQWBBQrh8dvXu72
IET1KLgskpso1VcyrDAfBgNVHSMEGDAWgBQ2mtAZLGdOeDIizTKFZreUErGPJjAY
BgNVHSABAf8EDjAMMAoGCCsGAQUFBw4CMF8GCCsGAQUFBwEBBFMwUTBPBggrBgEF
BQcwAoZDcnN5bmM6Ly9sb2NhbGhvc3QvcmVwby8zNjlBRDAxOTJDNjc0RTc4MzIy
MkNEMzI4NTY2Qjc5NDEyQjE4RjI2LmNlcjBXBgNVHR8EUDBOMEygSqBIhkZyc3lu
YzovL2xvY2FsaG9zdC9yZXBvL3RhLzM2OUFEMDE5MkM2NzRFNzgzMjIyQ0QzMjg1
NjZCNzk0MTJCMThGMjYuY3JsMD4GCCsGAQUFBwELBDIwMDAuBggrBgEFBQcwC4Yi
cnN5bmM6Ly9sb2NhbGhvc3QvdGEvYW4tb2JqZWN0LmFzYTAaBggrBgEFBQcBCAEB
/wQLMAmgBzAFAgMA/mMwDQYJKoZIhvcNAQELBQADggEBACqgWrD692DUuN1jrriv
SGqI7JqAtbqOSvnwxNGMlPZ6oTkarf4aDMJNcwqMaOUOzZPQ9VXA48h5R8TgvAub
s/HJ263DLtPiScwkWqLZZs6Ius2zFothzDwWOoe/rHKiecCF0YpJqFhaa6dw6vsO
zK77Ze+Gfa1SeeuI1DTwGDjiNBtXCaRxAPmvpYGxSKQmRYCIo6vKPSZlzCPqzEZa
MVCmxdlaDQjVPUuWLbFV/bbZmS4wM1nbikt5WLZEVHMcAyqWOb7a3KO2GIIw0Ak9
O6JgUoex/8y0s6smSWcRe2y9d6kAhT0COW1KvFoNM5lFKSe1rYqTtMFY33XfCI7e
IgsxggGqMIIBpgIBA4AUK4fHb17u9iBE9Si4LJKbKNVXMqwwCwYJYIZIAWUDBAIB
oGswGgYJKoZIhvcNAQkDMQ0GCyqGSIb3DQEJEAExMBwGCSqGSIb3DQEJBTEPFw0y
NTAxMDYxMDI2NDhaMC8GCSqGSIb3DQEJBDEiBCCLMweRYDN5u5auRQYv6+Dx+b9X
m1t5R3gkH9c3aX40gzANBgkqhkiG9w0BAQEFAASCAQATWDI3fYgku2fJPzFXAbnz
IKabFMRvhp9LAhvl8oPkCp0zQu4SyJsdvoWxkpHKnXGwwdgub/d4GF0weoJgauDr
ugUsB2e40aQwFoTyPcVuS/BirhlW5j0NwYovmuJ9GiBe67/sCRBaPC5sBKRBpOIv
IpMpAe2QhaoeDUDc0KYCM42f1kFD+PD8PnlvXRisL3A3OFhB+0LY1Z3xivRAEdtf
qzobZT0Hq6CIlAjNeocG0gMfJj/lFyElLkJdfGHzCWLnVNX2I+9GbDDdYV13cUTG
yk5CFeQWetsX7D7XRNSdKwrrW6qWK/KxGF38SsuGeTyDQiNcOhwqvSfFc388ge/G
```

The above should decode as following:

```
Object SHA256 hash:        S6B+jKOCFXPlRn7ws6Kd5tgpsSx609tJZpw60CVaf9Y=
EE Subject keyid:          2B87C76F5EEEF62044F528B82C929B28D55732AC
EE Certificate issuer:     /CN=root
EE Certificate serial:     04
EE Authority keyid:        369AD0192C674E783222CD328566B79412B18F26
EE Authority info access:
     rsync://localhost/repo/369AD0192C674E783222CD328566B79412B18F26.cer
EE Subject info access:    rsync://localhost/ta/an-object.asa
CMS Signing time:          Mon 06 Jan 2025 10:26:48 +0000
EE notBefore:              Mon 06 Jan 2025 10:26:48 +0000
EE notAfter:               Tue 06 Jan 2026 10:26:48 +0000

ASPA eContent:
  customerASID:            65123
  providers:               64512, 65551, 4200000000
```

Authors' Addresses

Job Snijders
BSD Software Development
Amsterdam
Netherlands
Email: job@bsd.nl
URI: https://www.bsd.nl

Alexander Azimov
Yandex
Email: a.e.azimov@gmail.com

Eugene Uskov
JetLend
Email: eu@jetlend.ru

Randy Bush
Internet Initiative Japan
Email: randy@psg.com

Russ Housley
Vigil Security, LLC
918 Spring Knoll Drive
Herndon, VA 20170
United States of America
Email: housley@vigilsec.com

Ben Maddison
Workonline
Cape Town
South Africa
Email: benm@workonline.africa