SIDR Operations
Internet-Draft
Intended status: Standards Track
Expires: 20 September 2025

L. Chen
L. Liu
Zhongguancun Laboratory
D. Li
Tsinghua University
L. Qin
Zhongguancun Laboratory

19 March 2025

A Profile of Signed SAVNET-Peering Information (SiSPI) Object for Deploying Inter-domain SAVNET
draft-chen-sidrops-sispi-03

Abstract

This document defines a "Signed SAVNET-Peering Information" (SiSPI) object, a Cryptographic Message Syntax (CMS) protected content type included in the Resource Public Key Infrastructure (RPKI). A SiSPI object is a digitally signed object which carries the list of Autonomous Systems (ASes) deploying inter-domain SAVNET. When validated, the eContent of a SiSPI object confirms that the holder of the listed ASN produces the object and the AS has deployed inter-domain SAV and is ready to establish neighbor relationship for preventing source address spoofing.

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/.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on 20 September 2025.

Table of Contents

1. Introduction 2
1.1. Terminology 4
1.2. Requirements Language 4
2. The SiSPI ContentType 4
3. The SiSPI eContent 4
3.1. version 6
3.2. asID 6
3.3. addresses 6
3.3.1. Element IPFamilyAddresses 6
4. SiSPI Validation 7
5. IANA Considerations 8
5.1. RPKI Signed Object Registry 8
5.2. RPKI Repository Name Scheme Registry 8
5.3. SMI Security for S/MIME Module Identifier (1.2.840.113549.1.9.16.0) 9
5.4. Media Type Registry 9
6. Using SiSPI 9
7. Newly SAVNET-adopting ASes 12
8. Security Considerations 12
9. References 12
9.1. Normative References 12
9.2. Informative References 13
Authors' Addresses 14

1. Introduction

Attacks based on source IP address spoofing, such as reflective DDoS and flooding attacks, continue to present significant challenges to Internet security. Mitigating these attacks in inter-domain networks requires effective source address validation (SAV). While BCP84 [RFC3704] [RFC8704] offers some SAV solutions, such as ACL-based ingress filtering and uRPF-based mechanisms, existing inter-domain SAV mechanisms have limitations in terms of validation accuracy and operational overhead in different scenarios [inter-domain-ps].

Inter-domain SAVNET [savnet] proposes to exchange SAV-specific information among ASes to solve the problems of existing inter-domain SAV mechanisms. Two SAV-specific information exchanging protocols (or SAVNET protocols for short) are shown to achieve higher validation accuracy and lower operational overhead in large-scale emulations [emu-9-savs]. However, operators face significant difficulties in deploying SAVNET protocols. To benefit Internet routing, supporting incremental deployment is an essential requirement of SAVNET protocols [inter-domain-ps]. As illustrated in the Section 9.2 of [savnet], during the partial or incremental deployment of SAVNET protocols, protocol-speaking agents (or SAVNET agents) within the SAVNET-adopting ASes need to find and establish connections with other SAVNET agents. Currently, there is no mechanism to achieve this automatically, and operators of SAVNET-adopting ASes must configure peering SAVNET relationship by hand, which is slow and error-prone.

The neighbor discovery and connection setup process of SAV protocols can be done in an automatic and correct manner, with the introduction of a public registry that contains all ASes which both deploy SAVNET and are willing to setup SAVNET peering relationships. A newly adopting AS can use this registry as a reference, and pick appropriate ASes to setup SAVNET peering relationship.

The Resource Public Key Infrastructure (RPKI) is the most suitable to host this public registry, because the primary purpose of RPKI is to improve routing security [RFC6480], and defending against address spoofing is a main aspect of routing security. To this end, a mechanism is needed to facilitate holders of Automous System (AS) identifiers to declare their deployment of SAVNET [savnet]. The digitally Signed SAVNET-Peering Information (SiSPI) object described in this document serves the function.

A SiSPI object is a cryptographically verifiable attestation signed by the holder of an AS identifier. It contains the identification information of one AS, which means the listed AS has deployed SAVNET and can perform SAV on its data plane.

The SiSPI object makes use of the template for RPKI digitally signed objects [RFC6488], which defines a Crytopgraphic Message Syntax (CMS) [RFC5652] wrapper for the SiSPI content as well as a generic validation procedure for RPKI signed objects. In accordance with Section 4 of [RFC6488], this document defines:

1. The object identifier (OID) that identifies the SiSPI 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 SiSPI eContent, which is the payload that specifies the AS deploying SAVNET. The SiSPI eContent is encoded using the ASN.1 Distinguished Encoding Rules (DER) [X.690].

3. The steps required to validate a SiSPI beyond the validation steps specified in [RFC6488].

1.1. Terminology

This document makes use of the terms and concepts described in "Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile" [RFC5280], "X.509 Extensions for IP Address and AS Identifiers" [RFC3779], "Signed Object Template for the Resource Public Key Infrastructure (RPKI)" [RFC6488], and "A Profile for X.509 PKIX Resource Certificates" [RFC6487]. The readers should be familiar with the terms and concepts.

1.2. 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.

2. The SiSPI ContentType

The content-type for a SiSPI object is defined as id-ct-rpkiSiSPI, which has the numerical value of 1.2.840.113549.1.9.16.1.TBD. This OID MUST appear both within the eContentType in the encapContentInfo structure as well as the ContentType signed attribute within the signerInfo structure (see [RFC6488]).

3. The SiSPI eContent

The content of a SiSPI object identifies a single AS that has deployed SAVNET [savnet] for inter-domain SAV and a list of its IP addresses. The eContent of a SiSPI object is an instance of SAVNETAttestation, formally defined by the following ASN.1 [X.680] module:

```
RpkiSiSPI-2024
     { iso(1) member-body(2) us(840) rsadsi(113549)
       pkcs(1) pkcs9(9) smime(16) mod(0)
       id-mod-rpkiSiSPI-2024-2024(TBD0) }

DEFINITIONS EXPLICIT TAGS ::=
```

```
BEGIN

IMPORTS
  CONTENT-TYPE
  FROM CryptographicMessageSyntax-2010 -- in [RFC6268]
    { iso(1) member-body(2) us(840) rsadsi(113549) pkcs(1)
      pkcs-9(9) smime(16) modules(0) id-mod-cms-2009(58) } ;

ct-rpkiSiSPI CONTENT-TYPE ::=
  { TYPE SAVNETAttestation IDENTIFIED BY id-ct-rpkiSiSPI }

id-ct-rpkiSiSPI OBJECT IDENTIFIER ::=
  { iso(1) member-body(2) us(840) rsadsi(113549) pkcs(1)
    pkcs-9(9) id-smime(16) id-ct(1) TBD1 }

SAVNETAttestation ::= SEQUENCE {
  version [0]   INTEGER DEFAULT 0,
  asID          ASID,
  addresses     SEQUENCE OF IPFamilyAddresses }

ASID ::= INTEGER (0..4294967295)

IPFamilyAddresses ::= SEQUENCE {
  ipFamily    IP-ADDRESS-FAMILY.&afi ({IPAddressFamilySet}),
  ipAddresses IP-ADDRESS-FAMILY.&IPAddresses ({IPAddressFamilySet}{@ipFamily}) }

IP-ADDRESS-FAMILY ::= CLASS {
     &afi          OCTET STRING (SIZE(2)) UNIQUE,
     &IPAddresses
   } WITH SYNTAX { AFI &afi IP &IPAddresses }

IPAddressFamilySet IP-ADDRESS-FAMILY ::= { ipAddressFamilyIPv4 | ipAddressFamilyIPv6 }

ipAddressFamilyIPv4 IP-ADDRESS-FAMILY ::= { AFI afi-IPv4 IP IPv4Addresses }

ipAddressFamilyIPv6 IP-ADDRESS-FAMILY ::= { AFI afi-IPv6 IP IPv6Addresses }

afi-IPv4 OCTET STRING ::= '0001'H

afi-IPv6 OCTET STRING ::= '0002'H

IPv4Addresses ::= SEQUENCE (SIZE(1..MAX)) OF IPAddress{ub-IPv4}

IPv6Addresses ::= SEQUENCE (SIZE(1..MAX)) OF IPAddress{ub-IPv6}

ub-IPv4 INTEGER ::= 32

ub-IPv6 INTEGER ::= 128
```

```
IPAddress {INTEGER: ub} ::= BIT STRING (SIZE(0..ub))

END
```

Note that this content appears as the eContent within the encapContentInfo as specified in [RFC6488].

3.1. version

The version number of the SAVNETAttestation that compiles with this specification MUST be 2 and MUST be explicitly encoded.

3.2. asID

The asID field contains the AS number that has deployed SAVNET and can perform SAV on the data plane.

3.3. addresses

The addresses field contains a SEQUENCE of IPFamilyAddresses, which stores the router’s IP addresses within the AS whose ID is asID, which is utilized for establishing SAVNET connections.

3.3.1. Element IPFamilyAddresses

This field contains a SEQUENCE which contains one instance of ipFamily and one instance of ipAddresses.

3.3.1.1. ipFamily

This field contains an OCTET STRING which is either ’0001’H (IPv4) or ’0002’H (IPv6).

3.3.1.2. ipAddresses

This field contains a SEQUENCE of IPAddress instances.

3.3.1.3. Element IPAddress

This element is length bounded through the Information Object Class IP-ADDRESS-FAMILY and its type is a BIT STRING.

4. SiSPI Validation

Before, a relying party can use a SiSPI object to validate the deployment of SAVNET for inter-domain SAV, the relying party MUST first validate the SiSPI object. To validate a SiSPI object, the relying party MUST perform all the validation checks specified in [RFC6488] as well as the following additional specific validation steps of the Signed AS List.

* The contents of the CMS eContent field MUST adhere to all the constraints described in Section 2.

* The AS Identifier Delegation Extension [RFC3779] MUST be present in the end-entity (EE) certificate (contained within the SiSPI object), and the asID in the SiSPI object eContent MUST be contained within the set of AS numbers specified by the EE certificate's AS Identifier Delegation Extension.

* The EE certificate's AS Identifier Delegation Extension MUST NOT contain any ''inherit'' elements.

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

The pseudocode for SiSPI validation is as follows:

```
function ValidateSiSPI(sispiObject, eeCertificate):
    // Step 1: Validate the SiSPI object using the generic RPKI
    //         validation procedure.
    // This includes checking the CMS wrapper, signature, and
    //         certification path.
    if not IsValidRPKISignedObject(sispiObject):
        return False, "Invalid RPKI Signed Object"

    // Step 2: Check the content-type of the SiSPI object.
    if not sispiObject.eContentType == SAVNETAuthzOID:
        return False, "Invalid content-type"

    // Step 3: Parse the eContent of the SiSPI object as
    //         SAVNETAttestation.
    sispiContent = ParseSAVNETAttestation(sispiObject.eContent)
    if sispiContent is None:
        return False, "Unable to parse SAVNETAttestation"

    // Step 4: Ensure the version number is explicitly set to 2.
    if not (sispiContent.version exists and sispiContent.version==2):
        return False, "Invalid version"

    // Step 5: Validate the AS Identifier Delegation Extension in
```

```
      //         the EE certificate.
      if not ValidateASIdExt(eeCertificate, sispiContent.asID):
          return False, "AS Identifier Extension validation failed"

      // Step 6: Ensure the EE certificate's AS Identifier Delegation
      //         Extension does not contain 'inherit'.
      if "inherit" in eeCertificate.asIdentifiers:
          return False,
                 "AS Identifier Delegation Extension contains 'inherit'"

      // Step 7: Ensure the IP Address Delegation Extension is absent.
      if HasIPAddressDelegationExtension(eeCertificate):
          return False, "IP Address Delegation Extension is present"

      // Step 8: Determine if all validation checks are successful.
      return True, "SiSPI object is valid"

  function ValidateASIdentifierExtension(eeCertificate, asID):
      // Check if the asID is within the set of AS numbers
      //   specified by the AS Identifier Delegation Extension.
      return asID in eeCertificate.asIdentifiers

  function HasIPAddressDelegationExtension(eeCertificate):
      // Check for the presence of the IP Address Delegation
      //   Extension.
      return "ipAddresses" in eeCertificate.extensions
```

5. IANA Considerations

5.1. RPKI Signed Object Registry

Please add an item for the SiSPI object file extension to the RPKI Signed Object registry (https://www.iana.org/assignments/rpki/rpki.xhtml#signed-objects) as follows:

Name	OID	Reference
Signed SAVNET-Peering Information	1.2.840.113549.1.9.16.1.TBD	draft-chen-sidrops-sispi

5.2. RPKI Repository Name Scheme Registry

Please add an item for the SiSPI object file extension to the "RPKI Repository Name Scheme" registry created by [RFC6481] as follows:

Filename Extension	RPKI Object	Reference
.sav	Signed SAVNET-Peering Information	draft-chen-sidrops-sispi

5.3. SMI Security for S/MIME Module Identifier (1.2.840.113549.1.9.16.0)

IANA is requested to allocate the following in the "SMI Security for S/MIME Module Identifier (1.2.840.113549.1.9.16.0)" registry:

Decimal	Description	Reference
TBD	id-mod-rpkiSiSPI-2024-2024	draft-chen-sidrops-sispi

5.4. Media Type Registry

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

Type name: application
Subtype name: rpki-sispi
Required parameters: N/A
Optional parameters: N/A
Encoding considerations: binary
Security considerations: Carries Signed SAVNET-Peering Information.
  This media type contains no active content. See
  Section 4 of draft-chen-sidrops-sispi for further information.
Interoperability considerations: None
Published specification: draft-chen-sidrops-sispi
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 an AS identifer
      as defined in draft-chen-sidrops-sispi.
Magic number(s): None
File extension(s): .sav
Macintosh file type code(s):
Person & email address to contact for further information:
Li Chen <lichen@zgclab.edu.cn>
Intended usage: COMMON
Restrictions on usage: None
Change controller: IETF

6. Using SiSPI

A router can use the AS_Path from BGP announcements, ASPA objects, and SiSPI to find the closest ASes to set up SAVNET peering, as described below:

1. BGP AS_Paths Analysis:

    * Collect AS paths from BGP announcements.

* Determine the frequency or preference of certain AS paths based on routing policies, which may involve path attributes like AS path length, origin type, local preference, and MED (Multi-Exit Discriminator).

2. ASPA Verification:

* Use ASPA objects to verify the legitimacy of customer-provider AS relationships.

* Ensure that the AS paths conform to the customer-provider relationships indicated by the ASPAs, thereby validating the correctness of the routing information.

3. Peering Candidates Determination:

* Identify the ASes that frequently appear on the preferred paths to various destinations, implying they are topologically ’close’ or significant transit providers.

* Among these ASes, rank those according to their frequency in an descending order, since the frequency indicates the weight of traffic from the local AS and higher frequency represents more volume of traffic to transmit for the local AS.

4. SiSPI Objects Utilization:

* Retrieve SiSPI objects from the RPKI repository to determine which ASes have deployed SAVNET.

* Filter the previously identified candidate ASes by checking whether they have a valid SiSPI object, which would indicate their readiness to establish SAVNET peering.

5. Peering Candidates Selection:

* From the set of candidate ASes with valid SiSPI objects, select candidates for SAVNET peering based on their rankings.

* The selection criteria may include additional factors such as existing peering policies, traffic volumes, and peering agreements.

6. Peering Establishment:

* Initiate peering negotiations with the selected candidate ASes.

* Upon successful negotiation, establish SAVNET peering relationships and configure the necessary SAVNET protocols.

Based on the above steps, a description of the detailed procedure to establish SAVNET peering relationships is as follows:

1. Let the set of selected AS paths to all the potential destinations be denoted as ASPaths.

2. Let i = 1. Validate ASPaths(i) using ASPA objects.

3. Let the set of validated AS paths be denoted as ASPaths-V.

4. If ASPaths(i) passes the validation of ASPA objects, add it to ASPaths-V.

5. Increment i to i+1.

6. If ASPaths(i) is null, then set i_max = i - 1 and go to Step 7. Else, go to Step 4.

7. Let j = 1 and k = 1. Initialize AS-set S(1) = ASPaths-V(1)(1) and N(ASPaths-V(1)(1)) = 1.

8. If ASPaths-V(j)(k) belongs to S, N(ASPaths-V(j)(k)) = N(ASPaths-V(j)(k)) + 1. Else, N(ASPaths-V(j)(k)) = 1 and S(J * k) = ASPaths-V(j)(k).

9. Increment k to k+1.

10. If ASPaths-V(j)(k) is null, then go to Step 11. Else, go to Step 8.

11. Increment j to j+1.

12. If ASPaths-V(j)(k) is null, then go to Step 13. Else, go to Step 8.

13. Rank the AS-set N according to its values in descending order.

14. Retrieve SiSPI objects from the RPKI repository and let the set of ASes within the SiSPI objects be denoted as O.

15. Let m = 1. Create a SAVNET neighbor candidate set C.

16. If N(m) belongs to O, add N(1) to C.

17. Increate m to m + 1.

18. If N(m) is null or the number of ASes in set C exceeds 4000, go to Step 19. Else, go to Step 16.

19. Establish SAVNET peering relationship with the selected candidate ASes in set C.

7. Newly SAVNET-adopting ASes

The newly SAVNET-adopting ASes need to register the SiSPI object proactively to help other SAVNET-adopting ASes find it and establish SAVNET peering relationships, as well as using the SiSPI objects to establish SAVNET peering relationships with other SAVNET-adopting ASes.

To register the SiSPI object, the newly SAVNET-adopting ASes should share its information as described in Section 3.

To establish SAVNET peering relationships with other SAVNET-adopting ASes, the newly SAVNET-adopting ASes should collect BGP announcements, ASPA objects, and SiSPI objects, and run the procedures described in Section 6.

8. Security Considerations

The security considerations of [RFC6481], [RFC6485], and [RFC6488] also apply to the SiSPI object.

9. References

9.1. Normative References

[RFC3704] Baker, F. and P. Savola, "Ingress Filtering for Multihomed Networks", BCP 84, RFC 3704, DOI 10.17487/RFC3704, March 2004, <https://www.rfc-editor.org/rfc/rfc3704>.

[RFC8704] Sriram, K., Montgomery, D., and J. Haas, "Enhanced Feasible-Path Unicast Reverse Path Forwarding", BCP 84, RFC 8704, DOI 10.17487/RFC8704, February 2020, <https://www.rfc-editor.org/rfc/rfc8704>.

[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/rfc/rfc6480>.

[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/rfc/rfc6488>.

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

[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/rfc/rfc3779>.

[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/rfc/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/rfc/rfc6485>.

[RFC5280] Cooper, D., Santesson, S., Farrell, S., Boeyen, S., Housley, R., and W. Polk, "Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile", RFC 5280, DOI 10.17487/RFC5280, May 2008, <https://www.rfc-editor.org/rfc/rfc5280>.

[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/rfc/rfc6487>.

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

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

[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/rfc/rfc2119>.

[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/rfc/rfc8174>.

9.2. Informative References

[inter-domain-ps]
"Source Address Validation in Inter-domain Networks Gap Analysis, Problem Statement, and Requirements", 2024, <https://datatracker.ietf.org/doc/draft-ietf-savnet-inter-domain-problem-statement/>.

[savnet] "Inter-domain Source Address Validation (SAVNET) Architecture", 2024, <https://datatracker.ietf.org/doc/draft-wu-savnet-inter-domain-architecture/>.

[emu-9-savs]
"Emulations of 9 SAV Mechanisms with SAV Open Playground", 2023, <https://datatracker.ietf.org/meeting/118/materials/slides-118-savnet-emulations-of-nine-sav-mechanisms-with-sav-open-playground-00>.

Authors' Addresses

Li Chen
Zhongguancun Laboratory
Beijing
China
Email: lichen@zgclab.edu.cn

Libin Liu
Zhongguancun Laboratory
Beijing
China
Email: liulb@zgclab.edu.cn

Dan Li
Tsinghua University
Beijing
China
Email: tolidan@tsinghua.edu.cn

Lancheng Qin
Zhongguancun Laboratory
Beijing
China
Email: qinlc@zgclab.edu.cn