PDF Profile for Signature Validation Tokens

Version 1.0 - 2020-12-16 - *Draft version*

Registration number: **2020-61**

Table of Contents

1. Introduction

The "Signature Validation Token" specification [SVT] defines a basic token to support signature validation in a way that can significantly extend the lifetime of a signature.

This specification defines a profile for implementing SVT with a signed PDF document, and defines the following aspects of SVT usage:

- How to include reference data related to PDF signatures and PDF documents in an SVT.
- How to add an SVT token to a PDF document.

PDF document signatures are added as incremental updates to the signed PDF document and signs all data of the PDF document up until the current signature. When more than one signature is added to a PDF document the previous signature is signed by the next signature and can not be updated with additional data after this event.

To minimize the impact on PDF documents with multiple signatures and to stay backwards compatible with PDF software that do not understand SVT, PDF documents add one SVT token for all signatures of the PDF as an extension to a document timestamp added to the signed PDF as an incremental update. This SVT covers all signatures of the signed SVT.

1.1. Requirements Notation

The key words **MUST**, **MUST NOT**, **REQUIRED**, **SHALL**, **SHALL NOT**, **SHOULD**, **SHOULD 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.

These keywords are capitalized when used to unambiguously specify requirements over protocol features and behavior that affect the interoperability and security of implementations. When these words are not capitalized, they are meant in their natural-language sense.

1.2. Definitions

The definitions in [SVT] apply also to this document.

2. SVT in PDF Documents

An SVT added to a signed PDF document SHALL be added to a document timestamp accordance with ISO 32000-2:2017 [PDF].

The document timestamp contains an RFC 3161 timestamp token (TSTInfo) in EncapsulatedContentInfo of the CMS signature. The SVT SHALL be added to the timestamp token (TSTInfo) as an Extension object as defined in section 2.1.

2.1. SVT Extension to Timestamp Tokens

The SVT extension is an Extension suitable to be included in TSTInfo as defined by [RFC3161][*].

The SVT extension is identified by the Object Identifier (OID) `1.2.752.201.5.2` as defined in [EidRegistry].

This extension data (OCTET STRING) holds the bytes of SVT JWT, represented as a UTF-8 encoded string.

This extension SHALL NOT be marked critical.

> [*]: Extensions in timestamp tokens according to [RFC3161] are imported from the definition of the X.509 certificate extensions defined in [RFC5280].

3. SVT Claims

3.1. Signature Reference Data

The SVT SHALL contain a **SigReference** claims object that SHALL contain the following data:

Claim	Value
`id`	Absent or a **Null** value.
`sig_hash`	The hash over the signature value bytes.
`sb_hash`	The hash over the DER encoded `SignedAttributes` in `SignerInfo`.

3.2. Signed Data Reference Data

An SVT according to this profile SHALL contain exactly one instance of the **SignedData** claims object. The **SignedData** claims object shall contain the following data:

Claim	Value
`ref`	The string representation of the **ByteRange** value of the PDF signature dictionary of the target signature. This is a sequence of integers separated by space where each integer pair specifies the start index and length of a byte range.
`hash`	The hash of all bytes identified by the **ByteRange** value. This is the concatenation of all byte ranges identified by the **ByteRange** value.

3.3. Signer Certificate References

The SVT SHALL contain a **CertReference** claims object. The `type` claim of the **CertReference** claims object SHALL be either `cert`, `chain`, `cert_hash` or `cert_and_chain_hash`.

- The `chain` type SHALL be used when signature validation was performed using one or more certificates where some or all of the certificates in the chain are not present in the target signature.
- The `chain_hash` type SHALL be used when signature validation was performed using one or more certificates where all of the certificates are present in the target signature.

Note: The `cert` type MUST NOT be used with a PAdES signatures (SubFiler in the signature dictionary is set to "ETSI.CAdES.detached") where the signing certificate in the target signature is bound to the signature through ESSCertID or ESSCertIDv2 [RFC5035].

4. JOSE Header

4.1. SVT Signing Key Reference

The SVT JOSE header must contain one of the following header parameters in accordance with [RFC7515], for storing a reference to the public key used to verify the signature on the SVT:

Header Parameter	Value
`x5c`	Holds an X.509 certificate [RFC5280] or a chain of certificates. The certificate holding the public key that verifies the signature on the SVT MUST be the first certificate in the chain.
`kid`	A key identifier holding the Base64 encoded hash value of the certificate that can verify the signature on the SVT. The hash algorithm MUST be the same hash algorithm used when signing the SVT as specified by the `alg` header parameter. The referenced certificate SHOULD be the same certificate that was used to sign the document timestamp that contains the SVT.

5. Normative References

[RFC2119]

> Bradner, S., Key words for use in RFCs to Indicate Requirement Levels, March 1997.

[RFC3161]

> Adams, C., Cain, P., Pinkas, D., Zuccherato, R., Internet X.509 Public Key Infrastructure Time-Stamp Protocol (TSP), August 2001.

[RFC5280]

> D. Cooper, S. Santesson, S. Farrell, S. Boeyen, R. Housley, W. Polk, Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile, May 2008.

[RFC5035]

> Shaad, J., Enhanced Security Services (ESS) Update: Adding CertID Algorithm Agility, August 2007.

[RFC7515]

> Jones, M., Bradley, J., Sakimura, N., JSON Web Signature (JWS), May 2015.

[RFC8174]

> Leiba, B., Ambiguity of Uppercase vs Lowercase in RFC 2119 Key Words, May 2017.

[PDF]

> ISO 32000-2:2017, Document management - Portable Document Format - Part 2: PDF 2.0, July 2017.

[EidRegistry]

> Swedish eID Framework - Registry for identifiers.

[SVT]

> Signature Validation Token.