MitSoft

Qualified Long-term Preservation Service

Preservation Profile Without Storage

QLPS/PP-WOS

Unique object ID (OID): **1.3.6.1.4.1.57890.1.5.2**

Document version 1.02

Valid since 2023-05-15

UAB „MIT-SOFT", Kalvarijų st. 276-100, LT-08316 Vilnius, Tel. +370 233 3922, Fax +370 5 213 6191, www.mitsoft.lt, info@mitsoft.lt, Company ID 120792080, VAT payer code LT207920811

Approvals

Revision history

Version	Valid since	Description
1.00		First official version of the document
1.01		Fixed minor wording errors
1.02	2023-05-15	CMS signatures in PDF are permitted; clarification for non-valid digital signature preservation

Approval of the document

	Name	Date	Signature
Reviewed by	Adomas Birštunas	2023-04-04	
Approved by	Antanas Mitašiūnas	2023-05-15	

Table of content

1. Introduction

The joint stock company "MIT-SOFT" (further – the MitSoft) is qualified long-term preservation service provider (further – PSP).

This document defines one of the preservation profiles supported by the MitSoft PSP. The name of the preservation profile defined in the current document is as follows:

- MitSoftQWOS profile – MitSoft electronic document without storage preservation profile for qualified signatures and seals.

MitSoftQWOS profile should be used when qualified electronic signatures, advanced electronic signatures, qualified electronic seals or advanced electronic seals contained within electronic documents or containers are stored outside MitSoft PSP (in the subscriber's storage), but MitSoft PSP preservation services are used to assure electronic signatures and/or electronic seals preservation.

Preservation storage model, preservation goals, operations supported by the MitSoftQWOS profile and preservation protocol to be used are further described in this document.

2. References

[ADOC-V1.0] – Lietuvos vyriausiasis archyvaras. "Elektroniniu parašu pasirašyto elektroninio dokumento specifikacija ADOC-V1.0".

[ADOC-V2.0] – Lietuvos vyriausiasis archyvaras. "Elektroninio dokumento specifikacija ADOC-V2.0".

[EGAS-V1.0] – Valstybinio socialinio draudimo fondo valdybos prie Socialinės apsaugos ir darbo ministerijos direktoriaus įsakymas "Dėl elektroninės gyventojų aptarnavimo sistemos naudojimo taisyklių ir elektroninės gyventojų aptarnavimo sistemos elektroniniu parašu pasirašyto dokumento specifikacijos EGAS-V1.0 patvirtinimo".

[eIDAS] – Regulation (EU) No 910/2014 of the European Parliament and of the Council on electronic identification and trust services for electronic transactions in the internal market and repealing Directive 1999/93/EC.

[EN 319 122-1] – ETSI EN 319 122-1: "Electronic Signatures and Infrastructures (ESI); CAdES digital signatures; Part 1: Building blocks and CAdES baseline signatures".

[EN 319 132-1] – ETSI EN 319 132-1: "Electronic Signatures and Infrastructures (ESI); XAdES digital signatures; Part 1: Building blocks and XAdES baseline signatures".

[EN 319 142-1] – ETSI EN 319 142-1: "Electronic Signatures and Infrastructures (ESI); PAdES digital signatures; Part 1: Building blocks and PAdES baseline signatures".

[EN 319 162-1] – ETSI EN 319 162-1: "Electronic Signatures and Infrastructures (ESI); Associated Signature Containers (ASiC); Part 1: Building blocks and ASiC baseline containers".

[MDOC-V1.0] – Lietuvos vyriausiasis archyvaras. "Elektroniniu parašu pasirašyto kompiuterio skaitomo elektroninio dokumento specifikacija MDOC-V1.0".

[PDF-LT-V1.0] – Lietuvos vyriausiasis archyvaras. "Elektroninio dokumento PDF-LT-V1.0 specifikacija".

[PDF-RC-V1.0] – Valstybės įmonė Registrų centras. „Elektroninio dokumento specifikacija PDF-RC-V1.0".

[DSS Core 2.0] - OASIS: "Digital Signature Service Core Protocols, Elements, and Bindings Version 2.0", Committee Specification 02, 11 December 2019.

[TS 101 733] – ETSI TS 101 733: "Electronic Signatures and Infrastructures (ESI); CMS Advanced Electronic Signatures (CAdES)".

[TS 101 903] – ETSI TS 101 903: "Electronic Signatures and Infrastructures (ESI); XML Advanced Electronic Signatures (XAdES)".

[TS 102 778-2] - ETSI TS 102 778-2: "Electronic Signatures and Infrastructures (ESI); PDF Advanced Electronic Signature Profiles; Part 2: PAdES Basic - Profile based on ISO 32000-1".

[TS 102 778-4] – ETSI TS 102 778-4: "Electronic Signatures and Infrastructures (ESI); PDF Advanced Electronic Signature Profiles; Part 4: PAdES Long Term - PAdES-LTV Profile".

[TS 103 171] – ETSI TS 103 171: "Electronic Signatures and Infrastructures (ESI); XAdES Baseline Profile".

[TS 103 172] – ETSI TS 103 172: "Electronic Signatures and Infrastructures (ESI); PAdES Baseline Profile".

[TS 103 173] – ETSI TS 103 173: "Electronic Signatures and Infrastructures (ESI); CAdES Baseline Profile".

[TS 103 174] – ETSI TS 103 174: "Electronic Signatures and Infrastructures (ESI); ASiC Baseline Profile".

[TS 119 312] – ETSI TS 119 312 "Electronic Signatures and Infrastructures (ESI); Cryptographic Suites" standard.

[TS 119 512] – ETSI TS 119 512: "Electronic Signatures and Infrastructures (ESI); Protocols for trust service providers providing long-term data preservation services".

3. Definitions of terms and abbreviations

Compromise: a loss, theft, modification, illegal use, or any other security violation of the confidential data.

Container: data object, which contains a set of data objects and optional additional information, which describes the contained data objects and optionally its content and its interrelationships.

Data object: actual binary/octet data being operated on (e.g. transformed, digested, or signed) by an application and which may be associated with additional information like an identifier, the encoding, size or type.

Expected evidence duration: for a preservation service with temporary storage or without storage, duration during which the preservation service expects that the preservation evidence can be used to achieve the preservation goal.

Long-term: time period during which technological changes may be a concern.

Long-term preservation: extension of the validity status of a digital signature over long periods of time and/or extension of provision of proofs of existence of data over long periods of time, in spite of obsolescence of cryptographic technology such as crypto algorithms, key sizes or hash functions, key compromises or of the loss of the ability to check the validity status of public key certificates.

Preservation client: component or a piece of software which interacts with a preservation service via the preservation protocol.

Preservation evidence: evidence produced by the preservation service which can be used to demonstrate that one or more preservation goals are met for a given preservation object.

Preservation evidence policy: set of rules that specify the requirements and the internal process to generate or how to validate a preservation evidence.

Preservation goal: one of the following objectives achieved during the preservation time frame: extending over long periods of time the validity status of digital signatures, providing proofs of existence of data over long periods of time, or augmenting externally provided preservation evidences.

Preservation interface: component implementing the preservation protocol on the side of the preservation service.

Preservation object: typed data object which is submitted to, processed by or retrieved from a preservation service.

NOTE: This covers submission data objects, preservation object containers and preservation evidences.

Preservation object container: container which contains a set of data objects and optionally related metadata providing information about the data objects and optionally preservation manifest(s) specifying its content and relationships.

Preservation object identifier: unique identifier of a (set of) preservation object(s) submitted to a preservation service.

Preservation period: for a preservation service with storage, duration during which the preservation service preserves the submitted preservation objects and the associated evidences.

Preservation profile: uniquely identified set of implementation details pertinent to a preservation storage model and one or more preservation goals which specifies how preservation evidences are generated and validated.

Preservation protocol: protocol to communicate between the preservation service and a preservation client.

Preservation scheme: generic set of procedures and rules pertinent to a preservation storage model and one or more preservation goals which outlines how preservation evidences are created and validated.

NOTE: Different preservation profiles can implement the same preservation scheme.

Preservation service: service capable of extending the validity status of a digital signature over long periods of time and/or of providing proofs of existence of data over long periods of time.

Preservation service provider: trust service provider providing a preservation service.

Preservation storage model: one of the following ways of implementing a preservation service: with storage, with temporary storage, without storage.

Proof of existence: evidence that proves that an object existed at a specific date/time.

Signature augmentation: process of incorporating to an electronic signature or electronic seal information aiming to maintain the validity of that signature/seal over the near term and/or the long term

Signature validation constraint: technical criteria against which an electronic signature or electronic seal can be validated

Signature policy: signature creation policy, signature augmentation policy, signature validation policy or any combination thereof, applicable to the same signature or set of signatures.

Submission data object: original data object provided by the client.

Subscriber: legal or natural person bound by agreement with a preservation trust service provider to any subscriber obligations.

Other definitions are used as in Regulation (EU) 910/2014 [eIDAS].

ETSI	–	European Telecommunications Standards Institute
OID	–	Object identifier
PO	–	Preservation object
POC	–	Preservation object container
PSP	–	Preservation service provider
SubDO	–	Submission data object

4. MitSoft electronic document without storage preservation profile for qualified signatures and seals

4.1. Description

MitSoftQWOS profile defines operational details for the MitSoft preservation service for the case it is used without integrated electronic document storage. Preserved objects are stored outside preservation service (in some subscriber storage). The goal of the preservation service using MitSoftQWOS profile is preservation over long periods of time of the ability to validate electronic signatures and electronic seals, maintenance their validity status and getting proofs of existence of the associated signed data.

Preservation objects to be preserved by preservation service are qualified electronic signatures and qualified electronic seals. Preservation service preserves advanced electronic signatures and advanced electronic seals (that are not qualified) only if such a requirement is listed in the Subscriber agreement. Further in this document they are called electronic signatures or seals, or just digital signatures.

Preservation service preserves valid electronic signatures and valid electronic seals. Non-valid electronic signatures and non-valid electronic seals are preserved only if such a requirement is listed in the Subscriber agreement and it is allowed according signature policy.

Preservation service does not accept electronic signatures or electronic seals that are not part of some submitted electronic document or container. Every submitted electronic document or container shall contain actual data signed by the preserved electronic signature or sealed by the preserved electronic seal (subscribers are not allowed to provide only hash values of the signed data). The supported electronic document specifications and container standards are listed in the section 4.12. Preservation objects to be preserved by this profile are defined in the section 4.8.

This profile should be used when signed/sealed electronic documents/containers are stored in the subscriber storage, but preservation is ensured by the MitSoft preservation service. Subscriber is responsible for storing and maintenance of the electronic signatures and electronic seals presented within stored electronic documents/containers. Subscriber is also responsible for the timely calls to MitSoft preservation service in order to get electronic signatures/seals augmented with preservation evidences and to get expected evidence durations. Preservation service is responsible for the preservation of the provided electronic signatures and electronic seals. Preservation service ensures electronic signatures/seals preservation by the means of their validation, collection of validation data and preservation evidences, electronic signatures/seals augmentation and estimation of the expected evidence durations. Since preservation service does not store provided electronic documents/containers, according to this profile, preservation service does not keep track of when preservation actions should be performed.

MitSoftQWOS profile offers operations for:

- performing preservation actions (validation and augmentation by including preservation evidences) for electronic signatures and electronic seals within provided electronic documents/containers (see Augment, PreservePO operations).

4.2. Identification

The unique identifier (OID) of the MitSoftQWOS profile is as follows:

- identifier presented as OID is
  - 1.3.6.1.4.1.57890.1.5.2;

  the values of its fields are given in the Table 1.
- Identifier presented as URI is

- http://uri.mitsoft.lt/preservation/profile/qwos/2

Table 1. The values of the fields of the unique identifier of the MitSoftQWOS profile

Name	Value
ISO	1
Organization recognized by ISO	3
U.S. Department of Defence	6
Internet	1
Private company	4
Private company registered by IANA	1
Joint stock company "MIT-SOFT"	57890
Subdivision MitSoft	1
Document type (preservation profile)	5
Preservation profile identifier	2

4.3. Validity period

The present preservation profile become active from 2022-12-01. The same preservation profile is applied during the whole preservation period.

4.4. Preservation scheme

The MitSoftQWOS profile implements the following preservation scheme:

- preservation scheme with signature augmentation and without storage, which is defined in Annex F.4 of ETSI TS 119 512 [TS 119 512] and indicated by the identifier:
  - http://uri.etsi.org/19512/scheme/pds+wos+aug

4.5. Preservation goal

The MitSoftQWOS profile supports the following preservation goal:

- PDS - extending over long periods of time the validity status of digital signatures ETSI TS 119 512 [TS 119 512], which is indicated by the URI:
  - http://uri.etsi.org/19512/goal/pds

4.6. Preservation storage model

The MitSoftQWOS profile supports preservation service without storage – WOS according to ETSI TS 119 512 [TS 119 512].

4.7. Supported operations

The present preservation profile supports the following operations:

- Augment,
- Download,
- RetrieveInfo,
- PreservePO.

4.8. Submission data objects and preservation objects

MitSoftQWOS preserves electronic signatures and electronic seals contained in some electronic document or container.

Submission data object (SubDO) is an electronic document/container, presented as one file, which:

- contains at least one electronic signature or electronic seal,
- contains every data object signed by electronic signature or sealed by electronic seal, and
- optionally contains its own metadata.

Supported SubDOs are defined in the section 4.12.

Preservation object (PO) preserved by the MitSoft PSP is an electronic signature or an electronic seal of the B-LTA signature level (or corresponding archival signature format for non-baseline signatures).

Preservation service preserves qualified electronic signatures and qualified electronic seals. Preservation service preserves advanced electronic signatures and advanced electronic seals (that are not qualified) only if such a requirement is listed in the Subscriber agreement.

Preservation service preserves valid electronic signatures and valid electronic seals (having validation status PASSED). Non-valid electronic signatures and non-valid electronic seals (having validation status other than PASSED) are preserved only if validation result does not contain any validation error, which prevents digital signature from the augmentation according signature policy, and if such a requirement is listed in the Subscriber agreement. The precise list of validation error subindications, that do not prevent digital signature from the augmentation, is listed in the signature policy.

Preservation object preserved by the MitSoft PSP is derived from the submission data object by augmenting electronic signature or electronic seal presented within SubDO. Augmentation is performed to reach B-LTA signature level (or corresponding archival signature format for non-baseline signatures). Augmentation involves getting validation data and time stamps, and their inclusion into digital signature. Preservation of the electronic signature or electronic seal is not performed in the case when B-LTA signature level (or corresponding archival signature format for non-baseline signatures) cannot be reached (caused by its invalidity, or by impossibility to get reliable missing validation data).

In the case when submission data object does not contain any electronic signature or electronic seal to be preserved, no preservation actions will be performed with such a SubDO, but some validation data may be included into digital signatures. This allows to not lose collected validation data, which is a possible invalidity proof of the provided digital signature.

Preservation of electronic signatures and electronic seals is assured by the new preservation evidence (archival time stamp) inclusion. In the case when SubDO already contains B-LTA level electronic signature or electronic seal, new preservation evidence (new archival time stamp) will be included if only augmentation period is already started (see section 4.8.2). Otherwise, new preservation evidence will not be included into PO and only new information about next augmentation time (including calculated expected evidence duration) will be returned. Expected evidence duration is calculated by preservation service and provided to client as a result of the preservation operation. See section 4.8.1 for the details.

In the context of electronic documents, digital signature usage (and preservation) without electronic document/container it belongs to (and actually signed) is meaningless. Therefore, every operation deals with preservation object container instead of separate preservation object preserved by the MitSoft PSP.

Preservation object container (POC) is a submitted electronic document or container, which contains electronic signatures and/or seals (preservation objects) with preservation

evidences (time stamps) included by the preservation service. POC is derived from the SubDO, by augmenting preserving digital signatures and preservation evidences (time stamps) inclusion. In the special case, when there are no preserving digital signatures in the SubDO, POC may be the same unmodified SubDO.

Operations defined by this profile accepts and returns preservation object, which is a whole preservation object container - electronic document or container (together with all preserving digital signatures and preservation evidences within it), or submission data object. Separate electronic signature or seal cannot be accessed using this profile operations.

4.8.1. Expected evidence duration

According MitSoftQWOS profile, submission data objects and preservation objects are stored in some client's storage and, therefore, client is responsible for maintenance of the preservation objects and triggering preservation actions. To enable client to determine the proper time of the next preservation action, MitSoft PSP will calculate expected evidence duration of the preservation objects.

Expected evidence duration is calculated for every preserving electronic signature, or electronic seal. Since, preservation is performed only for digital signatures having B-LTA signature level (or corresponding archival signature format for non-baseline signatures), expected evidence duration is calculated according signing certificate of the last valid archival time stamp and cryptographic algorithms used for its creation.

Expected evidence duration is limited by one of the following (which is earlier):

- Expiration date of the time stamping authority certificate used to sign the last valid archival time stamp,
- Minimal expected reliability period of the cryptographic algorithms used by the last valid archival time stamp.

Expected cryptographic algorithm reliability period is maintained by the preservation service cryptographic monitoring. Section 7.14 of the MitSoft PSP Practice statement contains details how expected cryptographic algorithm reliability period is determined and maintained by preservation service cryptographic monitoring.

If aggregated expected evidence duration for the whole POC (electronic document or container) should be determine, it should be a minimal expected evidence duration calculated according to every preserving electronic signature/seal contained in the POC. Note, that in the normal case, the same time stamping authority and the same cryptographic algorithms will be used for archival time stamps creation, and, therefore, expected evidence duration for every preserving digital signature usually will be the same.

Calculated expected evidence duration is returned as one of the result fields of the preservation operations *Augment* (*AugmentNotAfter* field, see section 4.9.2) and *PreservePO* (*eed* attribute, see section 4.9.4).

4.8.2. Preservation event time determination

Preservation actions should be performed time by time to ensure PO validity during its preservation period. Since PSP client stores PO in its own storage (according MitSoftQWOS profile), it is client's responsibility to trigger preservation action on the proper time. The proper time should be neither too early nor too late. Preservation action is too late if digital signature validity already expired, or there is a risk of not being able to extend digital signature validity due to a large number of preservation actions to be performed and/or temporary third-party service unavailability. Preservation action is too early if new preservation evidence (archival time stamp) inclusion into digital signature is redundant. Preservation service helps clients to plan a proper time for the next preservation action. The proper time of the next preservation action is defined using expected evidence duration, augmentation period and caution period.

Preservation evidence validity is limited by the time stamping authority certificate expiration date and expected reliability period of cryptographic algorithms. Since a big amount of preserving digital signatures will contain preservation evidences with the same validity period, it is important to ensure, that every such a digital signature will be augmented (preserving action to be performed) before its validity expiration time. Therefore, digital signatures should be augmented before caution period – some time before validity expiration. Preservation service does not guarantee that provided digital signature will be successfully upgraded with a new preservation evidence if caution period is already started. Caution period used by the MitSoftQWOS profile is indicated in Subscriber Agreement.

In the case when provided SubDO already contains B-LTA level digital signature, which will not expire in the near future, new archival time stamp inclusion into digital signature may be redundant. Therefore, new preservation evidence will be included into the digital signature only if augmentation period is already started. Augmentation period is a reasonable time period, which starts some time before expected digital signature validity expiration and lasts till the expected digital signature validity expiration date. The exact used augmentation period is indicated in the Subscriber agreement.

The following elements (among others) will be returned as a result of the preservation operations:

- *AugmentNotBefore* – start date of the augmentation period; new preservation evidence will be included into the digital signature if only *AugmentNotBefore* is prior to current time,
- *AugmentAt* – recommended time for the next preservation event; it is some time inside augmentation period, but before start of the caution period,
- *AugmentNotAfter* – it is the end of the augmentation period, which is equal to the end of the expected evidence duration; digital signature may lose its validity if no successful preservation action will be performed till *AugmentNotAfter* time.

It is recommended, that the next preservation action with the same PO to be performed at (or near) the time recommended by the previous preservation operation execution (*AugmentAt*).

Note, that expected reliability period of the cryptographic algorithms may be extended (or shortened) over time. Augmentation period and expected evidence duration returned by the preservation operation may differ from the ones returned in the previous operation results. Operation executed at recommended time (*AugmentAt*) returned by the previous operation execution may extend digital signature validity by new preservation evidence inclusion, or may just calculate and return updated augmentation period and new recommended augmentation time.

4.9. Preservation protocol

The present section describes semantics and syntax of the supported operations. Operations are implemented as REST web services. Web service caller authentication shall be used and communication encryption (Secure Sockets Layer) shall be applied.

Operations originally defined in the ETSI TS 119 512 are shortly introduced in this section using references to the standard and specifying specific issues only. Other specific operations semantics and syntax are defined in this section. Only the main fields are described, field names are unformal, types and field cardinalities are omitted. The exact preservation interface - requests and responses fields - is described separately using JSON notation (see section 4.9.5).

MitSoftQWOS profile supports the following operations:

4.9.1. RetrieveInfo

RetrieveInfo operation is used to get the set of supported preservation profiles. Operation returns the list of the supported preservation profiles.

RetrieveInfo operation syntax and semantics are fully described in the ETSI TS 119 512 section 5.3.2. Current profile supports this operation only used with JSON syntax.

4.9.2. Augment

Augment operation is an extended analogue of the PreservePO operation defined in the ETSI TS 119 512. Augment operation is used to perform preservation of the provided preservation object, which is some new submission data object, or preservation object container previously obtained by this operation. The provided preservation object is an electronic document or container containing electronic signatures or electronic seals. For this operation preservation object shall be provided. Preservation is performed by augmenting electronic signatures and electronic seals. It involves creation/collection of the validation data (in the case it is missing) and preservation evidences and their inclusion into the preservation objects – electronic signatures or electronic seals.

If an electronic signature or seal already reached B-LTA signature level (or corresponding archival signature format for non-baseline signatures) operation behaviour depends on augmentation and caution periods (see section 4.8.2):

- If operation is executed before augmentation period (expected digital signature validity expiration time is far in the future), new preservation evidence (archival time stamp) will not be included and only augmentation period, expected evidence duration and recommended next augmentation time will be calculated and returned,
- If operation is executed within augmentation period and before caution period (expected digital signature validity expiration time is in the near future), new preservation evidence (archival time stamp) will be included and new augmentation period, expected evidence duration and recommended next augmentation time will be calculated and returned,
- If operation is executed within caution period or after augmentation period (digital signature validity may be already expired), preservation service will try to include new preservation evidence (archival time stamp) if only digital signature validity is not expired yet, and on success, new augmentation period, expected evidence duration and recommended next augmentation time will be calculated and returned.

If an electronic signature or seal has B-T or B-LT signature level, operation tries to collect proper validation data:

- If proper validation data is unavailable at current time (i.e., revocation data is too old due to grace period), then new preservation evidence (archival time stamp) will not be included and only recommended next augmentation time reflecting the grace period will be returned.
- If proper validation data is available at current time, then collected validation data and new preservation evidence (archival time stamp) will be included and new augmentation period, expected evidence duration and recommended next augmentation time will be calculated and returned.

If an electronic signature or seal has B-B signature level, then, at first, Augment operation execution will include proof of existence for signature value (signature time stamp) and after its behaviour is the same as for B-T level (see above).

This operation supports submission data objects of the formats listed in the Section 4.12. SubDOs and preservation objects derived from them are not stored in PSP storage, but information of the performed actions (traces) are stored. In the traces,

SubDO and preservation object will be identified using *DocumentTicket* element provided by this operation.

Current profile supports this operation used with JSON syntax. JSON schema file and OpenAPI Document are presented in the section 4.9.5.

4.9.2.1 Augment Request

The *Augment* request shall accept the preserved object content and additional parameters.

The request shall include the following fields:

- The *client ID*. It shall contain the identifier of the preservation client which performs the preserved object augmentation call.
- The *document file*. It shall contain the content of SubDO submitted as a file. The content shall be provided in its original format without encoding or transformation.

The request may contain the following fields:

- The optional *document specification*. If present, it shall contain the identifier of document specification defining the type of SubDO content. The identifier represents one of specifications listed in section 4.12. If this parameter is omitted, then the SubDO type shall be determined from its file name extension or content.
- The optional *signature filters*. If present, it shall contain filters identifying electronic signatures or electronic seals to be processed by this augmentation operation. The filters can be expressed as a list of digital signatures identifiers within the SubDO, or as a list of signature purposes, if applicable.
- The optional *time to live*. If present, it shall contain the time in seconds for preserved object to be kept in preservation service memory after augmentation execution before it is retrieved by the preservation client. After this period of time the preserved object is disposed by the preservation service and no longer available for retrieval by the preservation client.

The fields above shall be implemented as the following HTTP multi-part POST request parameters:

Fields	Description	HTTP request parameters
Client ID	Identifier of the preservation client	cid
Document file	The content of submission data object	docFile
Document specification	Identifier of specification defining the type of SubDO content	docSpecId
Signatures filters	List of digital signatures or types of signatures in SubDO to be processed during augmentation	sigIds, sigPurposes
Time to live	Time in seconds for preservation object to be kept in archive system memory after augmentation	ttl

4.9.2.2 Augment Response

The *Augment* response contains status of the operation execution, whether the operation was successful or not, and errors in the case of failure.

On the success, HTTP response with status code 200 is returned and it contains status information about the performed augmentation.

In case of an error, including failures during submitted SubDO processing, HTTP response with status code different from 200 is returned and it contains error message.

The successful response shall contain the following elements:

- The *AugmentationStatus* element. It shall contain the value, indicating if electronic signatures and/or electronic seals of preservation object are augmented or unchanged.
- The *DocumentSpecificationId* element. It shall contain the identifier of document specification defining the type of SubDO content. The identifier represents one of specifications listed in section 4.12.
- The *AugmentAt* element. It shall contain the date/time value indicating when next augmentation of preserved object needs to be executed. The value of this element shall be *null,* if no next augmentation time can be determined during this augmentation execution. It is calculated using AugmentAt elements returned for separate digital signatures to be preserved within PO. It should be used as a recommended time for the next preservation action.
- The *DocumentAvailableUntil* element. It shall contain date/time value until which preservation object is kept in the memory of preservation service before it is retrieved by the preservation client. After this moment the preservation object is disposed and no longer available for retrieval. The value of this element shall be *null*, if preservation object is not updated during augmentation and it is not available for retrieval for the preservation client.
- The *DocumentTicket* element. It shall contain identifier associated to this preservation object by the preservation service. It shall be used by the preservation client to retrieve preservation object after some or all its electronic signatures or electronic seals are augmented. In order to retrieve the content of preservation object after augmentation, the preservation client will call the *Download* operation immediately after the receiving *Augment* response, but no later than the moment indicated by *DocumentAvailableUntil* element.

  Note: This element is also used as a preservation object (SubDO and preservation object container derived from it) identifier in the traces of the performed preservation action.
- The *Signatures* elements. It shall contain the array of *Signature* elements. The elements of this array shall represent all electronic signatures and electronic seals present in preservation object.

The *Signature* component shall contain the following elements:

- The *AugmentationValidation* element. It shall contain the code, indicating the status of electronic signature or electronic seal validation, performed during its augmentation.
- The *AugmentNotBefore* element. It shall contain the date/time value indicating start date of the next augmentation period. New preservation evidence will be included into digital signature if operation is performed within augmentation period, therefore, next operation execution (preservation action) should be performed not before time returned by this element. The value of this element shall be *null,* if it is digital signature that preservation is not performed, or B-LTA signature level (or corresponding archival signature format for non-baseline signatures) was not reached.
- The *AugmentAt* element. It shall contain date/time value indicating when the next augmentation of this digital signature needs to be executed. The value of this element shall be *null*, if no next augmentation time for this signature can be determined during this augmentation execution. It should be used as a recommended time for the next preservation action to be called by the client.

- The *AugmentNotAfter* element. It shall contain the date/time value indicating the expected digital signature expiration time, which also indicates the end of the calculated expected evidence duration for this digital signature. The value of this element shall be *null,* if the preservation of digital signature is not performed, or B-LTA signature level (or corresponding archival signature format for non-baseline signatures) was not reached.
- The *Info* element. It shall contain electronic signature or electronic seal information: information on signer certificate (certificate issuer, certificate subject, and validity date/time), signing time, signature timestamp, if present, signature level and format.
- The *Validation* element. It shall include the status of digital signature validation and an array of validation errors, if detected.

The Augment response JSON object shall be defined as in JSON Schema file (signa-arch-api-schema.json) provided in section 4.9.5.1. The elements of JSON Schema shall implement elements of Augment response mapped by names as show in the following table:

Element	Description	JSON member name
AugmentationStatus	The status of augmentation operation execution	augmentStatus
DocumentSpecificationId	The identifier of specification defining the content of SubDO	docSpecId
AugmentAt	The date/time of the next preservation action execution	augmentAt
DocumentAvailableUntil	The time until augmented preservation object is available for retrieval by preservation client	docAvailableUntil
DocumentTicket	The unique code to be used by preservation client to retrieve augmented preservation object	docTicket
Signatures	Array of "signature" objects providing information on each digital signature and its augmentation status	signatures
Signature element		
AugmentationValidation	The status of electronic signature or electronic seal validation performed during augmentation	augmentValidation
AugmentNotBefore	The start date/time of the augmentation period for the next preservation action execution	augmentNotBefore

AugmentAt	The date/time of the next preservation action execution for this electronic signature or electronic seal	augmentAt
AugmentNotAfter	The end date/time of the augmentation period for the next preservation action execution	augmentNotAfter
Info	The information on electronic signature or electronic seal	info
Validation	The general status of electronic signature or electronic seal validation and array of error messages, if detected	validation

4.9.3. Download

The *Download* operation will retrieve preservation object previously augmented with *Augment* operation. This operation will be called immediately after receiving *Augment* operation response with returned *DocumentTicket* value, this way allowing preservation client to retrieve preservation object content. The client will execute the call to *Download* operation before the time indicated in *DocumentAvailableUntil* element of *Augment* response, after which the preservation object will no longer be available.

4.9.3.1 Download Request

The *Download* request shall include following fields:

- The *document ticket*. It shall contain identifier assigned to this preservation object by the preservation service. Its value shall be the same, as the one provided in previously executed *Augment* response (*DocumentTicket*).

The request may contain the following fields:

- The *client ID*. If present, it shall contain the identifier of the preservation client which performs the preservation object retrieval call. If present, this value shall be the same as in previously executed corresponding *Augment* request.

The fields above shall be implemented as the following HTTP GET request parameters:

Fields	Description	HTTP request parameters
Client ID	Identifier of the preservation client	cid
Document ticket	Preservation object identifier to be retrieved	docTicket

4.9.3.2 Download Response

The Download operation response is returned as HTTP response with status code 200 and the content of preservation object. The content of preservation object is returned without encoding or transformation.

In case of a failure, the HTTP response with status code different from 200 and content with error messages are returned.

4.9.4. PreservePO

PreservePO operation is used to perform preservation of the provided preservation object, which is some new submission data object, or preservation object container previously obtained by this operation. The provided preservation object is an electronic document containing electronic signatures or electronic seals. For this operation preservation object shall be provided. Preservation is performed by augmenting electronic signatures and seals. It involves creation/collection of the validation data (in the case it is missing) and preservation evidences and their inclusion into the preservation objects – electronic signatures or electronic seals.

Electronic signatures or seals with B-B, B-L, B-LT and B-LTA levels are augmented and expected evidence duration and recommended next augmentation time are determined and returned following the same principles as in operation Augment (section 4.9.2).

This operation supports submission data objects of the formats listed in the Section 4.12. SubDOs and preservation objects derived from them are not stored in PSP storage, but information of the performed actions (traces) are stored. In the traces, SubDO and preservation object will be identified using preservation object identifier (id) provided by this operation.

PreservePO operation syntax and semantics are described in the ETSI TS 119 512 section 5.3.3. Current profile supports this operation only used with JSON syntax.

PreservePO operation shall accept the following elements from *OptionalInputs* element, as defined in section 4.2.8 of OASIS DSS-X Core 2.0 [DSS Core 2.0]:

- The optional *lang* element. If provided, it shall contain the code of the language to be used for preservation service messages.
- The *other* element. It shall contain an array with single element with attributes:
  - The *ID* attribute. It shall have fixed value "cid".
  - The *value* attribute. It shall have Base64 encoded preservation client identifier value.

PreserverPO operation shall be available to preservation clients that can be identified by the preservation service. To allow for the preservation client identification, *PreserverPO* operation request shall include *client identifier* with the help of element *other*, as described above.

PreservePO operation shall include in its result the following elements from *OptionalOutput* element, as defined in section 4.2.9 of OASIS DSS-X Core 2.0 [DSS Core 2.0]:

- The *other* element with an array of elements. For each PO, this array might contain two elements:
  - Element indicating *expected evidence duration* time. This time is calculated taking the minimal expected evidence duration time among all digital signatures to be preserved within provided PO.

    The *ID* attribute of this element shall have fixed value "eed". The attribute *value* shall have Base64 encoded time value in ISO Date Time Format. The attribute *idRef* shall have PO identifier value (id).
  - Element indicating *next augmentation* time. It shall contain time value indicating when the next augmentation of corresponding PO needs to be executed. It should be used as a recommended time for the next preservation action to be called by the client. This time is calculated

taking the minimal next augmentation time among all digital signatures to be preserved within PO.

The *ID* attribute of this element shall have fixed value "aat". The attribute *value* shall have Base64 encoded time value in ISO Date Time Format. The attribute *idRef* shall have PO identifier value (id).

If no *expected evidence duration* or *next augmentation* time can be determined for the PO, then corresponding elements shall be omitted in the *other* element.

4.9.5. JSON Schemas and OpenAPI Documents

4.9.5.1 JSON Schema files

The JSON schema definitions for the *Augment* and *Download* operations are presented at:

- https://qtsp.mitsoft.lt/repo/qlps/arch-protocol/v1/signa-arch-api-schema.json

The JSON schema definitions for the *RetrieveInfo* and *PreservePO* operations are presented at:

- https://qtsp.mitsoft.lt/repo/qlps/preserv-protocol/v1/signa-19512-preservation-api-schema.json

4.9.5.2 OpenAPI specifications

The OpenAPI specification for the *Augment* and *Download* operations are presented at:

- https://qtsp.mitsoft.lt/repo/qlps/arch-protocol/v1/signa-arch-api-openapi.json

The OpenAPI specification for the *RetrieveInfo* and *PreservePO* operations is presented at:

- https://qtsp.mitsoft.lt/repo/qlps/preserv-protocol/v1/signa-19512-preservation-api-openapi.json

4.10. Preservation protocol usage guidelines

Sections 4.8 and 4.9 has detailed description of the process of augmentation of electronic signatures and electronic seals present in submission data object. They also explain how the expected evidence duration is determined, and how augmentation period and caution period are used when choosing the next augmentation time for preservation object. Preservation operations should be used taking into account this information.

Subscriber shall use the preservation operations according to the following guidelines:

- MitSoft PSP highly recommends to plan next call to preservation service at the next augmentation time indicated in the result (*AugmentAt*, *aat*) at whole preservation object level of preservation operation (*augment*, *PreservePO*).
- If preservation client ignores the recommended next augmentation time, then it shall plan the next call of preservation operation in the following way:
  - Plan next call of preservation operation in a time frame between the start of augmentation period and start of caution period.
  - Determine the exact time of start of augmentation period and start of caution period for a submission data object using *expected evidence duration* time returned by previous call of preservation operation and augmentation period and caution period indicated in subscriber agreement.

  - Avoid calling preservation operation before the start of augmentation period. Such call shall be ignored by the preservation service, unless it is necessary because of changed expected evidence duration time (see below).
  - Avoid calling preservation operation during caution period. While such call can still be successful, it poses the risk to miss object preservation in case of failures of preservation service or other related services.
- If call of preservation operation fails due to the failure of preservation service (service unavailable, software errors, unavailable related services), then the preservation client shall repeat the preservation operation calls until the preservation service becomes available.
- The preservation client shall implement the possibility to call preservation operations prior to planned next augmentation time. For example, upon reception of information that preservation of data objects must be executed earlier than planned due to (unexpectedly) changed reliability period of cryptographic algorithms.

4.11. Applicable policies

The MitSoftQWOS profile supports the following preservation evidence policy:

- MitSoft Qualified Long-term Preservation Service Preservation Evidence Policy, which is indicated by the unique object identifier (OID):
  - 1.3.6.1.4.1.57890.1.7.1.X

  where x stands for the latest version. All versions are available for the subscribers on the repository of MitSoft PSP. Each version contains the point in time from which on this version of the policy has become or will become active. The validity period of the version ends when new version becomes active.

The MitSoftQWOS profile supports the following signature validation policy:

- MitSoft Qualified Long-term Preservation Service Signature Policy, which is indicated by the unique object identifier (OID):
  - 1.3.6.1.4.1.57890.1.6.1.X

  where x stands for the latest version. All versions are available for the subscribers on the repository of MitSoft PSP. Each version contains the point in time from which on this version of the policy has become or will become active. The validity period of the version ends when new version becomes active.

4.12. Supported submission data objects

Preservation service accepts electronic documents or containers that are signed with electronic signatures or sealed with electronic seals. Every submitted electronic document or container shall contain actual data signed by the electronic signature or sealed by the electronic seal.

Every submission data object shall contain at least one electronic signature or electronic seal.

Submission data object formats supported by this preservation profile are defined in the following table:

Submission data object format (electronic document specification or container standard)	Electronic signature/seal format	Submission data object (specification) identifier
ADOC-V1.0 electronic document	XAdES	ADOC-V1.0
ADOC-V2.0 electronic document	XAdES baseline	ADOC-V2.0
EGAS-V1.0 electronic document	XAdES	EGAS-V1.0
MDOC-V1.0 electronic document	XAdES	MDOC-V1.0
PDF-LT-V1.0 electronic document	PAdES baseline	PDF-LT-V1.0
PDF-RC-V1.0 electronic document	PAdES baseline	PDF-RC-V1.0
ASiC-E container according to ETSI TS 103 174	XAdES baseline	ASiC-E-XAdES-TS
ASiC-E container according to ETSI EN 319 162-1	XAdES baseline	ASiC-E-XAdES-EN
ASiC-S container according to ETSI TS 103 174	XAdES baseline	ASiC-S-XAdES-TS
	CAdES baseline	ASiC-S-CAdES-TS
ASiC-S container according to ETSI EN 319 162-1	XAdES baseline	ASiC-S-XAdES-EN
	CAdES baseline	ASiC-S-CAdES-EN
PDF document with PAdES signatures according to ETSI TS 103 172	PAdES baseline	PDF-PAdES-TS
PDF document with PAdES signatures according to ETSI EN 319 142-1	PAdES baseline	PDF-PAdES-EN
PDF document with CMS signatures	PAdES	PDF-PAdES-CMS

This set of supported submission data object formats for particular subscriber may be reduced by the Subscriber agreement between this preservation service subscriber and MitSoft PSP.

4.12.1. ADOC-V1.0 electronic documents

ADOC-V1.0 electronic document [ADOC-V1.0], which contains at least one XAdES electronic signature or electronic seal conformant to XAdES standard ETSI TS 101 903 [TS 101 903].

This submission data object is identified by the following identifier:

- `ADOC-V1.0`

This submission data object is identified by the following URI identifier:

- `http://uri.mitsoft.lt/specification/type/adoc-v1.0`

The MIME type of this submission data object file is as follows:

- `application/vnd.lt.archyvai.adoc-2008`

4.12.2. ADOC-V2.0 electronic documents

ADOC-V2.0 electronic document [ADOC-V2.0], which contains at least one XAdES electronic signature or electronic seal conformant to XAdES baseline profile standard ETSI TS 103 171 [TS 103 171].

This submission data object is identified by the following identifier:

- `ADOC-V2.0`

This submission data object is identified by the following URI identifier:

- `http://uri.mitsoft.lt/specification/type/adoc-v2.0`

The MIME type of this submission data object file is as follows:

- `application/vnd.etsi.asic-e+zip`

4.12.3. EGAS-V1.0 electronic documents

EGAS-V1.0 electronic document [EGAS-V1.0], which contains at least one XAdES electronic signature or electronic seal conformant to XAdES standard ETSI TS 101 903 [TS 101 903].

This submission data object is identified by the following identifier:

- `EGAS-V1.0`

This submission data object is identified by the following URI identifier:

- `http://uri.mitsoft.lt/specification/type/egas-v1.0`

The MIME type of this submission data object file is as follows:

- `application/vnd.lt.sodra.egas-2009`

4.12.4. MDOC-V1.0 electronic documents

MDOC-V1.0 electronic document [MDOC-V1.0], which contains at least one XAdES electronic signature or electronic seal conformant to XAdES standard ETSI TS 101 903 [TS 101 903].

This submission data object is identified by the following identifier:

- `MDOC-V1.0`

This submission data object is identified by the following URI identifier:

- `http:/uri.mitsoft.lt/specification/type/mdoc-v1.0`

The MIME type of this submission data object file is as follows:

- `application/vnd.lt.archyvai.mdoc-2010`

4.12.5. PDF-LT-V1.0 electronic documents

PDF-LT-V1.0 electronic document [PDF-LT-V1.0], which contains at least one PAdES electronic signature or electronic seal conformant to PAdES baseline profile standard ETSI TS 103 172 [TS 103 172].

This submission data object is identified by the following identifier:

- `PDF-LT-V1.0`

This submission data object is identified by the following URI identifier:

- `http://uri.mitsoft.lt/specification/type/pdf-lt-v1.0`

The MIME type of this submission data object file is as follows:

- `application/pdf`

4.12.6. PDF-RC-V1.0 electronic documents

PDF-RC-V1.0 electronic document [PDF-RC-V1.0], which contains at least one PAdES electronic signature or electronic seal conformant to PAdES baseline profile standard ETSI EN 319 142-1 [EN 319 142-1].

This submission data object is identified by the following identifier:

- `PDF-RC-V1.0`

This submission data object is identified by the following URI identifier:

- `http://uri.mitsoft.lt/specification/type/pdf-rc-v1.0`

The MIME type of this submission data object file is as follows:

- `application/pdf`

4.12.7. ASiC-E container according to ETSI TS 103 174

ASiC-E container which is conformant to ASiC baseline profile standard [TS 103 174] and which contains at least one XAdES electronic signature or electronic seal conformant to XAdES baseline profile standard ETSI TS 103 171 [TS 103 171].

This submission data object is identified by the following identifier:

- `ASiC-E-XAdES-TS`

This submission data object is identified by the following URI identifier:

- `http://uri.mitsoft.lt/ades/type/asic-e/xades/ts`

The MIME type of this submission data object file is as follows:

- `application/vnd.etsi.asic-e+zip`

4.12.8. ASiC-E container according to ETSI EN 319 162-1

ASiC-E container which is conformant to Building blocks and ASiC baseline container standard ETSI EN 319 162-1 [EN 319 162-1] and which contains at least one XAdES electronic signature or electronic seal conformant to Building blocks and XAdES baseline signatures standard ETSI EN 319132-1 [EN 319 132-1].

This submission data object is identified by the following shrt identifier:

- `ASiC-E-XAdES-EN`

This submission data object is identified by the following URI identifier:

- `http://uri.mitsoft.lt/ades/type/asic-e/xades/en`

The MIME type of this submission data object file is as follows:

- `application/vnd.etsi.asic-e+zip`

4.12.9. ASiC-S container with XAdES signatures according to ETSI TS 103 174

ASiC-S container which is conformant to ASiC baseline profile standard ETSI TS 103 174 [TS 103 174] and which contains electronic signature or electronic seal conformant to XAdES baseline profile according ETSI TS 103 171 [TS 103 171].

This submission data object is identified by the following identifier:

- `ASiC-S-XAdES-TS`

This submission data object is identified by the following URI identifier:

- `http://uri.mitsoft.lt/ades/type/asic-s/xades/ts`

The MIME type of this submission data object file is as follows:

- `application/vnd.etsi.asic-s+zip`

4.12.10. ASiC-S container with CAdES signatures according to ETSI TS 103 174

ASiC-S container which is conformant to ASiC baseline profile standard ETSI TS 103 174 [TS 103 174] and which contains electronic signature or electronic seal conformant to CAdES baseline profile according ETSI TS 103 173 [TS 103 173].

This submission data object is identified by the following identifier:

- `ASiC-S-CAdES-TS`

This submission data object is identified by the following URI identifier:

- `http://uri.mitsoft.lt/ades/type/asic-s/cades/ts`

The MIME type of this submission data object file is as follows:

- `application/vnd.etsi.asic-s+zip`

4.12.11. ASiC-S container with XAdES signatures according to ETSI EN 319 162-1

ASiC-S container which is conformant to Building blocks and ASiC baseline container standard ETSI EN 319 162-1 [EN 319 162-1] and which contains electronic signature or electronic seal conformant to Building blocks and XAdES baseline signatures standard ETSI EN 319 132-1 [EN 319 132-1].

This submission data object is identified by the following identifier:

- `ASiC-S-XAdES-EN`

This submission data object is identified by the following URI identifier:

- `http://uri.mitsoft.lt/ades/type/asic-s/xades/en`

The MIME type of this submission data object file is as follows:

- `application/vnd.etsi.asic-s+zip`

4.12.12. ASiC-S container with CAdES signatures according to ETSI EN 319 162-1

ASiC-S container which is conformant to Building blocks and ASiC baseline container standard ETSI EN 319 162-1 [EN 319 162-1] and which contains electronic signature or electronic seal conformant to Building blocks and CAdES baseline signatures standard ETSI EN 319 122-1 [EN 319 122-1].

This submission data object is identified by the following identifier:

- `ASiC-S-CAdES-EN`

This submission data object is identified by the following URI identifier:

- `http://uri.mitsoft.lt/ades/type/asic-s/cades/en`

The MIME type of this submission data object file is as follows:

- `application/vnd.etsi.asic-s+zip`

4.12.13. PDF documents with PAdES signatures according to ETSI TS 103 172

PDF electronic document with PAdES electronic signature or electronic seal conformant to PAdES baseline profile standard ETSI TS 103 172 [TS 103 172].

This submission data object is identified by the following identifier:

- `PDF-PAdES-TS`

This submission data object is identified by the following URI identifier:

- `http://uri.mitsoft.lt/ades/type/pdf/pades/ts`

The MIME type of this submission data object file is as follows:

- `application/pdf`

4.12.14. PDF document with PAdES signatures according to ETSI EN 319 142-1

PDF electronic document with PAdES electronic signature or electronic seal conformant to Building blocks and PAdES baseline signatures standard ETSI EN 319 142-1 [EN 319 142-1].

This submission data object is identified by the following identifier:

- `PDF-PAdES-EN`

This submission data object is identified by the following URI identifier:

- `http://uri.mitsoft.lt/ades/type/pdf/pades/en`

The MIME type of this submission data object file is as follows:

- `application/pdf`

4.12.15. PDF documents with CMS signatures

PDF electronic document with PAdES electronic signature or electronic seal conformant to Profile for CMS Signatures in PDF defined in the standard ETSI TS 102 778-2 "PDF Advanced Electronic Signature Profiles; Part 2: PAdES Basic - Profile based on ISO 32000-1" [TS 102 778-2]. Long-term form for this profile is defined in the standard ETSI TS 102 778-4 "PDF Advanced Electronic Signature Profiles; Part 4: PAdES Long Term - PAdES-LTV Profile" [TS 102 778-4].

This submission data object is identified by the following identifier:

- `PDF-PAdES-CMS`

This submission data object is identified by the following URI identifier:

- `http://uri.mitsoft.lt/ades/type/pdf/pades/cms`

The MIME type of this submission data object file is as follows:

- `application/pdf`

4.13. Supported preservation evidence formats

The following preservation evidence formats are supported by this preservation profile: XAdES Archive Time Stamp, CAdES Archive Time Stamp V3, PAdES Document Time Stamp.

4.13.1. XAdES Archive Time Stamp

The XML-based Archive Time Stamp property according ETSI TS 101 903 [TS 101 903] and ETSI EN 319 162-1 [EN 319 162-1] is used for preserving XAdES digital signatures that are part of ADOC-V1.0, ADOC-V2.0, EGAS-V1.0, MDOC-V1.0 electronic documents or ASiC-E, ASiC-S containers with XAdES digital signatures.

4.13.2. CAdES Archive Time Stamp V3

The ASN.1-based Archive Time Stamp V3 attribute according ETSI TS 101 733 [TS 101 733] and ETSI EN 319 122-1 [EN 319 122-1] is used for preserving CAdES digital signatures that are part of ASiC-S containers with CAdES digital signatures.

4.13.3. PAdES Document Time Stamp

The Document Time-Stamp attribute according ETSI TS 102 778-4 [TS 102 778-4] and ETSI EN 319 142-1 [EN 319 142-1] is used for preserving PAdES digital signatures

that are part of PDF-LT-V1.0, PDF-RC-V1.0 electronic documents or PDF documents with PAdES digital signatures.