EPC API Security Framework

European Payments Council

European Payments Council AISBL,
Cours Saint-Michel 30 B-1040 Brussels
T +32 2 733 35 33
Enterprise N°0873.268.927
secretariat@epc-cep.eu

EPC164-22
Version 1.0
Date issued: 15 March 2023

Public

Approved

API Security Framework

Abstract	The present document provides the security requirements related to the use of APIs as part of an EPC scheme.
Document Reference	EPC164-22
Issue	Version 1.0
Date of Issue	15 March 2023
Reason for Issue	The approved Version
Reviewed by	EPC
Produced by	EPC
Circulation	Publicly available

Table of Contents

Note:

The capitalized keywords "MUST", “MAY", "SHOULD" and their variants, should be interpreted as defined in RFC 2119.

1. Background

The use of APIs for the exchanges between scheme participants will be either mandatory or a feasible option, depending on the EPC schemes.

Since API related security matters are essential and support the actual API exchange, the purpose of this document is to define an API security framework based on widely available European or international security standards, listing the minimum-security related requirements applicable, regardless of the scheme, to the SRTP and SPAA scheme participants using APIs.

These requirements are independent of the API functionalities and technical implementations and are applicable to all the SRTP and SPAA schemes and to all the API specifications (market or EPC) used by a scheme participant, whether the scheme participant chooses to send and receive messages directly, or whether it chooses to use mutualised services of a "hub" (technical solution provider) acting as a message gateway.

SPAA and SRTP schemes are both managed by the EPC and were designed to use APIs for the communication between scheme participants. Although there are some differences relative to how both schemes operate, as well as a difference in maturity between both schemes, they are sufficiently similar as messaging schemes to justify a joint effort in defining a common API security framework.

Wherever there is a difference in each scheme that justifies a different approach in the security framework, that difference will be highlighted.

2. Scope

The purpose of this document is to describe the requirements of an API Security Framework that can be shared initially by SPAA and SRTP and in the future by other EPC initiatives that requires the use of an API, including:

- the security-related requirements based on widely available European or international security standards. The recommended security measures shall be proportionate and affordable.
- the list of operational requirements that an Operational Scheme Manager (OSM) should provide to ensure a smooth functioning of the framework.

The requirements laid out in this framework only relate to the interaction between the scheme participants. The interaction between the scheme participants and their customers is outside the scope of the framework.

The specifications of the requirements laid out in this document will be the responsibility of each API specification that uses them. In the case of SPAA that would be any API initiative defining a technical specification of the SPAA Scheme Rulebook. In the case of the SRTP scheme it would be the responsibility of the SRTP scheme to define the content for its own API specifications and the scope of the homologation process related to the below requirements. It is also possible that different API initiatives define the content of other API specifications for SRTP. In any case, none of those specifications will be a part of this document.

The integration of each specification of the requirements laid out in this framework will be a responsibility of the scheme for which those specifications are intended. Furthermore, the obligation to implement any given specification by a scheme participant, will also be the responsibility of each scheme.

Although most of the requirements are common and applicable to all schemes, there could be some specificities that will be indicated accordingly.

The requirements in this document are about securing the API itself as an "envelope", not about securing each piece of information inside each individual message exchanged through the API. Messages may contain information that could lead to fraud if insufficiently secured, such as a Payee's IBAN. If such information, inside messages, is to be protected, it is the role of Schemes to describe how. The current framework will only ensure that the messages flow correctly between identified, authenticated and authorised participants, ensuring confidentiality, and that the content of the message was not tampered with in transport. The framework does not look at the content of the messages.

3. Actors and Roles

3.1. Operational Scheme Manager

The Operational Scheme Manager will collect, validate, maintain and when applicable, make available additional data related to the scheme participants to ensure an effective functioning of the schemes.
The scheme participants' requirements related to this role are described in chapter 14.
The requirements applicable to the OSM are specified in a separate document.

3.2. SPAA scheme – Actors and Roles

The following two actors are the Scheme Participants:

- **Asset Holder** (i.e. ASPSPs in a PSD2 context, which can be Credit Institutions, Electronic Money Institutions or Payment Institutions)
    - Role: The entity that holds the asset(s) for the Asset Owner. The assets can be transaction assets and information assets.
    - An Asset Holder needs to be a license ASPSP

- **Asset Broker** (e.g. TPPs or other PSPs with the appropriate license extension in a PSD2 context)
    - Role: The entity that uses the asset(s) (e.g. transactions or data) from the Asset Holder, with permission of the Asset Owner, to deliver value to the Asset User.
    - It is envisaged that some assets offered by AH through SPAA might not require being a supervised PSP to act as AB e.g. branch information, catalogues etc

Two other players are involved in the processes but are not Scheme participants:

- **Asset Owner** (client of the Asset Holder and optionally of the Asset Broker):
    - Role: The client that owns the asset(s) (e.g. a legal entity or a consumer, in which case it would also be a data subject).
- **Asset User** (client of the Asset Broker only):
    - Role: The client of the Asset Broker that uses the asset(s). For transactional assets, this is typically the Payee/merchant. For data assets, it is typically the same as the Asset Owner, or in the case of legal entities an individual with the adequate Power of Attorney.

  Note that both the Asset User and the Asset Owner can have the role of a payer or a payee depending on the context.

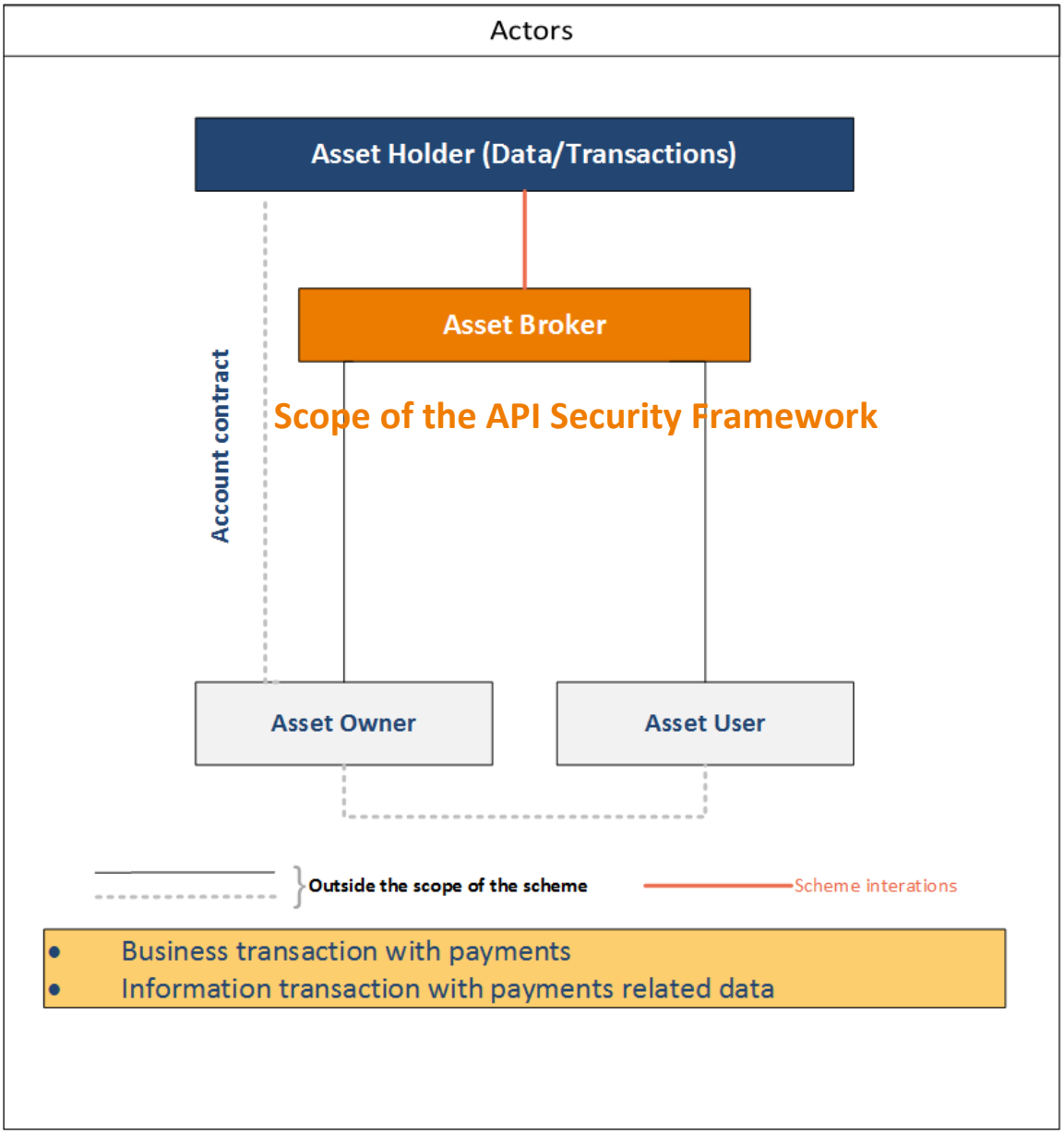

Figure 1 - SPAA actors

3.3. SRTP Scheme – Actors and Roles

The four roles involved in the Scheme include:

- **Payee**: The initiator of an RTP process and usually the beneficiary of the funds transferred if the resulting payment flow occurs. Depending on the business domain we are referring to, this role can be identified as the beneficiary when it comes up to the payment processing or the creditor from a financial perspective.
- **Payer**: It represents the party to whom the RTP is addressed, and usually the originator of the funds transferred if the resulting payment flow occurs. In payment processing this role is usually identified with the originator of a payment, which can be also defined as the debtor from a financial perspective. A Payer should always have the possibility to opt out from the RTP service.

- **Payee's RTP Service Provider** (who has adhered to the Scheme): Usually represented by a PSP but since the RTP can be part of end-to-end commerce processes, also other non-PSP entities can assume this role. Therefore, the Payee's RTP Service Providers can be for instance:
    - PSPs[1]
    - E-invoicing Service Providers
    - Commerce Service Providers
- **Payer's RTP Service Provider** (who has adhered to the Scheme): Usually represented by a PSP but also other non-PSP entities can assume this role. Therefore, the Payer's RTP Service Providers can be for instance:
    - PSPs[1]
    - E-invoicing Service Providers
    - Commerce Service Providers

It is up to this SRTP service provider (business decision) to decide if it wants to play one or the other role or even both, for its customers.

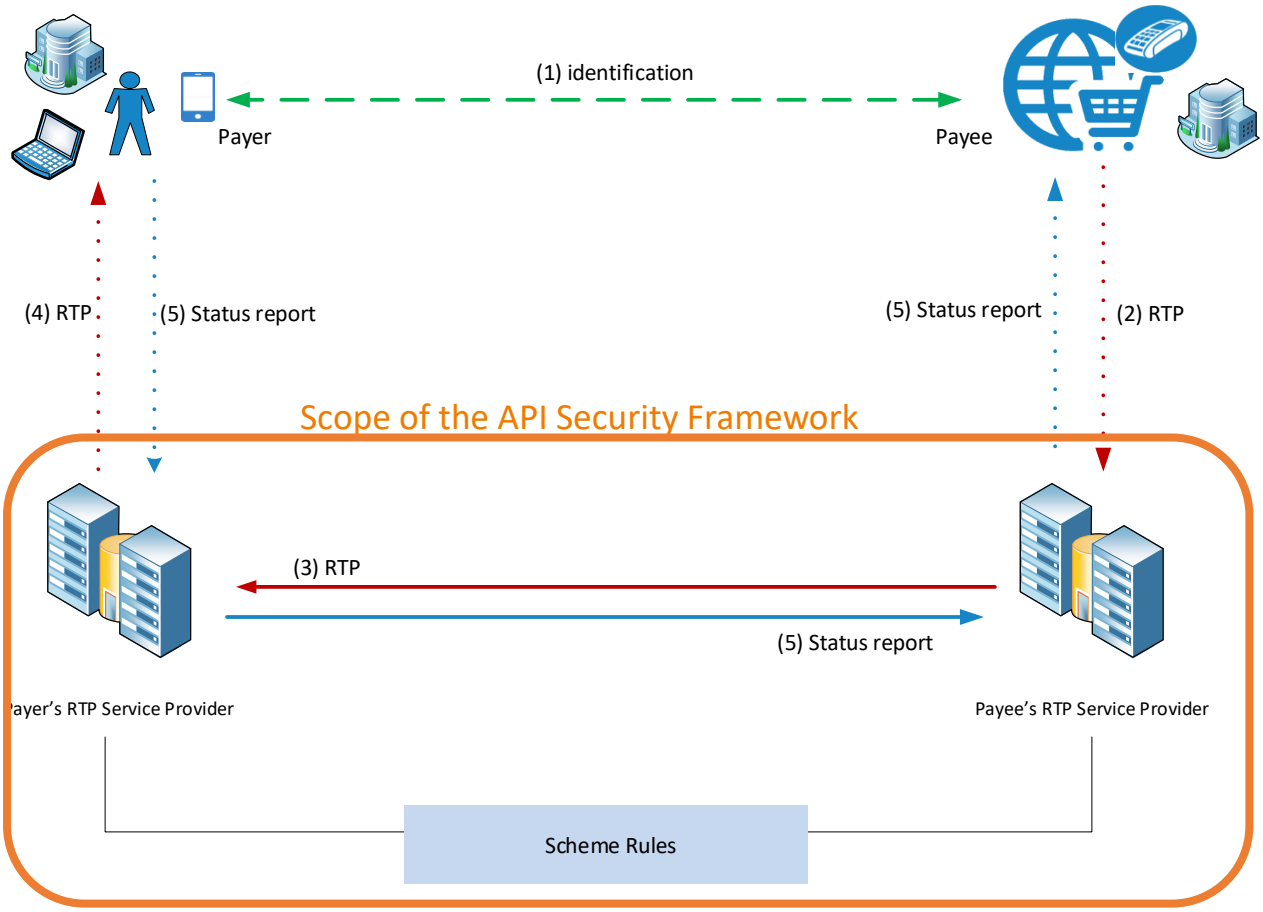

Figure 2 - SRTP roles and information flow in 4-corner eco-system

[1] Even though multiple types of providers can process RTPs, only PSPs can execute functions related to payment, such as initiation or execution of payment instructions through inter-PSP networks.

4. Defined terms and abbreviations

Term/Abbreviation	Definition
AISP	Account Information Service Provider
API	Application Programming Interface
API client	The party that sends the API request for a specific service or data to an API server
API server	The party that accepts the API request, processes it and sends the response information
ASPSP	Account Servicing Payment Service Provider
OSM	Operational Scheme Manager
Participant	An entity accepted to be a part of the Scheme in accordance with this scheme's rulebook.
PISP	Payment Initiation Service Provider
PSD2	Payment Services Directive
PSP	Payment Service Provider
RTP	Request-to-Pay
RTSP	Referenced Technical Solution Provider SRTP homologated "hub" or "proxy" acting as a message gateway which remains totally "transparent" from a scheme perspective.
SPAA	SEPA Payment Account Access
SRTP	SEPA Request to Pay
TLS	Transport Layer Security
TPP	Third Party Provider
URI	Uniform Resource Identifier
URL	Uniform Resource Locator

5. Identification

5.1. API Server/API Client

For usual services, the API server is the Asset Holder or the Payer's SRTP Service Provider. For call-back services, the API server is the Asset Broker or the Payee's SRTP Service Provider.

The SRTP/SPAA related API specifications cover the SRTP/SPAA related messages exchanged between the Payee's SRTP Service Provider/ Asset Broker and the Payer's SRTP Service Provider/Asset Holder, in both directions. The two technical roles being API client or API server depending on the situation.
Since some interactions may not be completed synchronously, a call back possibility has been set to send back asynchronous responses.

5.2. API Client Customer (dedicated to SPAA scheme)

This will be covered in the SPAA specificities annex (Annex 1).

5.3. Scheme Participants Identification requirements

Purpose: unambiguously identify a scheme participant from a "machine readable" perspective.

Requirement: each participant MUST have at least one identifier.

Identifiers are given under the responsibility of the Operational Scheme Manager (OSM), which must ensure their unicity per identifier type (the same identifier cannot be given twice) and unambiguousness (it identifies without ambiguity a single scheme participant).

The OSM may let scheme participants use identifiers they already have, provided they are compliant with the unicity and unambiguousness requirement (e.g., PSD2 identifier, LEI, VAT number etc…). This allows potential re-use of existing infrastructure for those scheme participants.

When a scheme participant does not have such existing identifier, the OSM must provide one.

Identifiers of scheme participants are the key to fetch other information about the scheme participants, and for authentication and authorisation.

Certificates to be used for identification of the API client MUST be Qualified Web Authentication certificates (QWAC) with a profile based on the European Standard ETSI EN 319 412-4. The extensions defined by ETSI TS 119 495 may also be used. A QWAC based on ETSI TS 119 495 already owned by a participant and used to access an API according to the requirements of the Directive (EU) 2015/2366 (PSD2) may be reused for accessing APIs within the SPAA scheme or the SRTP scheme. The list of Trust Services Providers able to provide eIDAS certificates is available by the European Commission (EU Trust service dashboard).

By construction of those qualified certificates, when a scheme participant wants to use several different identifiers, it MUST have a certificate for each of its identifier.

Certificates to be used for identification of the API server:
The API server does not need a QWAC, it can use a standard website certificate.

When a certificate is no longer valid, the scheme participants should initiate the revocation procedure and inform the OSM accordingly.

The scheme participants should be able to receive and act upon the broadcasted emergency messages issued by the OSM.

Technical remark:
In the context of SPAA and according to the current legislation, Asset Holders are not required to use a QWAC; EV certificates are sufficient.
It is possible that a participant in a API server role uses the same certificate as API client for call-

backs. In this case the field extendedKeyUsage SHALL contain both attributes clientAuth and serverAuth.

6. Scheme Participants Information requirement

Purpose: additional information that is required from a scheme participant to establish a secure connection.

- Scheme Participant Roles

Role's definition depends on the scheme. When a unique type of role (e.g., SRTP Participant) exists, this section is void and does not specify any requirement because being identified and authenticated means that the participant has the single role defined in the scheme. When several roles are defined in the scheme, being authorised to a type of role can limit the set of APIs available at an endpoint.

When a scheme participant intends to serve Payers, then:

- Finding the API's endpoint (the URL) of an identified scheme participant is **necessary** to be able to call the said API.

- Finding the API documentation (at least URL) is **necessary**. In the case this is the default API proposed by the scheme, this indication is enough. When this is an API proposed by an API Initiative, the link to the specification is enough.

The commercial name ("human readable") of a scheme participant might be **useful** if that name is to be presented to the Payers or the Payees.

7. Scheme Participants Authentication requirements

Purpose: proof the identity for a scheme participant.

When calling another scheme participant's API's endpoint, the participants MUST perform mutual authentication. This authentication serves to prove the identity they are claiming to be: see chapter 5.

The mutual authentication is done by applying TLS (including client authentication).

The TLS version, key length and algorithms MUST comply with the standard recommended by the EPC - PSSG[2] and as mentioned in the respective Risk Management Annex of each scheme for the securing of data transport.

8. Secured communication between Scheme Participants requirements

Purpose: avoid eavesdropping and tampering of communication.

The TLS layer discussed at the Authentication chapter (chapter 7) covers the requirements.

[2] Link to the guidelines on cryptographic algorithms usage and key management: EPC Document (europeanpaymentscouncil.eu)

9. Authorisation requirements

9.1. Authorisation principles

Authorisation is the process by which an API server will decide to allow or deny an API client request. In case of denial, the API server will answer with an HTTP 403 (Forbidden) response.

The URI requested by an API client may be subject to different access modalities for scheme participants:

1. Closed access with authentication
    - A prerequisite registration of the API client with the scheme is mandatory and subject to some conditions
    - The API client must authenticate when accessing the URI
2. Closed access with authentication and explicit authorisation
    - A prerequisite registration of the API client with the scheme is mandatory and subject to some conditions
    - The API client must also get the consent from the data owner. The consent could for example result in the provision of an authorisation token to the API client, for instance using OAuth2 (RFC 6749) (not currently applicable to the SRTP scheme).
    - The API client must authenticate when accessing the URI.

The registration conditions of an API client may include:

- A contractual relationship between both legal entities, on API client ad API server sides) (not currently applicable to the SRTP scheme).
- A legal role (e.g.: PISP, AISP roles as specified by PSD2) (not currently applicable to the SRTP scheme).
- A participation to a scheme (e.g.: SRTP).
- …

9.2. Schemes closed access with authentication

This use-case will apply:

- to all the SRTP-API interactions
- to some of the SPAA transactional assets

In both cases, the registration will be subject to the participation of the API client to the relevant EPC scheme.

In the case of the SPAA Scheme, this registration must be completed by the PISP role of the API client.

9.2.1. Basic approach

To allow or deny the access to the API client, the API server must perform the following steps:

1. Authenticate the API client (cf. chapter 7)
2. Extract the subsequent identity of the API client (cf. chapter 5)
3. Check the participation of the API client to the relevant scheme

4. Check the legal role of the API client if needed
5. Check the access right of the API client to the API resources that are accessed through the URI

The last step is critical to avoid an unauthorised access, for instance to a resource that was submitted by another API client:

- A SEPA Request-To-Pay or a Cancellation Request in case of SRTP
- A Payment Initiation Request in case of SPAA

As the whole authorisation process may be repeatedly executed by the API server for each API client request, it is also possible to optimise this process through a pre-enrolment of the API client.

9.2.2 Optional optimised approach

9.2.2.1 Pre-enrolment

This pre-enrolment will allow the API server to get and store some characteristics of a given API client:

- Names, logos
- Certificates or public keys
- Call-back and redirect URIs
- Requested grants and scopes
- …

These characteristics may be complemented by information extracted and regularly refreshed from external repositories

- Scheme participation and roles
- Legal roles
- …

The pre-enrolment process will provide the API client with a client id that is linked to the usage context specified by the provided information. If needed, an API client may execute, with the same API server, several pre-enrolments for different specialised usage contexts.

An example for an enrolment protocol is proposed by the IETF in the OAuth2 context (RFC 7591 & 7592).

9.2.2.2 Client id usage

The client id may be used to get an authorisation token, for instance using OAuth2 (RFC 6749).

The API server can execute some of the authorisation steps at once before providing the authorisation token to the API client.

10.Scheme Participants sealing requirements

Purpose: providing the proof that an API client has indeed submitted a given request and vice versa an API server has indeed provided a given response.

Note: This section is only relevant when the scheme has required that some messages or part of messages must be protected by sealing.

The corresponding message (or part of it) must then be signed.

The identity of the signee is represented by certificates for sealing (QSealC). Those certificates SHALL be based on the profile defined by the European Standard ETSI EN 319 412-3. The owner is always a legal person acting as either the API client or the API server. If needed the extensions defined by ETSI TS 119 495 may also be used for these certificates.

Those certificates can be provisioned through the same list of Certification Authorities already mentioned above.

Key length and algorithms MUST comply with the EPC – PSSG recommendation for sealing.

The exact protocol to transport the seal along with the message not being an undisputed European or international standard, it is up to the API used to specify that protocol in the API documentation.

11.Availability requirements

Availability of the infrastructures, liabilities, emergency plans and minimal standardisation of the solutions.

The system MUST ensure high availability of services in accordance with the adopted classification of criticality, in particular through redundancy of components, backing up data and software as well as automatic maintenance of system continuity.

12.Security conformance and testing

The Scheme may require scheme participants to prove that security requirements among other requirements have been met during the homologation process and to confirm that it is maintained on a regular basis to the OSM. In this case Scheme Participants should provide a testing environment, test data or any other evidence requested as part of the homologation process.

13.Audit trail requirements

These requirements are applicable to all scheme participants, both on the server and client sides.

As a basic audit requirement, the audit log must be always enabled, and must include all the information that is legally required.

In additional, it is strongly recommended that entries (logEntry) include the following objects/fields:

- logName: The resource name of the log
- timestamp: The time the event described by the log entry occurred
- serviceName: the name of the service used/invoked
- uniqueID: A unique identifier for the log entry
- httpRequest: Information about the HTTP request associated with this log entry.
    - URL
    - Headers
    - Payload if any

- httpResponse: Information about the HTTP response associated with this log entry
    - HTTP return code
    - Headers
    - Payload if any

It is recommended to keep the audit trail information at least for six months or according to the applicable legal requirements.

14.Operational Scheme Manager (OSM) related requirements

The OSM will collect, validate, maintain and when applicable, make available additional data related to the scheme participants to ensure an effective functioning of the schemes.

These data will be included in an OSM Directory, only accessible to scheme participants in a secured way (in a push or pull mode).
Only the modified data must be updated in the weekly publication.

Each scheme participant will be able to access all the needed data of the other scheme participants (to be able to send the scheme messages) and will also be able to see which options are supported/accepted (or not) by the other participants.

Data that must be provided to the OSM during the adherence process and whenever a change occurs.
Mandatory data:

- [1.1] Legal name of the Scheme Participant
- [1.n] Identification of the Scheme Participant
- [1.n] API endpoint (URL) of the Scheme Participant[3]
- [1.n] API documentation (URL) of the Scheme Participant[4]

Recommended data:

- [1.1] Commercial/Trade name of the Scheme Participant
- [1.n] Roles of the Scheme Participant (only for the SPAA scheme)

Each scheme participant must have a identifier. As described in chapter 5.3, the identifiers are given under the responsibility of the OSM, which must ensure their unicity per identifier type (the same identifier cannot be given twice) and unambiguousness (it identifies without ambiguity a single scheme participant).

When applicable, the scheme participants should inform the OSM about their role (as described in chapter 6). If the scheme participant does not intend to implement Payer then the URL can be filled in with 'not provided'.

The OSM will update the data directory on a weekly basis; therefore, the scheme participants must put in place deprecation mechanisms for data that is managed.

[3] Not applicable to Asset Brokers

[4] Not applicable to Asset Brokers

For instance, if data can only be changed weekly and API's endpoint (URL) are managed, if a participant adds a new URL, this has to be added on the next week where both URLs will be valid, the old one been "deprecated". The deprecated URL can then be removed at the next week.

The OSM can perform intraday urgent updates of the directory for security purposes (e.g., certificate corruption).

Annex 1: SPAA scheme specificities

1. User Identification and Authentication – SPAA Scheme

In the SPAA scheme, the user would be the Asset Owner (AO) or somebody rightfully acting on his behalf. The user identification is a key-feature that must be properly completed before going further with authentication and any business processes (transactional or data assets).

The purpose of identification is to make sure that the user is the relevant one and that no-one else will be annoyed by any notification or request.

This purpose might go beyond the strict identification of the user by also specifying a usage context. Some Asset Holders may know a same user with different profiles (private access, delegated access, business access etc.), according to scenarios listed below:

- Private access on his/her own account
  - E.g., the user wants to access his own banking account
- Delegated private access
  - E.g., the user wants to access his mother's banking account
- Company access on his/her own account
  - E.g., the user wants to access to his professional banking account
- Delegated company access
  - E.g., the user wants to access the banking account of their employer

So, SPAA functionality has to allow to manage the above cases where necessary in order to distinguish different usage context and simplify the Identification process of the customer between Asset Broker and Asset Holder.

Thus, the recognition process must provide an unambiguous result for a single couple of Asset Broker and Asset Holder, i.e., the identification key, upon which each actor of the value chain can rely.

In the context of an API, this identification key must be sharable between the API client (i.e., the Asset Broker) and the API server (i.e., the Asset Holder).

On one hand, this identification key shall reasonably be as stable as possible within time although identification process may evolve and change their identification keys and mechanisms. On the other hand, the identification key is not meant to provide any additional information (e.g., personal address, phone number, email address…) by itself.

However, according also to GDPR rules adopted by each Participant's country, it is still possible to use some of these pieces of data as identification key. Moreover, the identification key can also be used to retrieve additional information when possible. The possible identification keys will be defined in the different API specifications according to what is mandated by PSD2.

Annex 2: SRTP scheme specificities

1. Referenced Technical Solution Provider's attribution

'Referenced Technical Solution Provider' is a label given by the Homologation Body in the SRTP scheme.

Purpose: a scheme participant's proxy acts as a gateway to expose or consume other scheme participant's APIs.

A proxy can either be technically transparent, or visible (at the level of mutualised certificates). It is not an actor as it is always legally transparent because the liabilities and obligations stay at the level of each scheme participant.

For auditability purpose, the proxy MUST be given an identity by the OSM. The OSM MUST manage these entities as having an infrastructure's role: proxy. Other constraints might apply to the proxy depending on the scheme (such as a special kind of homologation).

A scheme participant can also act as an infrastructure proxy.

The proxy MUST provision its own QWAC authenticating itself toward callers and callees of APIs.

The proxy only acts as a gateway in the communication, but does not have, in its proxy role, any final customers (e.g., Payers or Payees).

When the use of proxies is made possible by the scheme, the Identification of the scheme participant MUST be provided at the scheme level in the message (e.g., attributes AT-N001 and AT-E005 for the SRTP scheme).

Although this indication in the message is redundant when a direct route API is used, it is RECOMMENDED to always populate this information even when the scheme makes it optional.

When a scheme participant wants to use a proxy as its entry or exit point, this route MUST be declared and managed by the OSM.

Scheme participants making use of visible proxies (at certificate level) MUST indicate to the OSM that their messages are signed by the proxy instead of themselves. In such case, the proxy must also provision a QSealC as explained in the "non-repudiation" section (chapter 11).

The communication and non-repudiation mechanism between the proxy and the scheme participants it represents must comply with the same security requirements, however implementations are left to private/commercial relationship between the proxy and the scheme participants.