OPEN
BANKING
EXCHANGE

PSD2 API Communities - Survey on Communication Security Practices

Table of Contents

About Open Banking Exchange Europe

Purpose

The revised Payment Services Directive (PSD2) came into force in January 2018, with a requirement deadline of 14 September 2019 to implement Strong Customer Authentication (SCA). At this point, all regulated entities (Payment Service Providers) had to ensure that they individually complied with PSD2 and the Regulatory Technical Standards (RTS) set out by the European Banking Authority (EBA).

There is a clear regulatory expectation that the financial industry will organise itself to make sure that the implemented solutions for PSD2 are interoperable. However, at the time of writing, there remains a few outstanding activities required to successfully achieve this expectation.

Open Banking Europe was created to support Payment Service Providers (PSPs) and Third Party Providers (TPPs) in meeting the Access to Account (XS2A) requirements of PSD2 and to facilitate the wider aims of Open Banking.

History

Following a series of stakeholder consultations that started in 2016 to determine industry requirements, PRETA S.A.S. launched Open Banking Europe to build a PSD2 Directory solution to support PSPs and TPPs in meeting the PSD2 XS2A requirements. The Open Banking Europe Directory Service was released in January 2019, providing a single, standardised reference point for banks to accurately identify which TPPs are authorised to access their interfaces and which roles and services they are authorised to perform on behalf of their customers. Additionally, a Transparency Directory has been developed to help TPPs understand developer portals, and to help Account Servicing Payment Services Providers (ASPSPs) understand TPP brands. Open Banking Europe continues to work with stakeholders on a range of initiatives to facilitate a greater understanding of Open Banking and enable collaboration between interested parties. Open Banking Exchange is a subsidiary of Konsentus Ltd.

Audience

Open Banking Europe is aimed at the following audiences:

- Competent Authorities
- Payment Service Providers (PSPs), including:
  - Account Servicing Payment Services Providers (ASPSPs)
  - Third Party Providers (TPPs)
- Qualified Trust Service Providers (QTSPs)
- Service Providers, Solution Providers, and relevant consultancies

Disclaimer

Whilst care has been taken to ensure that the information contained in this document is true and correct at the time of publication, there are still clarifications needed around PSD2's scope and implementation and thismay impact on the accuracy of the information contained within this document.

As such, Open Banking Europe cannot guarantee the accuracy or reliability of any information contained within this document at the time of reading, or that it is suitable for your intended use.

Summary of Methodology & Findings

Purpose of the Survey

All Payment Service Providers (PSPs) are required to communicate securely when providing access to account services. There is also an expectation that those who provide access services do so in an interoperable way, to reduce the difficulty of using the Application Programming Interface (API) or other secure channels.

This survey looks at how the security of communications has been defined by different API communities and what requirements have driven them to design their security solutions.

Key Findings

Whilst there are differences in the approach taken by the different API communities, there is much in common. They all use Transport Layer Security (TLS) with mutual TPP/ASPSP authentication to provide basic security. However, there are many identified limitations in the use of TLS as the only method of securing the communications between ASPSPs and TPPs (see Appendix B on page 17). These limitations are overcome through also supporting Digital Signatures carried in the HTTP header.

Different communities have made different choices when it is necessary to apply digital signature to PSD2 requests and responses, and what data needs to be protected. However, given that all API communities take the same general approach in securing PSD2 communication this should not prohibit interoperable security.

A key difference is the technical protocol used for carrying digital signatures, with some communities adopting HTTP Signatures (Cavage v10) whilst others using JSON Web Signatures (JWS RFC 7515).

This could potentially divide the overall PSD2 communities into two non-interoperable groups. However, Open Banking Europe (OBE), working with the API communities and the ETSI European Standards Organisation, are working on standard solution which is based on JWS but has the capability to protect HTTP header information as in HTTP Signatures (Cavage v10).

Background & Methodology to the Survey

The Regulatory Technical Standards (RTS) for PSD2 on strong customer authentication and common and secure communications in Commission Delegated Regulation (EU) 2018/389 [1], places requirements on Open Banking to use Qualified Certificates, as defined under eIDAS [2]. One of the possible uses of such certificates is to identify the originator of a transaction using a digital signature linked to a 'Qualified Certificate for Electronic Seals' (QSealC). The other approach is to use 'Qualified Certificates for Website Authentication' (QWAC) to authenticate the parties on a secure communications channel.

In March 2019, preliminary questions to some of the ASPSPs about securing PSD2 Interfaces showed that there were multiple ways of achieving the same goal.

At the same time, ETSI produced a liaison statement raising concerns about some the techniques being used to secure interface for PSD2, using the new PSD2 compliant eIDAS certificates.

A series of workshops were held with OBE and ETSI that brought together the API communities and the ETSI ESI group. Following the first workshop, a questionnaire was put together to understand how different Open Banking API communities addressed the requirements of the RTS for secure and open communications. This document presents the main results from this survey and suggests a way forward for further harmonising secure communications for Open Banking.

The API communities who contributed to this survey are:

- The Berlin Group pan-European payments interoperability standards and harmonisation initiative (https://www.berlin-group.org/)

- Czech Banking Association - Standard for Open Banking (https://www.czech-ba.cz/cs/aktivity/standardy/cesky-standard-pro-open-banking)
- Polish API standard for the Polish financial market (https://polishapi.org/en/)
- SIBS based in Portugal supporting international financial services (https://www.sibs.com/en/)
- STET based in France supporting European Open Banking (https://www.stet.eu/)

Survey Results

The details of all the responses for the survey are appended to this document. This section summarises the overall response. Further clarification is added to some of the questions to provide further information on the understanding of the reason for the responses.

Transport Layer Security

Q1. Is MTLS (Mutual client/server Transport Layer Security) using QWACs required?

Overall Response

All the API communities use mutual authentication based on Qualified Website Certificates (QWACs) with Transport Layer Security [5].

Further Clarification

The TLS protocol protects the integrity and confidentiality of data during communications, and the QWACs provide mutual authentication between the communicating TPP and ASPSP.

Q2. What, if any, are the limitations of the use of MTLS?

Overall Response

Many of the API communities identified limitations with the use of TLS with QWACs for secure communications.

Further Clarification

The external communications with the Payment Service Provider (PSP) is often terminated at a general purpose gateway and the transaction forwarded by the gateway through an internal trusted network. In this situation, the TPP's identity certificate (QWAC) is bound to the secure transport layer channel up to the gateway and would not be forwarded to the payment application.

Similarly, intermediate parties may relay the transaction over a separately secured connection. Again, the TPP identity certificate (QWAC) is bound to the transport channel up to external relay and would not be forwarded by to the payment application.

Even if there is no gateway or relay system as the TPP identity certificate is not bound to the transaction data. Thus, if there is a dispute over the transaction, after it has been processed and logged, the source is not provable and so the transaction maybe repudiated.

See Appendix B on page 17 for illustration of the protection provided by TLS, its limitations, and how this is overcome by digital signatures.

Digital Signatures - When & What to Sign

Q3. Are TPP digital signatures required on requests?

Overall Response

All the API communities recognised the need to be able to protect TPP to ASPSP requests with digital signatures.

Further Clarification

Many considered it necessary only to protect certain, whereas some (in particular SIBS) applied digital signatures to all communications.

Q4. Are ASPSP digital signatures required on responses?

Overall Response

There were varying views of the need to protect ASPSP to TPP responses.

Further Clarification

Some did not consider it necessary to protect ASPSP to TPP responses with digital signatures. Some considered it necessary for some operations. One respondent indicated that it applied signatures to all responses.

Q5. What information, other than the payload, needs to be protected by the digital signature?

Overall Response

There were varying responses to what information, other than the transaction payload, is considered necessary to protect using a digital signature.

Further Clarification

For some respondents, it was considered that no other information than the payload need be protected. Others considered that much of the HTTP header contained important information about the transaction and so should be protected by the digital signature. The protection of any information used to identify the payment service user was commonly seen as important. Also, the time of signing, as required for long term validation of signatures, was identified as necessary in some cases. Some consideration is being given to the need to also bind the signing certificate to the digital signature.

Other Technical Requirements on Secure TPP to ASPSP Communications

Q6. Is it required that signatures are relayed transparently via intermediate systems?

Overall Response

In many cases it is required that signatures, and the authenticated identity they represent, are passed transparently through intermediate systems,

Further Clarification

See also limitations on transport layer security in Q2 on page 5.

Q7. Are there any special features of architecture which impacts requirements?

Overall Response

Features mentioned:

- Lightweight support for signatures having minimal impact on transactions
- Use of trusted intermediary to verify signatures on behalf of PSPs
- Support for OATH2 including protection of call back to TPP after user authentication

Q8. Is it required to apply signatures on HTTP payloads independent of the payload format (e.g., to transparently carry ISO 20022 based payload)?

Overall Response

Most respondents required support for legacy transaction formats such as ISO 20022.

Q9. Is there a requirement to authenticate HTTP Requests without any body (e.g. GET or DELETE)? (note Other Signed Information above)

Overall Response

Generally, it is required to be able to sign requests for some operations which do not have any body.

Q10. Is there any other information that is relevant to the requirements for secure TPP to ASPSP communications?

Overall Response

Other requirements mentioned:

- Carrying a range of payloads including ISO 20022, JSON, XML and Binary
- Support for OATH2 standards: RFC 6749, 6750, 7009, 7591 and 7592 and OATH MTLS

How to Sign

Q11. Are QSealCs used with public key cryptography based digital signatures used for signing?

Overall Response

Qualified Certificates for electronic seals are used by all the respondents, but in some cases an alternative may be selected.

Q12. Is it required that signatures are created using a Qualified Electronic Signature/Seal Creation Device to make them 'Qualified' under eIDAS?

Overall Response

Most respondents do not require the use of a Qualified Electronic Signature/Seal Creation Device.

Further Clarification

In addition to specifying requirements on particular forms of Qualified Certificates the eIDAS regulation [2] also specifies requirements on the hardware device which holds the key used for signing called a Qualified Signature or Seal Creation Device (QSCD). The PSD2 RTS [1] makes does not require use of a QSCD.

Q13. Is the signature held in the HTTP header?

Overall Response

All the respondents carried the signature against a transaction payload in the HTTP header.

Q14. What existing standards are used as the basis of applying digital signatures to the PSD2 secure communications?

Overall Response

Two different protocols adopted in carrying an HTTP header. One is the Internet Draft for HTTP Signature (Cavage v10). The other is to use JSON Web Signatures (JWS) as defined in Internet RFC 7515.

Further Clarification

The main feature of HTTP Signatures (Cavage) is that HTTP header information is protected as well as the payload as illustrated in the following diagram. This has been adopted by those requiring more than the payload to be protected (see 0 on page 5):

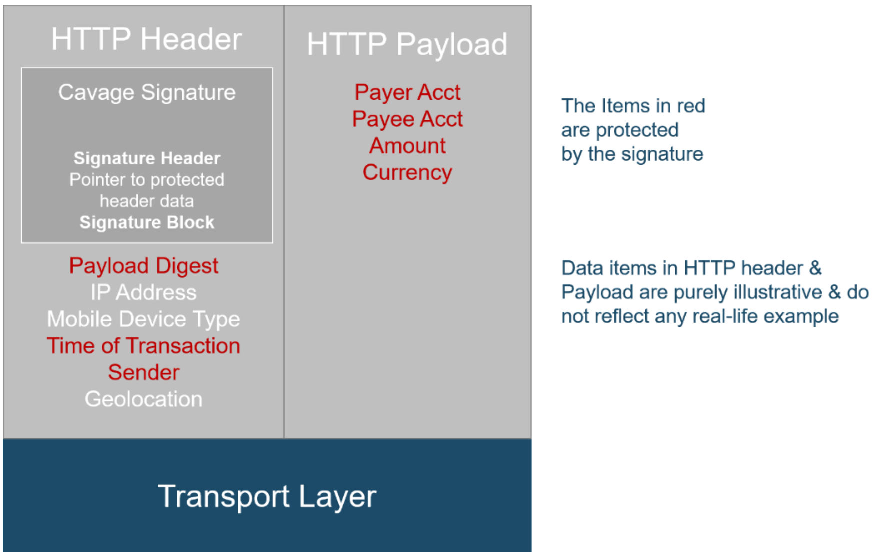


Figure 1: Illustration of HTTP Signature (Cavage v10)

As illustrated in the following diagram, RFC 7515 JSON Web Signatures do not protect the HTTP header information. It does have security improvements in that the properties of the signature are signed. This has been adopted by those only requiring the payload be protected (see 0 on page 5):

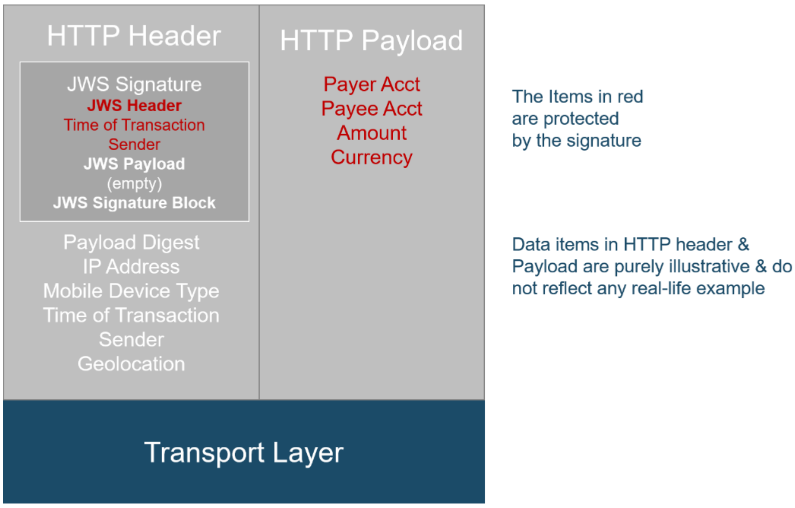


Figure 2: Illustration of HTTP Signature (Cavage v10)

OBE are working the API communities and ETSI (the European Standardisation organisation defining standards for eIDAS [2]), to define a standardised approach which adds a feature of protecting HTTP Signatures to the RFC 7515 standard, thereby avoiding a divergence in approach to applying digital signatures between the API communities.

Security Properties of Signatures

Q15. Is it required to have ‘non-repudiation’ properties of the digital signatures (i.e. can be independently verified subsequent to operation, possibly using other information available from logs, etc.)?

Overall Response

Signatures with non-repudiation are required in some cases.

Further Clarification

It is suggested that this requirement can be met with logs. However, for long term validity under validation rules defined by ETSI, a signature is required to have a "proof of existence" at the time that it is claimed to have been created. Also, the revocation information and CA certificates used to validate a signature need to be retained.

Q16. Is the digital signature required for authentication and identification?

Overall Response

The primary requirement under the RTS [1] is for identification. It is also recognised that this protects the integrity of the business information being signed.

Q17. Is the digital signature also required to support encryption?

Overall Response

Some communities also support encryption of the payload in addition to the encryption provided by TLS while the information is being exchanged.

Further Clarification

Whilst qualified certificates may be used for establishing keys used later for encryption their primary purpose is for identification.

Appendices

Appendix A: Detailed Reponses

The following table is a collection of information provided by PSD2 API communities in response design questions identified by OBE. This was initially produced following the meeting on digital signature formats for PSD2 on 2 July 2019, updated by written input from API Communities and then changed as discussed at the meeting on 24 September 2019.

	Berlin Group	Czech Open Banking	Poland	SIBS (Portugal)	STET	UK Open Banking	Comparison
Transport Layer Security							
Q1. Is MTLS (mutual client/server transport layer security) using QWACs required	Yes, for all communications	For all communication except OAuth2 resources	Required for all communication	For all communications	Yes. As specified in OATH MTLS	Yes. MTLS For all communications. QWACs supported but also OBIE root also supported. User choice.	Generally same for all communities
Q2. What, if any, are the limitations of the use of MTLS?	If the TLS connection endpoint is provided by a separate service provider the ASPSP has to solve the problem how to get the information contained in the QWAC from this service provider.	Complicated use of MTLS authentication on user centric API resources. Not possible use it if the direct consumer of this API is PSU's mobile application.	None	No limitations identified considering it is used just to establish a secure channel between the TPP and SIBS providing confidentiality and authentication		When applied to eIDAS (with multiple roots & used for identification, as opposed to just securing communication). Difficult to maintain digital records and evidence for non-repudiation. Expensive if the TLS connection endpoint is provided by a separate service.	Generally recognise limitations

	Berlin Group	Czech Open Banking	Poland	SIBS (Portugal)	STET	UK Open Banking	Comparison
Digital Signatures – When & What to Sign							
Q3. Are TPP digital signatures required on requests?	Required for certain operations (if requested by the ASPSP)	Required only for requests that perform active operations (POST, PUT, DELETE)	Required for all communication	Required for all communications	Yes	As required	Mostly TPP signatures used as required, SIBS always
Q4. Are ASPSP digital signatures required on responses?	Not required	Not required, only recommended in response to TPP signed request	Required for all communication	Not required	Optional ASPSP Choice	As required	Some not required, some optional
Q5. What information, other than the payload, needs to be protected by the digital signature?	Only if TPP signature is requested: • request-ID • PSU-ID • PSU-Corporate-ID (if applicable) • TPP-redirect-uri For next version: • TPP certificate (additional to above)	COBS does not specify this, it is on each of the ASPSPs.	No	Signing time PSU-Id Other - transaction identifier Under what situations: All operations	Signing time Not yet PSU identifier/ authorisation OATH2 Token or Payload or Dynamic Linking (Depending on type PSP) Geolocation & other fraud based parameters HTTP URL and Query parameters For all signatures	Signing time Yes - seconds from 1970 Key ID PSP Identifier Trust anchor	Varying. Varied support for certificate protection and signing time as required by AdES.
Other Technical Requirements on Secure TPP to ASPSP Communications							
Q6. Is it required that signatures are relayed transparently via intermediate systems?	Yes	Yes	Not required	No	Yes	Any relay transparent - in which case relay holds TPP keys	Several involve relays. Requires signature above transport layer

	Berlin Group	Czech Open Banking	Poland	SIBS (Portugal)	STET	UK Open Banking	Comparison
Q7. Are there any special features of architecture which impacts requirements?	No	No	It depends how 'special features' are defined. There are some specific security requirements defined in Polish API standard that results from its architecture. There are also many recommended security requirements (not only technical, but also on the process and organisational level) defined in Polish API security standard.	SIBS support TPP certificate verification on behalf of all banks. Banks need not verify signatures and certificates.	The signature mechanism impact on the weight of the requests/responses must be as less as possible. Using RFC 7591 and RFC 7592 for registration of the OAUTH2 client (including to be used call-back URLs, certificates…). This registration intends to avoid any misuse of any call-back URL or certificate.		Lightweight. Use with OAuth including misuse of URL call back. Verification can be done by trusted intermediate systems. Avoiding misuse of call back.
Q8. Is it required to apply signatures on HTTP payloads independent of the payload format (e.g. to transparently carry ISO 20022 based payload)?	Yes	Yes	No	Yes	Yes Can be XML or JSON Separate signature from business data	Where content type is present normally set to application/json	Most required to support legacy payload formats. ISO2022 can encoded in XML or JSON.
Q9. Is there a requirement to authenticate HTTP Requests without any body (e.g. GET or DELETE)? (note Other Signed Information above)	Yes	Yes	There are no GET or DELETE requests defined in the standard	Yes	Yes for GET	Yes for Delete	Generally yes Poland not required

	Berlin Group	Czech Open Banking	Poland	SIBS (Portugal)	STET	UK Open Banking	Comparison
Q10. Is there any other information that is relevant to the requirements for secure TPP to ASPSP communications?	Content either JSON or ISO 20022				OAUTH2 RFC 6749, 6750, 7009, 7591 and 7592, and OATH MTLS	JSON / Binary or XML	Content JSON & ISO 20022 (encoded in XML or JSON) Linking MTLS authentication to authentication of signer. Registration TPP certificate to OATH2
How to Sign							
Q11. Are QSealCs used with public key cryptography based digital signatures used for signing?	Yes (as an option. Usage to be decided by the ASPSP)	Yes, with public key	Yes	Yes	Yes	Yes - also support other key material (e.g. non-certified)	Generally, yes
Q12. Is it required that signatures are created using a Qualified Electronic Signature/Seal Creation Device to make them 'Qualified' under eIDAS?	Not required	Not required	Not required	Not required. Signatures are accepted as soon as the TPP can sign using the key pair related with the QSeal certificate issued by a Qualified Trust Service Provider recognised under eIDAS Regulation	Yes	No	STET, SIBS QSCD required otherwise no. Not checked by relying bank.
Q13. Is the signature held in the HTTP header?	Yes	Yes	Yes	Yes	Yes	Yes	All use HTTP header to carry signature

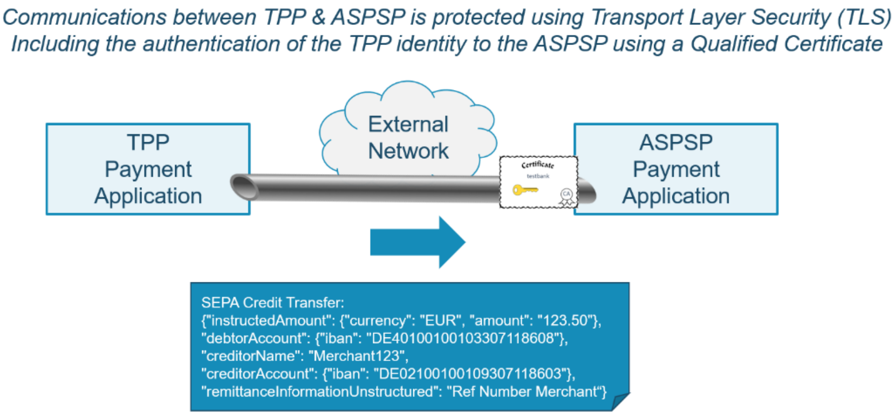
	Berlin Group	Czech Open Banking	Poland	SIBS (Portugal)	STET	UK Open Banking	Comparison
Q14. What existing standards are used as the basis of applying digital signatures to the PSD2 secure communications?	Security Properties of Signature	HTTP Signatures (Cavage) v. 10 or JWS	JWS (RFC 7515), JWT not allowed	HTTP Sig (Cavage v10)	HTTP Sig (Cavage v10)	Detached JWS (RFC 7515) carried in HTTP header or JWT is just used to support OATH2 includes signature but does not protect business information.	JWS or Cavage signature carried in HTTP header
Security Properties of Signature							
Q15. Is it required to have 'non-repudiation' properties of the digital signatures (i.e. can be independently verified subsequent to operation, possibly using other information available from logs, etc.)?	No special requirements defined by the specification of the API	Optional	Applicable TPP or ASPSP signatures or both	To be confirmed	Applicable TPP or ASPSP signatures or both. Using external logs of the requests/ responses	Evidence for non-repudiation on the HTTP Body (optional depending on API). This uses logs.	Can be required for some APIs. Can be based on logs.
Q16. Is the Digital Signature Required for Authentication & Identification?	Identification of TPP based on QSealC according to RTS. No special requirements defined by the specification of the API	Only to ensure undeniable data transfer	For registration. For protection of information passed via the PSU.	For protection of information passed via the PSU	For checking the business integrity of the request/response. For a posteriori proof in case of dispute (e.g. wrong amount or wrong beneficiary account number within a given payment request).	For protection of information passed via the PSU. UK holds central register of certificates their status.	STET – MTLS main means of identification. Identification of TPP required by RTS. Also protection of business information

	Berlin Group	Czech Open Banking	Poland	SIBS (Portugal)	STET	UK Open Banking	Comparison
Q17. Is the digital signature also required to support encryption?	No - encryption only at the TLS level	No	Digital certificates are used to support encryption (under TLS 1.2+ protocol). Encryption protocols should be in line with industry best practices (e.g. NIST recommendations).	No	No	Transport Specific app objects	
Further Information on APIs							
Q18. What is current version of API community specifications?	1.3.4	COBS 2.0 and proposed 3.0	2.1.3	PSD2 Product 2.1.13	STET 1.4.1 released in January 2019	Read/Write Data API Specification - v3.1.2	
Q19. Are there plans to significantly update the specifications? What is the timescale?	Version 2.0 planned at the end of 2019	It is planned to add more definitions (revisions) and new non-PSD2 functionalities in mid-2020.	Minor version planned at the end of 2019	Not yet defined	1.4.2 to be released in Q4 2019	No	Likely to influence beyond 2019 updates
Q21. Links to current specifications	https://www.berlin-group.org/psd2-access-to-bank-accounts NextGenPSD2 XS2A Framework, Implementation Guidelines	https://www.czech-ba.cz/cs/aktivity/standardy/cesky-standard-pro-open-banking	https://polishapi.org/en/	https://developer.sibsapimarket.com/live/product	https://www.stet.eu/en/psd2/	https://openbanking.atlassian.net/wiki/spaces/DZ/pages/1077805207/Read+Write+Data+API+Specification+-+v3.1.2#Read/WriteDataAPISpecification-v3.1.2-MessageSigning.1	

	Berlin Group	Czech Open Banking	Poland	SIBS (Portugal)	STET	UK Open Banking	Comparison
Q22. Other relevant documents	› draft-cavage-http-signatures › draft-ietf-oauth-mtls › RFC 5843 Additional hash algorithms › OATH2: RFC 6749 › RFC 3230	OpenAPI (Swagger) definition and issue tracker: https://github.com/Czech-BA/COBS	https://polishapi.org/wp-content/uploads/2019/09/PolishAPI-recommendations-security-v1.0.pdf	Berlin Group Implementation Guidelines v1.0	› draft-ietf-oauth-mtls › draft-cavage-http-signatures › OATH2: (RFC 6749, RFC 6750, RFC 7009, RFC 7591, RFC 7592)	› RFC 7515 JWS › RFC 7797 unencoded payload › RFC 7518 JW Algorithms › RFC 7516	

Appendix B: Illustration of the limitations of Transport Layer Security as the only security measure to secure communications & an explanation of how this is overcome through use of digital signatures

Transport Layer Security (TLS) provides a secure channel between two parties who are directly communicating. This has the capability of securely identifying TPPs to ASPSPs as illustrated below:

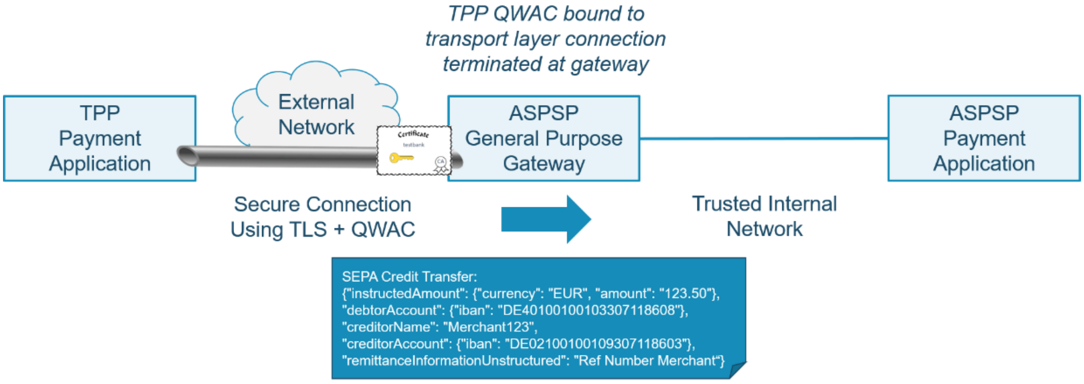

Communications between TPP & ASPSP is protected using Transport Layer Security (TLS)
Including the authentication of the TPP identity to the ASPSP using a Qualified Certificate
TPP Payment Application
External Network
ASPSP Payment Application

```
SEPA Credit Transfer:
{"instructedAmount": {"currency": "EUR", "amount": "123.50"},
"debtorAccount": {"iban": "DE40100100103307118608"},
"creditorName": "Merchant123",
"creditorAccount": {"iban": "DE02100100109307118603"},
"remittanceInformationUnstructured": "Ref Number Merchant"}
```

Figure B-1: Illustration of End to End Transport Layer Security

However, as indicated in section 0 0 this has limitations

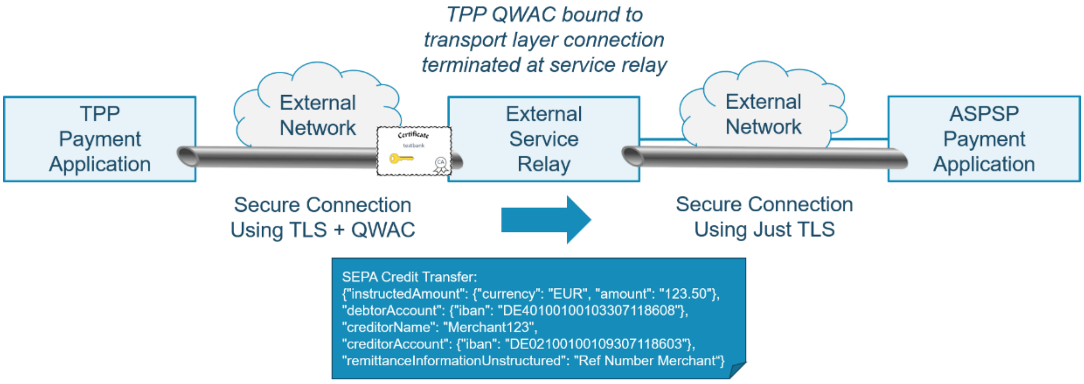
a) External communication with the payment service provider is often terminated at a general purpose gateway and the transaction forwarded by the gateway through an internal trusted network. In this situation, the TPP's identity certificate (QWAC) is bound to the secure transport layer channel up to the gateway and would not be forwarded to the payment application as illustrated in the following figure:


TPP QWAC bound to transport layer connection terminated at gateway
TPP Payment Application
External Network
ASPSP General Purpose Gateway
ASPSP Payment Application
Secure Connection Using TLS + QWAC
Trusted Internal Network

```
SEPA Credit Transfer:
{"instructedAmount": {"currency": "EUR", "amount": "123.50"},
"debtorAccount": {"iban": "DE40100100103307118608"},
"creditorName": "Merchant123",
"creditorAccount": {"iban": "DE02100100109307118603"},
"remittanceInformationUnstructured": "Ref Number Merchant"}
```

Figure B-2: Illustration of Transaction Passing Through ASPSP Gateway

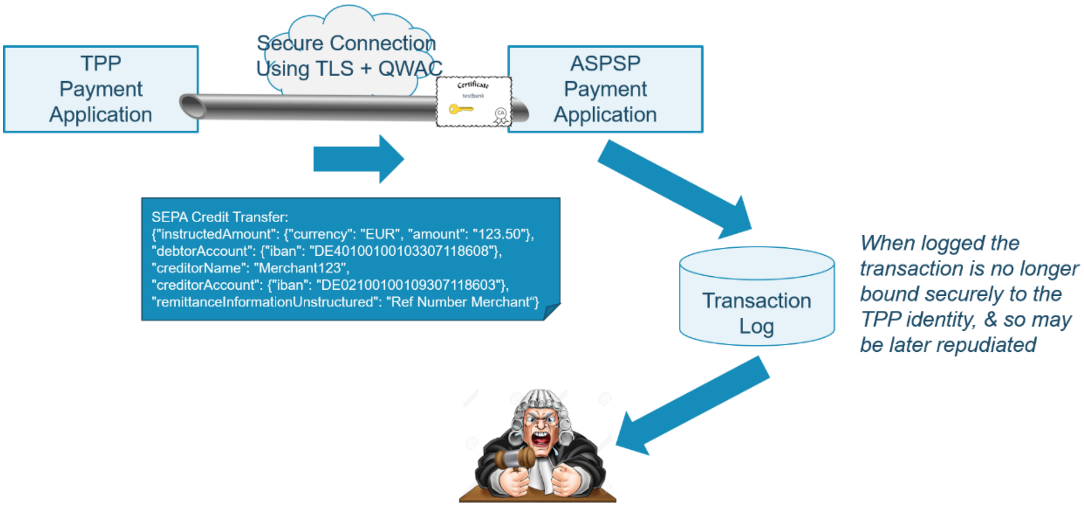
b) Similarly, intermediate parties may relay the transaction over a separately secured connection. Again, the TPP identity certificate (QWAC) is bound to the transport channel up to external relay and would not be forwarded by to the payment application as illustrated in the following figure:



```
SEPA Credit Transfer:
{"instructedAmount": {"currency": "EUR", "amount": "123.50"},
"debtorAccount": {"iban": "DE40100100103307118608"},
"creditorName": "Merchant123",
"creditorAccount": {"iban": "DE02100100109307118603"},
"remittanceInformationUnstructured": "Ref Number Merchant"}
```

Figure B-3: Illustration of Transaction Passed Through External Service Relay

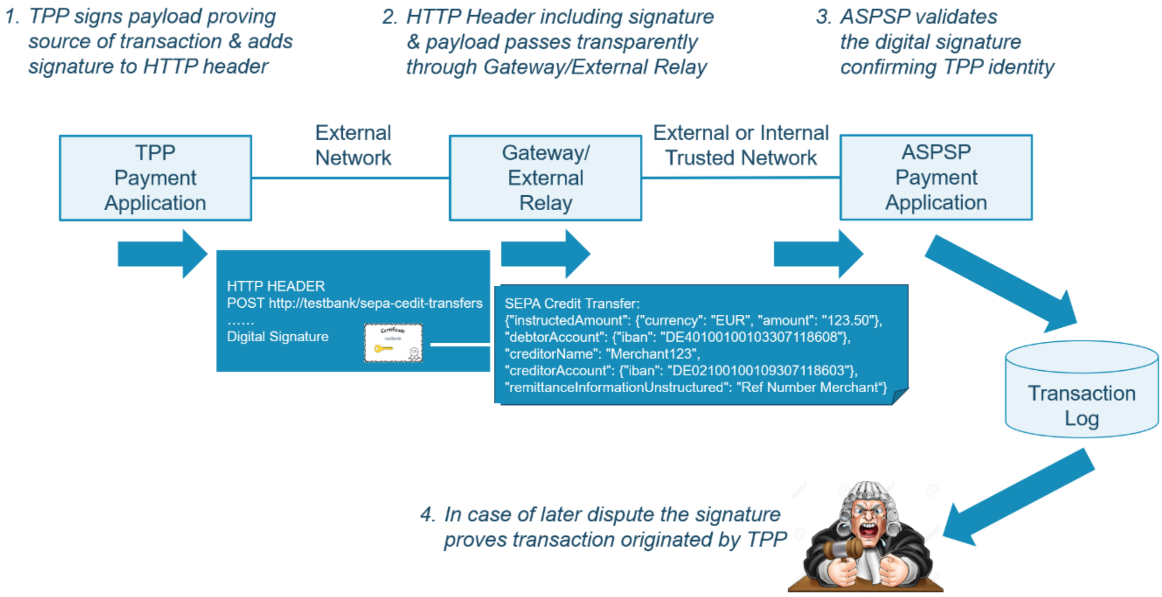
c) Even if there is no gateway or relay system as the TPP identity certificate is not bound to the transaction data. Thus, if there is a dispute over the transaction, after it has been processed and logged, the source is not provable and so the transaction maybe repudiated.



```
SEPA Credit Transfer:
{"instructedAmount": {"currency": "EUR", "amount": "123.50"},
"debtorAccount": {"iban": "DE40100100103307118608"},
"creditorName": "Merchant123",
"creditorAccount": {"iban": "DE02100100109307118603"},
"remittanceInformationUnstructured": "Ref Number Merchant"}
```

Figure B-4: Illustration of Disputed Transaction

Recognising the limitations identified above, all the API communities have added a second layer of protection based on the use of digital signatures with Qualified Certificate for Electronic Seals (QSealC). This signature is carried in the HTP header along with the payload as illustrated below:


1. TPP signs payload proving source of transaction & adds signature to HTTP header
2. HTTP Header including signature & payload passes transparently through Gateway/External Relay
3. ASPSP validates the digital signature confirming TPP identity
TPP Payment Application
External Network
Gateway/ External Relay
External or Internal Trusted Network
ASPSP Payment Application
Certificate
testbank
Transaction Log
4. In case of later dispute the signature proves transaction originated by TPP

```
HTTP HEADER
POST http://testbank/sepa-cedit-transfers
……
Digital Signature
```

```
SEPA Credit Transfer:
{"instructedAmount": {"currency": "EUR", "amount": "123.50"},
"debtorAccount": {"iban": "DE40100100103307118608"},
"creditorName": "Merchant123",
"creditorAccount": {"iban": "DE02100100109307118603"},
"remittanceInformationUnstructured": "Ref Number Merchant"}
```

Figure B-5: Illustration of Signatures Carried in HTTP Payload

As illustrated the TPP creates the signature, using its Qualified Certificate (QSealC), and adds this to the HTTP header providing proof of the source of the transaction. The HTTP header and payload passes transparently through any gateway or external relay. The ASPSP can then check the signature to confirm the source of the transaction and log the transaction. In case of any dispute the ASPSP has evidence that the transaction was originated by the TPP.

References

[1] COMMISSION DELEGATED REGULATION (EU) 2018/389 of 27 November 2017 supplementing Directive (EU) 2015/2366 of the European Parliament and of the Council with regard to regulatory technical standards for strong customer authentication and common and secure open standards of communication. Available at:

https://eur-lex.europa.eu/legal-content/EN/TXT/PDF/?uri=CELEX:32018R0389&rid=7

[2] REGULATION (EU) No 910/2014 of the European Parliament and of the Council of 23 July 2014 on electronic identification and trust services for electronic transactions in the internal market and repealing Directive 1999/93/EC

[3] Internet Draft draft-cavage-http-signatures-10 " Signing HTTP Messages"

[4] RFC 7515: "JSON Web Signature (JWS)". May 2015.

[5] IETF RFC 5246: "The Transport Layer Security Protocol Version 1.2".