Access Control Server (ACS) and EUDI Wallet - Integration Guide

DESCRIPTION OF THE ROLE OF ACCESS CONTROL SERVERS (ACS) IN PAYMENT USE CASES WITH THE EUROPEAN UNION DIGITAL IDENTITY WALLET (EUDIW)

Document properties

Name	Access Control Server (ACS) and EUDI Wallet – Integration Guide
Document Version	1.0
Status	Approved
Publication date	June 30, 2025
Contact	EURotterdamTrainingAndDoc@visa.com
Authors	Stan van Haasteren, Ranjiva Prasad, Stefan Kauhaus

Legal notices

The information, materials and any recommendations contained or referenced in this document (collectively, "Information") is furnished to you by the EU Digital Identity Wallet Consortium ("EWC", https://eudiwalletconsortium.org/) Payment Taskforce for informational purposes only.

While we aim to provide accurate and up-to-date information, the EWC and/or EWC Payment Taskforce is not responsible for errors in or omissions from this document. The Information is provided "AS-IS" and should not be relied upon for operational, marketing, legal, technical, tax, financial or other advice. The EWC and/or EWC Payment Taskforce make no warranty or representation of any kind, express or implied, about the completeness, accuracy, reliability, suitability, or availability of the Information, products, services, or related graphics contained in the document for any purpose, nor assumes any liability or responsibility that may result from reliance on or use of such Information.

Benefits/results are illustrative only and depend on business factors and implementation details. Any reliance you place on such Information is therefore strictly at your own risk. In no event will the EWC and/or EWC Payment Taskforce be liable for any loss or damage including without limitation, indirect or consequential loss or damage, or any loss or damage whatsoever arising, including from loss of data or profits arising out of, or in connection with, the use of this document or the Information.

The trademarks, logos, trade names and service marks, whether registered or unregistered, are the property of their respective owners, are used for illustrative purposes only and do not necessarily imply product endorsement or affiliation unless the Information indicates otherwise.

Please note that the Information may be updated or changed without notice, reflecting our ongoing efforts to provide the most current and useful information. By using the Information in this document, you agree to these terms.

Version History

Version	Date	Changes	Status
0.1	1st of April 2025	Created first draft	Draft
0.2	10th of June 2025	Comments (STi)	Draft
0.3	23rd of June 2025	Updated draft	Draft
1.0	24th of June	Published first version	Approved

Co-funded by the European Union

EWC
Co-funded by the European Union

1.Overview

1.1 Background

EWC Consortium (European Digital Identity Wallet Consortium) has been selected by the European Commission to experiment through Large Scale Pilots with the European Union Digital Identity Wallet (EUDI Wallet) in Travel, Business and Payment scenarios.

EWC has formed in 2023 its Payment Taskforce, led by Visa, with the objective of

- defining the EUDI Wallet payment specifications, build and pilot selected payment use cases
- identify barriers to adoption and evaluate opportunities in payment beyond Strong Customer Authentication, in particular by provisioning a card or account token in the EUDI Wallet and initiate an online or in-store payment
- use those specifications and findings to give feedback and offer inputs to the European Commission and future Payment and/or Digital Identity standards

This document is one of the deliverables of EWC's Payment Taskforce.

1.2 Scope

This document describes the role of EMV 3-D Secure Access Control Servers (ACS), which can play a role in authentication in EUDI Wallet use cases. It intends to explain the steps an ACS must perform in order to support these use cases.

1.3 Audience

This document is aimed at:

- European Commission ARF (Architecture and Reference Framework) experts drafting detailed specifications to enable EUDI Wallet usage,
- Organisations acting as Access Control Server (ACS) operators and product providers in the context of EUDI Wallet use cases.

1.4 Terms and Actors

3DS Server (3DS)[1]: Merchant's 3DS component that initiates an EMV 3-D Secure (EVM 3DS) authentication request.

Access Control Server (ACS): A provider, to a card Issuer/PSP, of authentication services using the EMV 3DS protocol. It receives authentication requests from the 3DS Directory Server (3DSS), to process those requests on behalf of the card Issuer. In Issuer-captured authentication, the ACS performs cardholder challenge.

Architecture and Reference Framework (ARF): Provides all the specifications needed to develop an interoperable EUDI Wallet Solution based on common standards and practices.

Attestation: A signed set of attributes, see Electronic Attestation of Attributes (EAAs)

Directory Server (DS): Card network's 3DS component. The DS has multiple functions in a 3DS ecosystem. In the scope of this document, the DS acts between the Merchant 3DS Server and Issuer ACS, routing transactions between the Merchant and Issuer domains.

[1] 3DS Server, Directory Server, and ACS are defined in more detail in EMV 3-D Secure specifications (www.emvco.com)

Co-funded by the European Union

Payment Wallet Attestation (PWA): Attestation issued to an EUDI Wallet instance by a (Q)EAA provider and which confirms that the instance can be used for SCA by the user of the instance.

Card credentials: A user's card details (also called card credentials) can take two different formats:

- **Primary Account Number (PAN):** The "long" number (usually 16 – digits) written on a payment card or
- **Payment Token:** A unique identifier generated by a card network as a proxy for the PAN, which can be used (as an option) by a Merchant or their payment service providers to store the card number in their system for future use. This provides enhanced security. Note that a payment card can be physical (e.g. a traditional piece of plastic) or virtual (a digital number only).

Card Network: A payment network that links cardholders, Merchants, and card Issuers to facilitate electronic payments. Examples of card networks include Visa, MasterCard, American Express, Bank Axept, Cartes Bancaires. The network provides rules and maintains systems enabling the functions performed by various stakeholders in the authentication (identity) and authorisation (blocking of funds) process.

European Union Digital Identity Wallet (EUDI Wallet**) :** The EUDI Wallet instance used to authenticate the user for a specific transaction (amount and Payee/Merchant name) and sends the result back to the user and to the authentication requestor. The requestor may be an ACS or a Merchant depending on which of the technical flow is used.

European Union Digital Identity Wallet (EUDI Wallet**) provider:** Provider (or Issuer) of EUDI Wallet instances to users.

EWC (European union digital identity Wallet Consortium): Consortium co-funded by the European Union to participate in the large-scale pilots to ensure interoperability and adoption of the European Digital Identity Wallet (learn more at https://eudiwalletconsortium.org/)

EWC Payment Taskforce: led by Visa, this EWC workgroup is composed of Identity and Payment experts dedicated to writing the specifications of how the EUDI Wallet will be used for card and account payments, and experiment those specifications in live pilots (see Appendix **Error! Reference source not found.**)

Key Binding JWT: This proves that the user has possession of the key bound to PWA

Issuer: A Person Identification Data Provider issuing Person Identification Data (PID) or a (Qualified) Trust Service Provider issuing (Q)EAA. In the case of the EUDI Wallet there may be multiple Issuers for PID and (Q)EAA.

Merchant's Acquirer: A Merchant's Payment Service Provider (e.g. a bank) that process debit or credit card payments on behalf of a Merchant by sending payment transaction information to the card network for authorisation (i.e. for blocking of the cardholder's funds/payment to the Merchant). Throughout this document, the term Payee PSP is used instead.

Merchant/Payee: The recipient of the payment, which is generally a provider of goods or services (can also be a marketplace facilitating sales for various providers, such as Expedia), located anywhere in the world (but initially most likely mainly European located Merchants). The Merchant is the one initiating a user's authentication request during a payment transaction.

Payer/PSU (payment service user)/User: Holder of a EUDI Wallet and of a card or payment account from a European PSP, purchasing good or services on a Merchant website or app who needs to be authenticated to pay for said purchase with her card.

Payment Service Provider (PSP): A generic term to designate a European provider of payment services. In this document, whenever the term PSP is used it specifically (and only) refers to either:

- A Card-Based Payment Instrument Issuer or,
- An ASPSP (Account servicing payment service provider)

In laymen's terms, those entities are often referred to as "Bank", but they can be any type of organisation with appropriate licenses to provide the above-mentioned payment services.

A PSP can work on behalf of the Payee (for example a Merchant selling goods) or a Payer (for example a consumer making a purchase). Therefore, this document refers to Payee PSP (often called acquirer) and Payer PSP, to indicate their role.

Co-funded by the European Union

Strong Customer Authentication (SCA): Requires that the Payer is authenticated through at least two factors, which must be independent from the other, from two of the three categories listed below.

- Something the Payers know (knowledge factor, e.g. a PIN code)
- Something the Payer has (possession factor, e.g. a mobile phone)
- Something the Payer is (inherence factor, e.g. biometric)

Trusted List Provider (TLP): Verifies the status of a role in the EUDI Wallet ecosystem. It provides a registration service for an entity performing a particular role(s) and maintains a registry ("Trusted List") to enable third parties to access registration information. In this use case, the card Issuer must check with a TLP that an EUDI Wallet provider and EUDI Wallet instance have a valid status on the Trusted List.

Verifiable Credential (VC): An Issuer-signed Credential whose integrity can be cryptographically verified. It can be any format used in the Issuer-Holder-Verifier Model.

Verifiable Presentation (VP): A Holder-signed Credential whose authenticity can be cryptographically verified to provide Cryptographic Holder Binding.

1.5 Use cases

In the context of financial transactions using the EUDI Wallet, the following authentication methods are possible:

- Method A: Issuer-captured authentication
- Method B: Merchant-captured authentication
- Method C: Merchant-captured authentication with a network token

An ACS plays a role in methods A and B, but not in Method C. Therefore, this document focuses on Methods A and B.

Co-funded by the European Union

2.Method A: Issuer-captured authentication

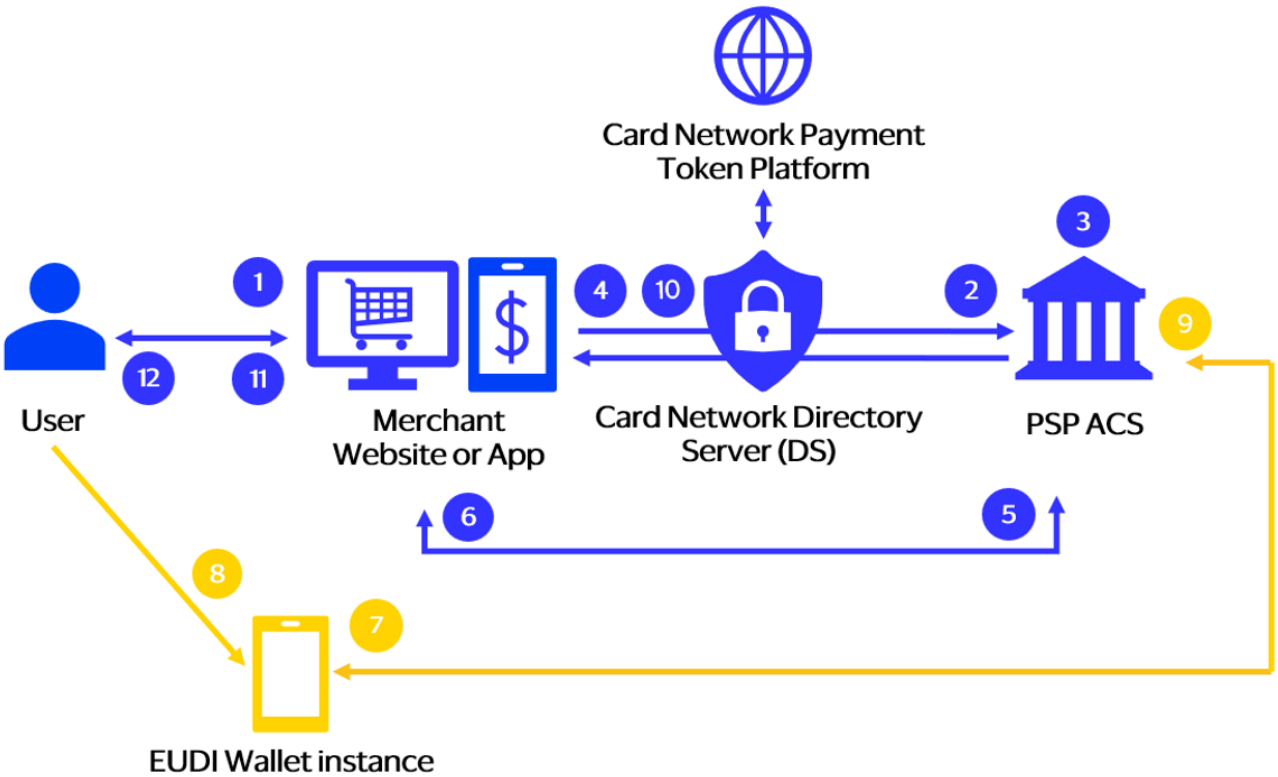
In the use case of Issuer-captured authentication, the ACS requests the authentication data, consisting of the Payment Wallet Attestation (PWA) and its associated key binding JWT from the EUDI Wallet. It then validates this data and returns the results of the validation to the Merchant via the card network Directory Server (DS).

2.1 Actors

The following actors play a role in this use case:

- User
- A valid EUDI Wallet Unit (Instance)
- Merchant
- Card Network DS
- Card Network Payment Token Platform
- ACS

2.2 Issuer-captured Authentication flow



EWC | Co-funded by the European Union

1. Consumer purchases goods and provides payment credentials (or confirms which "stored" credential to use).
2. Merchant sends authentication request to the PSP ACS via the appropriate card network DS. If a token was used the DS calls out the network token platform to detokenize and pass as a PAN to the ACS.
3. PSP ACS approves/declines/challenges authentication request (based on PSP risk policy). This diagram describes a challenge w here the PSP ACS has determined that a EUDI Wallet has been registered to the PAN.
4. PSP ACS sends response via the appropriate card network DS to Merchant with the intent to challenge.
5. Merchant sends a request to the PSP ACS to initiate the challenge.
6. In a merchant browser scenario, the PSP ACS returns a page displaying either a universal link that the user must click on to open their EUDI Wallet, or a QR code that the user must scan to open their EUDI Wallet. If the user is in a merchant app scenario, they will be automatically switched to their EUDI Wallet.
7. PSP ACS requests the Payment Wallet attestation linked to the PAN which was placed in the EUDI Wallet at registration and the Wallet Unit attestation. It passes the transaction details and the masked card number to the EUDI Wallet. The EUDI Wallet is invoked on consumer device (or has readiness to be invoked by the consumer).
8. The consumer authenticates into the EUDI Wallet. The EUDI Wallet asks the user to confirm sharing of the Payment Wallet attestation and Wallet Unit attestation for the purposes of authenticating the transaction details shown. Consumer confirms sharing for this purpose.
9. EUDI Wallet provides the authentication data (Payment Wallet attestation, cryptographic proof of dynamic linking, Wallet Unit attestation) back to the PSP ACS.
10. PSP or their ACS validates the authentication data and the ACS returns the results of the authentication back via the appropriate card network DS to Merchant.
11. In a merchant browser scenario and depending on implementation method, the EUDI Wallet asks the consumer to return to the merchant web site, if not automatically returned. In a merchant app scenario, the consumer is automatically switched to the merchant app when it is installed on the same device as the EUDI Wallet. When it is installed on a different device, the EUDI Wallet, asks the consumer to return to the merchant app.
12. Merchant confirms authentication success to cardholder.

Merchant must then proceed to authorisation flow (not covered in this flow).

2.3 Steps for ACS

- On receiving the authentication request (AReq, see 4.1) the ACS must determine whether a challenge is required. If so, it must establish whether an EUDI wallet has been registered against the PAN and therefore can be used to perform the challenge.
- Next, if the ACS receives a challenge request (CReq) it must:
  - Create the transaction_data object (see 4.5), using the transaction details passed through in the AReq.
  - Create an authorisation request (presentation request), requesting that the EUDI wallet present the Payment Wallet Attestation (PWA, see 4.3) and its proof of key binding (see 4.4) signed over the transaction_data object.
  - Display the presentation request as a QR code or universal link in the HTML returned in response to the CReq. If the EUDI Wallet Instance is on the same device as the purchase transaction, the cardholder will invoke the wallet using the universal link. If the purchase transaction is on a separate device (e.g. laptop), the cardholder uses the QR code to invoke the wallet (e.g. in user's mobile device).
- The Payment Wallet attestation (PWA) and the associated proof of key binding will be sent to an end point that the ACS specified in the presentation request. The PWA and associated key binding proof will be contained within a VP (Verifiable Presentation) token.
- On receipt of this information, the ACS should validate the PWA and the PWA proof of key binding according to the following steps.

2.3.1 Validation of PWA

To validate the PWA, the ACS must:

- Validate that the aud field of the payment wallet attestation is that of the card Issuer.
- Validate that the VP token was signed by the key bound to the payment wallet attestation.
- Validate that the payment wallet attestation proof was signed by the card Issuer.
- Validate that the sub field of the payment wallet attestation matches that held on file for this PAN / token.

EWC | Co-funded by the European Union

VIssuer

- Validate that the payment wallet attestation has not expired or been revoked.
- Validate that the wallet unit attestation presented during issuance of the PWA (registration) is still valid.

2.3.2 Validation of PWA Key Binding

To validate the PWA Key binding, the ACS must:

- Validate that the key binding proof was signed with the key bound to the PWA
- Validate that the nonce has not been presented before.
- Validate that the aud (JWT audience) parameter matches that associated with the Issuer in the Issuer-captured flow; or is associated with either the Merchant or 3DS Requestor in the Merchant-caotured flow.
- Validate that the iat (issued at, a timestamp of when the JWT was issued) parameter is no earlier than when the CRes was sent by the ACS and within a short period of time e.g. 10 minutes after it was sent. For the Merchant-captured flow the iat should be no earlier than a short period of time e.g. 10 minutes from when the AReq was received by the ACS.
- Validate that transaction_data_hashes matches the transaction_data object sent to the wallet for the hash algorithm specified in the transaction_data_hashes_alg. This may have been sent directly by the ACS to the wallet in the Issuer-captured flow or received in the AReq in the Merchant-captured flow.

Co-funded by the European Union

3.Method B: Merchant-captured authentication

In this flow, the Merchant requests authentication data (containing PWA and associated proof of key binding) from the EUDI Wallet in an EMV 3DS flow. This happens in Merchant environment before the EMV 3DS authentication flow is initiated by the 3DS Server. The difference with the Issuer-captured flow is that the Merchant does not attempt to validate the authentication data but sends them to the card network DS, which will pass it on to the ACS. The ACS then approves or rejects the transaction.

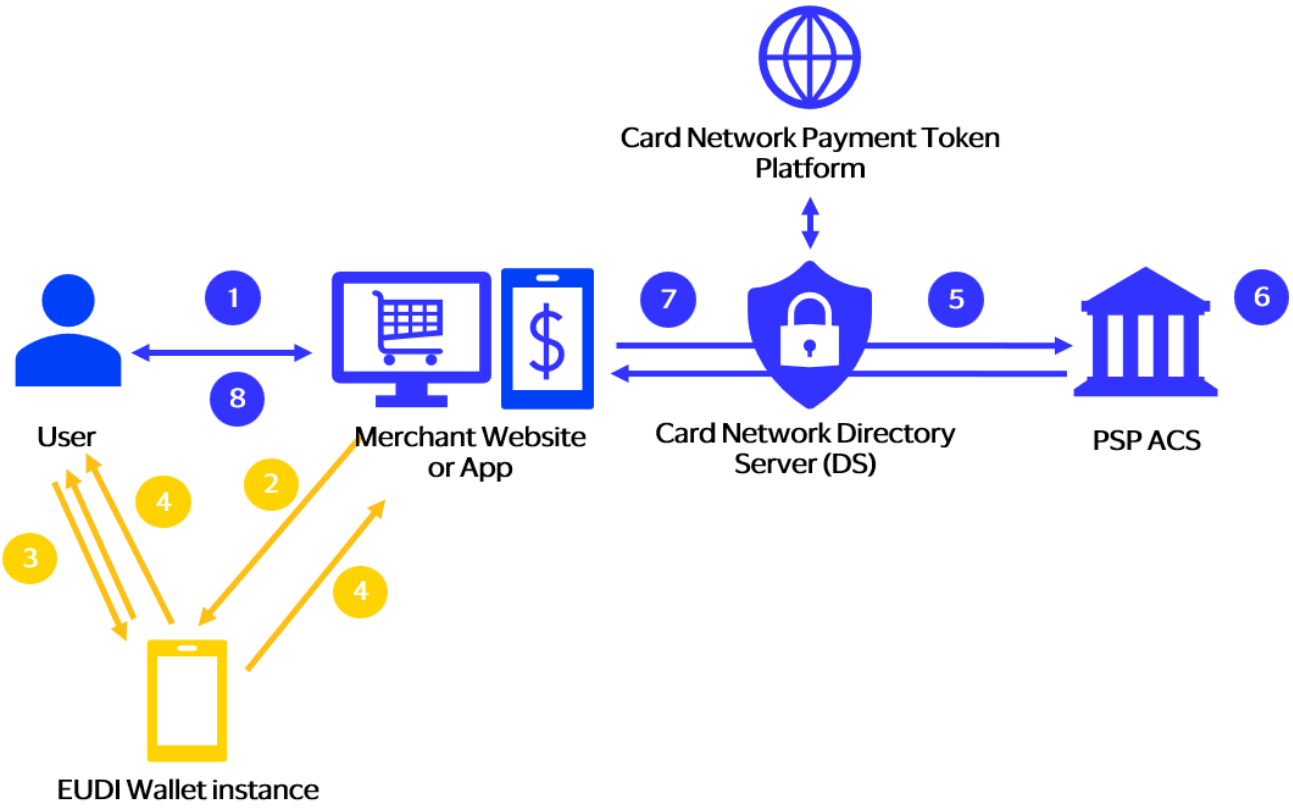
This flow is generic to apply both to browser or app. Some messaging details/steps may vary depending on which one will be used.

3.1 Actors

The following actors play a role in this use case:

- User
- A valid EUDI Wallet Unit (Instance)
- Merchant
- Card Network DS
- Card Network Payment Token Platform
- ACS

3.2 Merchant-captured Authentication flow



Co-funded by the European Union

1. Consumer purchases goods and provides payment credentials (or confirms which "stored" credential to use).
2. Merchant creates the presentation request to retrieve the Payment Wallet attestation linked to the payment credential. This was placed in the EUDI Wallet at registration. It passes the transaction details and optionally the last four digits of the card held on file, or keyed in by the cardholder to the EUDI Wallet. The presentation request may be displayed either as a universal link when the user is checking out at the merchant using the same device that their EUDI Wallet is installed upon, or a QR code when the user is checking out with a different device.
3. Consumer authenticates to the EUDI Wallet.
4. The EUDI Wallet asks the user to confirm sharing of the Payment Wallet attestation and Wallet Unit attestation for the purpo se of authenticating the transaction with details shown. Consumer confirms sharing for this purpose. The EUDI Wallet provides the authentication data (Payment Wallet attestation, cryptographic proof of dynamic linking, Wallet UNit attestation) back to the merchant.
5. Merchant sends the authentication data in an authentication request to the PSP ACS via the appropriate card network DS. This is where the EMV 3DS transaction starts. If a token was used the DS calls out the network token platform to detokenize and pass as a PAN to the ACS.
6. ACS approves/declines/challenges authentication request (based on PSP policy). This diagram describes a frictionless approva l.
7. PSP ACS sends authentication response back via the appropriate card network DS to Merchant.
8. Merchant confirms authentication success to cardholder.

Merchant must then proceed to authorisation flow (not covered in this flow).

3.3 Steps for ACS

On receiving the walletAuthenticationData (see 4.2) in the AReq (see 4.1), the ACS performs the following validations:

- Validate that the amount in the transaction_data object (see 4.5) match that sent in the purchaseAmount field of the 3DS AReq, if the transaction_data object recurring_schedule object is not present; or matches that sent in the recurringAmount field of the 3DS AReq, if the recurring_schedule object is present.
- Validate that the currency in the transaction_data object matches that sent in the purchaseCurrency field of the 3DS AReq, if the transaction_data object recurring_schedule object is not present; or matches that sent in the recurringCurrency field of the 3DS AReq, if the recurring_schedule object is present.
- Validate that the Payee in the transaction_data object match that sent in the merchantName field of the 3DS AReq.
- Validate that the value in the iss field matches either the Merchant or 3DS Requestor in the Merchant-captured flow
- Validate that the value of the aud field matches the card Issuer.
- Like the Issuer-captured flow, validate the PWA (see 4.3), the PWA proof of key binding (see 4.4).

EWC | Co-funded by the European Union

4.Data Objects

Below some info and examples of data objects the ACS needs to verify.

4.1 3DS Authentication Request

3DSReqAuthData =

{

“walletAuthenticationData”: <blob>

}

3DSReqAuthMethod = 04

4.2 Wallet Authentication Data BLOB

{

“iss”: “x509_sans_dns:shop.example.com”,

“aud”: https://superbank.com,

“iat”: 1541493724,

“transaction_data”: [“eyJwYXltZW50..ycmVuY3ki0”],

“vp_token”: “eyJhbGc..J9.eyjT2o..I6eyJ2In0.vcct70-bi1na..LZ38V-FJ8Ck7M”

}

4.3 Payment Wallet Attestation (PWA)

This is the VP token returned by the EUDI Wallet which is either directly or indirectly received by the ACS. It contains the PWA and key binding JWT in its verifiableCredential claim.

{

"iss": "https://mypsp.com",

"aud": "https://superbank.com",

"sub": "4f473c31-0e0a-4e54-8bb0-b69dc2a03b23",

"iat": 1541493724,

"nbf": 1541493724,

"exp": 1586247022,

"vct": "https://credentials.cardpaymentauthority.com/card-authenticator",

"panLastfour": "1234",

"iin": "444400",

“parLastfour”: “5D3E”,

"aliasId": "7d63281b-7c9b-4e5b-9121-053d8594bbfe",

"currency": "EUR“,

```
    "scheme": "Visa",
    "schemeLogo": "https://www.visa.com/logo.png"
    "cnf": {
        "jwk": {
            "kty": "EC",
            "crv": "P-256",
            "x": "TCAER19Zvu3OHF4j4W4vfSVoHIP1ILilDls7vCeGemc",
            "y": "ZxjiWWbZMQGHVWKVQ4hbSIirsVfuecCE6t4jT9F2HZQ"
        }
    }
}
```


4.4 PWA Key Binding JWT


```
{
    “nonce“: “1234567890”,
    “aud“: “https://example.com/verifier”,
    “iat“: “1698763547”,
    “sd_hash“: “Dy-RYwZfaaoC3inJbLslgPvMp09bH-clYP_3qbRqtW4”,
    “transaction_data_hashes“: [“TCAER19Zvu3OHF4j4W4vfSVoHIP1ILilDls7vCeGemc”],
    “transaction_data_hashes_alg“ : “sha-256“
}
```


4.5 Transaction Data Object

This is the Base64URL encoding of the transaction_data object sent by the Merchant to the EUDI wallet. It contains the "transaction_data_hashes_alg" which is the hash algorithm that the Merchant or ACS specified when constructing the *transaction_data* object. If this field is not present SHA-256 must be used to perform the validations.


```
{
    "type": "https://pay.example/trx/single_payment",
    "credential_ids": ["payment_credential"],
    "transaction_data_hashes_alg": "sha-256",
    “transaction_id”: “2edf9f89-0a8d-4ed0-864f-4917f089add6”
    "display": {
        "payee": "Kitchen Store",
        "currency": "EUR",
        "amount": "75.00"
    }
}
```