# Inclusive Purpose Consent Query (PCQ:)

## Version 7 KI-RIUP-WG draft - Date 2024-10-15

## Abstract

The Inclusive Purpose Consent Query is designed to send sufficient information to a smartphone to determine how to handle the query for personal information in a way that the holder can fully and quickly understand. This message might suffice for common, simple queries to acquire all the information a verifier requires. At the very least, it will establish a connection to a user agent (like a digital wallet) which can continue the query process. This document is not intended to be a complete formal specification but should be treated as an explainer for understanding the requirements for a broad range of inclusive use cases.

An inclusive query can be handled by all people who are entitled to access a service or other resource. To become inclusive a query must be able to be processed even when the subject is:

1. Not able to communicate in the local language,
2. Aware but not capable of handling the requirements of digital devices,
3. Unable to give any consent on their own behalf.

## Goals

This is the verifier subset of the identifier ecosystem’s goals which are stated elsewhere. This specifically addresses the needs of a human Holder of a wallet to get the information that’s needed to make an informed choice to share data.

1. The wallet holder can tap their device to a merchant-owned device, or navigate to a website using the same set of credentials. (ie in-person vs. online)
2. Functional for all subjects with digital credentials that are needed for their day-to-day transactions, with no exceptions.
3. If the subject needs a delegate to get necessary access, the wallet and verifier will accommodate multiple subjects or holders for a single device.
4. Works for first responders like medical technicians or disaster recovery operations where internet connectivity is not available.
5. Audit and fraud detection is built into the basic functionality.
6. A Query can be generated by a simple device in a small shop with all of the information required by the shop to complete the transaction. This should include payment as well as age verification, for example. An internet connection is not required to complete the transaction for any normal use case.
7. This query will help small verifiers to show compliance with privacy standards like the Kantara Privacy Enhanced Mobil Credentials (PEMC 2024).
8. Create delegated digital credential content such that the holder may access any resource that the subject wishes to delegate, either short-term or longer-term.

## Problem to be Solved

The user agent (which will be called the Wallet below) runs on a mobile device that enables a Holder to acquire credentials from Issuers and protect them with a Trusted Execution Environment (TEE) that may, or may not, be an integral part of the Device that hosts the Wallet. The diagram shows the Privacy Boundaries that need to be defined, centered around the Wallet, to protect user private data.

The Wallet knows nothing about the Verifier before the query is received. So the query must provide trust context to the wallet so that it can display a trust assurance that the user can understand. Presumably, the trust context would include a signature and certificate of some sort together with a Trustmark appropriate to the trust context.

When the user accepts the request then the first step of trust establishment is completed. The presentation response from the holder to the verifier should allow the completion of trust establishment. Trust establishment may fail at either step as either party may reject the establishment.

If the verifier rejects the presentation response from the holder, they may be able to continue the interchange by sending information to the holder’s wallet to allow a different response; for example if the holder has a different credential that might work.

The query is sent to the holder’s device which selects the appropriate wallet to process the request. If the device cannot find a wallet, it may be able to help the holder locate an appropriate solution.

## Context

There exist efforts to standardize the way that applications running on user devices can communicate. For example, the W3C WICG (Cappalli 2024) is working on a way for the browser to route a request to an appropriate wallet application to process the query string. What is missing is the means for a verifier to create a query request that can be captured by Near-Field Communication (NFC) or Bluetooth Low Energy (BLE) and route that request to an appropriate wallet application including any application that needs to be started to accept the request. This document addresses that requirement. It is suggested that device operating systems use this technique to direct requests coming into those (and similar) radio channels.

This document is dependent on the Kantara report on Digital Identifier Inclusion (Kantara RIUP 2024). The term holder is the controller of the Wallet. The subject of a credential in the Wallet might be the holder or some natural person who has delegated responsibility to the holder and wallet.

## Complexities

There are situations where multiple purposes can be requested resulting in multiple credentials in very different formats which might be processed in different code bodies in the verifier.

## Encryption of messages has been proposed to improve security and privacy. This means that the message may need to be decrypted before it can be determined if the message is to be addressed by one or more different functions that are not part of the receiving function. This

## User Experience

Success for this proposal will be an ecosystem for verifiers and subjects of credentials that is an improvement over what can be achieved today with a leather wallet containing cards issued with a variety of credentials that the holder needs in their normal activity of the day. The success of digital representations of credentials will only be possible if the holders and verifiers are satisfied with the results. Good experiences are already evident at many airports in the US. The major change needed is the presence of readers (like a kiosk or transaction terminal) at small merchants, door delivery personnel and other locations that holders experience multiple times a day. Another verifier that specifically addresses inclusion would be a safety-net service that needs to provide continuity of care or deduplication of services.

## Delegate Use Cases

Comatose, severely impaired or young child (Cognitively unable to Consent)

Language issues (Communications limitations to give informed consent)

Elderly parent that needs assistance (has become dependent; can delegate consent)

## Purpose

The purpose is designed to meet the desires of the verifier which includes compliance with local privacy requirements. The following wording is taken from the EU GDPR but should satisfy most jurisdictions. The EU website describes when data processing is allowed: “Data Protection under the GDPR” <https://europa.eu/youreurope/business/dealing-with-customers/data-protection/data-protection-gdpr/index_en.htm>

EU data protection rules mean the data controller (aka verifier) should process data fairly and lawfully, for a “**specified and legitimate purpose”** and only process “**the data necessary to fulfill this purpose”.**

## Name Value Pairs of data

| Name | Req | Value | Notes |
| --- | --- | --- | --- |
| ver:nam | R | Name of the verifier | Note that this will be in the localization of the verifier – if more than one alphabet is use, it should all be in one string (DBA) |
| ver:net | O | net ID of the verifier | Network address of the data processor uri, doi, did, etc. |
| ver:vid | O | ID of person  | an employee or badge number and may include if gov’t official or authority |
| rec | D | Record number | Required if data will be retained |
| ctd | D | Contact of processor | Required if data will be retained |
| jur | R | Jurisdiction or trust zone | EU US.CA HL7 |
| pur:pcd | 1 | Purpose code | See table below  |
| pur:typ | D | URI of purpose | Required if code = X |
| pur:exp | O | Epoch date of retention | If more than 24 hours after the current date for any purpose, the rec and ctd are required. |
| pur:dat | O | Data elements  | Any specific data elements required by the verifier. The goal is that this field is empty. |
| ath | O | Authentication factors | Any additional requirement for the wallet to provide, such as proof of presence or liveness |
| acc | O | Accepted protocols | Lets the verifier give the device and wallet hints about what protocols will be acceptable to the verifier. |

Requirements are: Required, Optional, and Dependent on other contents of the data or at least one entry is required.

Authentication may be indicated in the “ath” code if performed by the wallet, or by a Biometric purpose so that the authentication can be performed by the verifier.

The expiration date (exp) is the last time the data may be retained. Note that subsequent accesses may result in a new consent for the same data which is the equivalent of an extension of the retention period. It applies to the purpose so that a biometric authentication factor may have a short retention period (less than one day) and so is not considered to be retained by the verifier.

|  |  |  |
| --- | --- | --- |
| Purpose Code | Name | Notes |
| A | Age restriction for purchase or access | Followed by one or more integers, 13 18 21 |
| D | Driver’s License | Data needed for a license to drive |
| F | Fishing License | Data to show fishing or hunting license |
| E | Emergency | The verifier is a licensed first responder |
| X | The purpose is a URL | This is not for a point-to-point exchange |
| V | Visa or similar | Proof of permission to be in a country |
| B | Biometric data | An authentication factor |
| W | Right to work | May be from many different creds. |
| P | Payment required | May be followed by a currency code |
| L | Asylum request | From an application for any credential |

An example of an overall JWS payload that could follow this structure (before it is minified and compressed):

PCQ:{

 "ver": {

 "nam": "<<Verifier name>>",

 "url": "<<Verifier URL>>"

 };

 "rec": "<<record number>>",

 "ctc": "<<Contact https: or mailto:>>",

 "jur": "<<jurisdiction or trust zone>>",

 "ath": "<<other requirements for authentication of user>>",

 "pur": [

 "pcd": "<<purpose code>>",

 "typ":

 "<https://smarthealth.cards#health-card>",

 "exp": 1591037940,

 ]

}

## Message Flows and Experience

The query goes from the verifier to the holder’s device which determines which wallet application (wallet) in the holder's device gets the request. Once that user wallet has the query it creates a display for the holder’s consent. After the holder’s consent (which might only be to some of the purposes proposed by the verifier, the wallet builds a response to the verifier. The holder experience from the device or browser and the transition to the wallet will be key in user acceptance of this flow.

Verifier Authentication

It is often desirable for the wallet to authenticate the verifier so that the user is confident about who their counterparty is before sharing their data/ This is one way prevent fraudulent data requests. It is also possible for authentication of the verifier to support non-reputability, allowing wallets to present evidence that they were requested from a data set from a particular verifier, useful for reporting abuse of the system and inappropriate request patterns to governing authorities. However, methods to authenticate the verifier will vary significantly based on the protocols used for data sharing. It is recommended that the wallet combines the data from the `acc` field (accepted protocols) along with other request data to properly authenticate the verifier if supported by the protocols. Protocol authors are encouraged to create extensions to their protocols or specific guidelines on the interoperability of this specification and mappings to their own for correct authentication of verifiers (for example, see [9.2.4 mDL Reader authentication](https://mobiledl-e5018.web.app/ISO_18013-5_E_draft.pdf) in ISO/IEC 18013-5:2020). Implementers of relying party software making the requests should ensure that the data fields in the request are adequately populated to allow supporting wallets to perform the desired authentication of the verifier.

## Consent

The response from the wallet will come only when the holder consents to the query. The following are the considerations by the wallet in making that decision as to what data may be returned to the verifier. Biometric data is one element that needs attention as the biometric tests may be performed by the wallet but then the wallet must provide attestation as to its provenance. It is unclear whether privacy is improved by performing the biometric test in the wallet and thus requiring attestation about the wallet which could result in tracking data about the holder.

Note that one possible response is to establish a connection between one of the holder’s wallets and the verifier. In that case the consent is required to allow the verifier to get information from the wallet. The manner that consent is indicated in the device is under the control of the device and may include previous holder settings.

## Response to Verifier

This list includes all of the data either sent to the verifier as a consent to communicate, possibly with data for the similar cases.

Device identifiers that might be included in a wallet attestation:

* DeviceUniqueId – trackable undesirable
* Shared device indicator (holder not same as subject)
* Device binding - problematic unless it can be tokenized
* UserAgent / Wallet loadable package ID SBOM, etc
* Wallet app instance ID - trackable - can we tokenize this?
* Model
* Manufacturer
* Device Type
* AppID (from app store with version #) - not inherently traceable
* Bundle ID (apple & google)
* Build Number
* etc

| Name | Req | Value | Notes |
| --- | --- | --- | --- |
| dev:ref | R | Name of the provider | Typically, the o/s id and version |
| dev:net | O | net ID of the wallet | The way to access the user wallet app, if missing the device could not find an app that could respond to the request |
| rec | D | Record number | If provided by the Verifier |
| ctd | D | Contact of processor | If provided by the Verifier |
| Jur | O | Jurisdiction or trust zone | EU US.CA HL7 |
| res:pcd | 1 | Purpose code | See table |
| res:typ | D | URI of purpose | Required if code = X |
| res:exp | O | Epoch date of retention | If more than 24 hours after the current date for any purpose this is considered consent to allow retention for the time specified |
| res:dat | O | Data elements  | Any specific data elements required by the verifier |
| ath | O | Authentication factors | Response from request for authn |
| acc | O | Accepted protocols | This indicates the credential type responding. (could this be res:acc) |
| org | R | Origination data | Epoch date response created |

## References

Tim Cappalli 2024-03-01 “Web Platform and App Platform Layering / Interactions” <https://github.com/WICG/digital-identities/blob/main/resources/DigitalCredentialsAPI-Layering-v20240301.pdf> This addresses a similar problem from the point of view of the browser that is connected to the internet.

Kantara PEMC 2024-10-10 “REVIEW NOTICE FOR PUBLIC COMMENTS AND IPR REVIEW: PEMC Recommendations for Privacy Enhancing Mobile Credentials v1” <https://kantara.atlassian.net/wiki/spaces/GI/pages/683507722/PICPR20241010>

Kantara RIUP 2024-09-23 “Kantara report on Digital Identifier Inclusion” <https://kantarainitiative.org/download/riup-digital-identifier-inclusion-report/>