

Distributed Identity Assurance Specification

**Version:** 1.0.0 DRAFT 2

**Date:** 2020-02-04

**Editor:** Tom Jones

**Contributors****:** Jim Kragh, Jeff Brennan, Bev Corwin, Sal D’Agostino, Mary Hodder, Tom Sullivan, Noreen Whysel, Tom Jones

**Status:** Working Group APPROVED DRAFT

Kantara Working Group: Federated Identifiers for Resilient Ecosystems

Comments directed to: <https://github.com/KantaraInitiative/DistributedAssurance/issues>

**Abstract:**

Identity assurance is required for many high value applications. While assurance has been defined in other specifications as a centralized function where a Registration Authority and a Credential Service Provider (CSP) work together to create a user credential, this specification addresses the case where the subject seeks to leverage their existing high level of assurance at one provider to acquire a certificate from a CSP that can be carried with the user on their smart phone. The CSP combines an identity assurance statement from the existing provider (for identity proofing) with a software quality statement from the compliant app on the user’s smart phone (or other computing device - for authentication proofing). The result is a credential on the subject’s computing device that can be used with the certificate from the CSP to authenticate the user to any provider that is a part of a trusted ecosystem. This specification will give an example of providers that are part of a healthcare ecosystem where all members have agreed to protect the user's privacy at strictly enforced high levels.

**Notice:**

Copyright (c) 2020 Kantara and the persons identified as the document contributing authors. All rights reserved.

This document is subject to the [Kantara IPR Policy](https://kantarainitiative.org/confluence/display/GI/Option+Non-Assertion+Covenant)

[Abstract: 1](#_Toc29307891)

[1 Introduction 3](#_Toc29307892)

[1.1 Assumptions 3](#_Toc29307893)

[1.2 Goals 3](#_Toc29307894)

[1.3 Actors in the test use case 4](#_Toc29307895)

[2 Notations and Abbreviations 5](#_Toc29307896)

[3 Terms and definitions 6](#_Toc29307897)

[3.1 Trustworthy Digital Ecosystem 6](#_Toc29307898)

[3.2 Scenarios 7](#_Toc29307899)

[3.3 Results 8](#_Toc29307900)

[3.4 Contents of the Distributed Identity Assurance from the EHR 8](#_Toc29307901)

[3.5 Request Registration Certificate from the CSP 10](#_Toc29307902)

[3.6 CSP Validation of Record Locator DOI by the EHR 11](#_Toc29307903)

[3.7 Confirmation from the EHR about the DOI to the CSP 11](#_Toc29307904)

[3.8 Registration Certificate returned to the user Device 11](#_Toc29307905)

[3.9 Open Issues: 11](#_Toc29307906)

[4 Contents of messages 12](#_Toc29307907)

[4.1 Contents of Distributed Identity Assurance (DIA) 12](#_Toc29307908)

[4.2 Presentation and Delivery 13](#_Toc29307909)

[5 Distributed Assurance - JSON 15](#_Toc29307910)

[5.1 JSON Fields 15](#_Toc29307911)

[5.2 JSON Schema 16](#_Toc29307912)

[6 Conformance 17](#_Toc29307913)

[7 Considerations 18](#_Toc29307914)

[7.1 Distributed Assurance User Experience 18](#_Toc29307915)

[7.2 Privacy Considerations 19](#_Toc29307916)

[7.3 Security Considerations 19](#_Toc29307917)

[8 Acknowledgements 20](#_Toc29307918)

[9 References 21](#_Toc29307919)

[Appendix A: Categories of Data 22](#_Toc29307920)

[Appendix B: Example Distributed Assurance messages 23](#_Toc29307921)

[B.1 Human-readable Distributed Assurance– Simple 23](#_Toc29307922)

[B.2 Human-readable Distributed Assurance– Fancy 23](#_Toc29307923)

[B.3 JSON Contents 23](#_Toc29307924)

[Revision history 24](#_Toc29307925)

# Introduction

This concept of **Distributed Identity Assurance (DIA)** is described here as a use case for attaining IAL2 (abbreviations are defined in section 2) identifier assurance as well as AAL2 authentication assurance. No person creates their identity in a single place. It's made in the places we work and play, learn and advocate. So it is unlikely that any one’s identity can ever be completely encompassed by an authenticated identifier in one single [Credential Service Provider](https://wiki.idesg.org/wiki/index.php/Credential_Service_Provider). What people need is a collection of [Verified Claims](https://tcwiki.azurewebsites.net/index.php?title=Verified_Claim) that they can call upon as needed in their daily lives to control access to the high value information assets that represents their identity on line.

This concept is formalized in this "Kantara Distributed Identity Assurance Specification".

Assumptions

These assumptions on the existence of a trustworthy ecosystem are further described in section 3 but are not the subject of this specification.

* There is a trusted ecosystem consisting of a set of specifications, a trust anchor and a collection of digital endpoints which are registered as components of the ecosystem.
* The test use case for this specification is the US [Trustworthy Healthcare Ecosystem](https://wiki.idesg.org/wiki/index.php/Trustworthy_Healthcare_Ecosystem), but it is intended that it apply to other digital ecosystems as well.
  + The Patient or guardian (the user) has acquired a mobile phone that can protect the user credentials needed for authentication.
  + The Patient has been identified at a HIPAA covered healthcare provider by presentation of an insurance card, driver’s license and often a payment card or other evidence of financial responsibility.
  + The Patient has been (or will be) referred to another HIPAA covered healthcare provider.

## Goals

The goal for this specification is to enable the online users of one provider to leverage existing [Identity Assurance](https://wiki.idesg.org/wiki/index.php/Identity_Proofing) with other providers in a secure and privacy-preserving process.

Another goal is that the user has the protected data that they need to give consent to intentionally choose to move protected data outside of the protected environment described here. Otherwise the medical records will only be moved within a Trustworthy Healthcare Ecosystem.

## Actors in the test use case

1. Patient of one healthcare provider seeking services at another (referred) provider. In technical terms the patient is the resource owner of the protected health information (PHI).
2. A mobile device (e.g. smartphone) that can connect to the internet and securely run the mobile app. This device will also need to protect the user's private key when IAL2 or higher is required.
3. A mobile app that is certified by an appropriate federation authority to correctly relay the user's intent to any web site that needs the assurances described in this specification.
4. Assessor of claim of high assurance with provided Identifier (herein called the Credential Service Provider - CSP).
5. Existing provider of healthcare services. (Source of Identifier Assurance and later of patient's PHI.)
6. Another provider of healthcare services. (Sink of Identifier Assurance and later of patient's PHI.)
7. Trust Registry which can assure users that all digital actors are trustworthy by issuing digital trust indicators that attest to their membership in the trust ecosystem.
8. Medical Records Locator Service can find the patient’s records wherever they are located.

n. b. The terms “Source” and “Sink” are used in the sense of network data flows. A solid line arrow in the following diagram goes from the source of the data to the sink for the data.

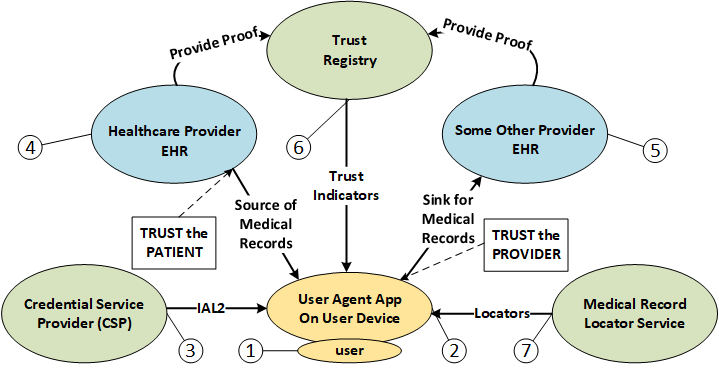


Figure 1 - The US Healthcare Assurance Framework

This specification uses the term “US Healthcare Assurance Framework” rather than the more common “[Federation](https://tcwiki.azurewebsites.net/index.php?title=Federation_Trust_Registry)” due to the large number of entities, governments and associations that are involved. Many of the concepts of federation are applicable to this test use case. This name will be used as a working place-holder for the trust\_framework element used in the DIA protocol with the assumption that the final term will be selected by the healthcare community with the cooperation of the [ONC – Office of the National Coordinator for health information technology in the US HHS.](https://www.healthit.gov/)

# Notations and Abbreviations

The keywords "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", “NOT RECOMMENDED”, "MAY", and "OPTIONAL" in this document are to be interpreted as described in [[RFC 2119](http://www.rfc-editor.org/info/rfc2119)].

All JSON [[RFC 7159]](http://www.rfc-editor.org/info/rfc7159) properties and values are case sensitive. JSON data structures defined by this specification MAY contain extension properties that are not defined in this specification. Any entity receiving or retrieving a JSON data structure SHOULD ignore extension properties it is unable to understand.

Definitions of the following terms are taken from the [Kantara Identity Assurance Framework](https://kantarainitiative.org/confluence/display/IAWG/IAF+1050+-+Glossary+and+Overview): (Kantara Identity Assurance Work Group, 2019) Assessor, Assurance, Authentication, and Credential Service Provider (CSP). Note that no distinction is made in this specification to the difference between Applicant and Subject, since the Subject Identifier (sub) is chosen by the user before the CSP is contacted. This sub is not of much use, however, until a CSP issues a certificate for the sub and it is always possible for more than one CSP to issue a certificate for the same sub with different levels of assurance.

The healthcare use case described here uses the term Patient to describe the owner of the protected health information (PHI), called the Electronic Health Records (EHR) at the provider. The user of the service is either the patient (aka subject) who owns the PHI, or a guardian which is delegated authority over the PHI in the EHR by the patient or appropriate sovereign entity.

User access to their PHI will be controlled by a **Credential** in their possession. In this specification the credential is created by a private key protected from disclosure in the mobile device. This is more fully described in [Phone as a Healthcare Credential](https://wiki.idesg.org/wiki/index.php/Phone_as_Health_Care_Credential).

AAL2 – Authentication Assurance Level 2 from [NIST 800-63-3B](https://pages.nist.gov/800-63-3/sp800-63b.html)

CR Consent Receipt [Specification Version: 1.1.0 Date: 2018-02-20](https://kantarainitiative.org/file-downloads/consent-receipt-specification-v1-1-0/)

DIA Distributed Identifier Assurance (defined in this specification)

DOI Document Object Identifier (in this case of the DIA or the medical record locator)

EHR Electronic Health records

GDPR General Data Protection Regulation

HIE Health Information Entity

IAL2 – Identity Assurance Level 2 from [NIST 800-63-3A](https://pages.nist.gov/800-63-3/sp800-63a.html)

IPR Intellectual Property Rights

JSON JavaScript Object Notation in IETF RFC 7159

JWE JSON Web Encryption in IETF RFC 7516

JWK JSON Web Key in IETF RFC 7517

JWKS JSON Web Key Set in IETF RFC 7517

JWS JSON Web Signature in IETF RFC 7515

JWT JSON Web Token in [IETF RFC 7797](https://tools.ietf.org/html/rfc7519)

ONC US Health and Human Services [Office of the National Coordinator](https://www.healthit.gov/)

PHI Personal Health Information

# Terms and definitions

This specification uses terminology and definitions from OpenID Connection and other specifications for JWT, JWE, JWS and JWK.

## Trustworthy Digital Ecosystem

It helps to understand how Distributed Assurance fits into the broader picture of a Trusted Digital Ecosystem by starting from the top of the trust chain and working down.

The simplest form of a digital ecosystem starts with a single **Trust Anchor,** which could be viewed as the one node to rule them all. This is the single source of trust from which all other entities in the ecosystem can trace the provenance of their trust. This is not about the centralized naming system of the internet, which will be assumed to be in operation, but rather about the trust that one node of the network can have with other nodes.

The trust in the trusted ecosystem is fully centralized as shown in the following diagram as it will be made available, at no charge, to any user in the ecosystem. The medical records are decentralized in the sense that there are expected to be a number of regional Health Information Entities where access to the patient’s PHI can be obtained. The assurance information is distributed in the sense that any EHR or HIE can be a source of patient identity assurance and patient matching to ensure that the right records are supplied to the right patient.

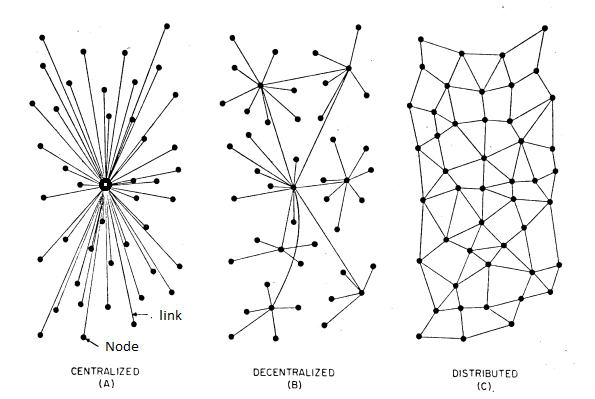


Figure 2 Paul Baran network types

## Scenarios

Primary Scenario:

1. Patient visits a HIPAA covered healthcare provider where they are known as a patient. After the medical visit they receive a paper with instructions for establishing a strong authentication credential on their smartphone.
2. Patient uses their smartphone to acquire the native app specific to their device that has been attested compliant with the code of conduct for such apps. (Today this would be the Apple or Android app store.)
3. Patient starts the app and picks a friendly name for a new medical identifier that will be assigned by the associated process.
4. Patient asks the app to "bind" the new identifier to the Distributed Identity Assurance claim that is also on the paper received in step one.
5. Typically the patient will use the smartphone's camera to "enter" the DIA claim into the phone (keying in the claim is possible, but not enjoyable.)
6. When the user has been acknowledged by their provider, the CSP will download a certificate to the user’s phone
7. The user can now use this smartphone as a credential to get access to any medical record of theirs that can be found using the record locator service.

A different path using biometrics: is under development.

Failed Paths:

1. Patient has no tolerance for technology and ignores or misunderstands the instructions or the purpose of the exercise.

## Results

Accepted Risks:

1. The patient loses the paper allowing some other person to attempt to steal their identity - mitigated by sign up process as described.

Post Condition:

1. If validation accepted by the CSP, the patient has a phone that can be used for sign in to any participating healthcare provider.

Dependencies:

1. Web Sites must be trusted before any user information is released.
2. Trust federations can be used to help users make informed decisions.
3. User consent and trust must begin with no user information transferred.
4. Standards exist to collect needed attributes where-ever they may be.

## Contents of the Distributed Identity Assurance Code from the EHR

This document is created by any organization that stores Electronic Health Records (EHR). The patient proofing is typically performed at an in-take or registration desk, which for most patients will be at the Primary Care Physician. The relationship between the in-take location and the EHR MUST be sufficiently trustworthy to enable the EHR to attest as to the assurance of the identity proofing process. The EHR will assign a record locator which can be used by any trusted care provider that has the patient’s consent to see those records. Also the EHR MUST provide the patient with a copy of their records in a common electronic format on demand. In order for the patient to be trusted by the EHR, the following steps will transfer the assurance information in a manner that can be trusted by all participants in the [Trustworthy Healthcare Ecosystem.](https://wiki.idesg.org/wiki/index.php/Trustworthy_Healthcare_Ecosystem)

There contents of the DIA from the EHR will contain the following fields; (see the following section 4 for a table of the fields.)

**doi Document Object Identifier**

This is a composite element consisting of the location of the record followed by a record number in the database. The location will be a DNS address in reversed form, the record number will be assigned by the EHR, and should be unusable by an attacker. The format will be: doi:tld.name.locator in url safe coding; for example doi:com.regence.3kjew98bl3. Note that the doi: part of the name may be skipped only in circumstances where there is no ambiguity. It is RECOMMENDED that the record number be the patient’s record locator number.

**Iat: Issue date-time of the DOI**

This field is used when the assurance is validated to ensure that after use by the patient it cannot be reused by an attacker. The EHR MUST use anti-hammering and reuse protection logic to assure the patient’s records do not fall in the wrong hands. Date time is in seconds since 1970-01-01. This code is RECOMMENDED for use in limiting the time for which the patient can use the code at the EHR to limit access by others. For example the code will only be allowed to be used for one week and only for 24 hours after. Note that the iat: part of the element may be skipped only in circumstances where the doi was also skipped.

There are multiple media that the EHR can use to communicate the DIA to the user for transmission to the CSP; for example, on the EHR web site or via a piece of paper given to the patient at their visit with the following information and QR code included: Other media for distributing the Distributed Identity Assurance Code are also acceptable.

**doi**

net.azurewebsites.controls.5741770546174656480

**doi\_date**

1576358115

QR code (a combination of the doi and doi date) is an example of one way to present the doi and date to the user on (for example) a piece of paper or the patient web site.



net.azurewebsites.controls.5741770546174656480;1576358115

Note that the link above is to a test sandbox site.

## Request Registration Certificate from the CSP

This message is sent by a user agent app on the user’s phone with the DOI and date from the EHR together with some information known to the user to assure the CSP that the message comes from the user, and a software statement to indicate the level of protection and user-presence is adequate assurance of authentication level 2 (AAL2).

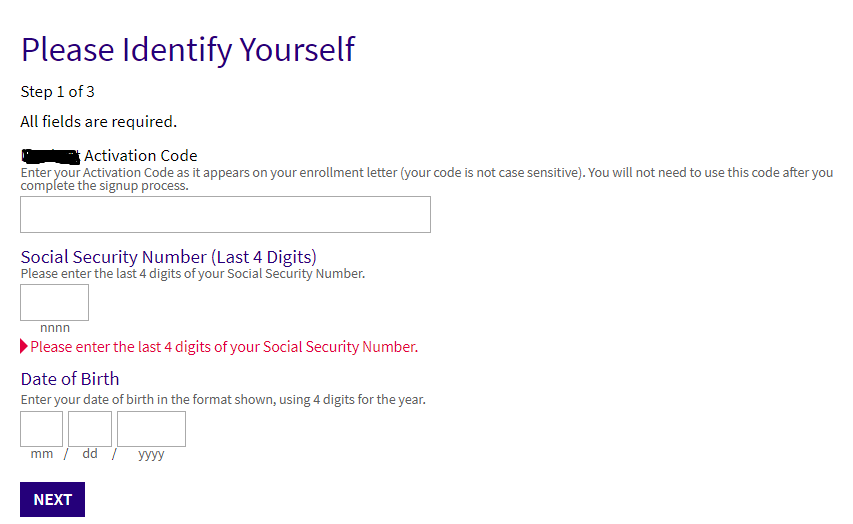
**Registration Ceremony**

The user needs to get the app for use on their mobile device before completing this step. The instructions from the EHR will tell the user how to acquire the app from the app store specific to their phone supplier. After the agent app is running the user will chose to create an identifier and add a binding of that identifier to the CSP. For IAL2 they will need the DIA code from the EHR as described above. For authentication assurance (AAL2) they will need to establish that their identifier is bound to a private key held in the Trusted Execution Environment on the Phone, called the KeyStore on Android. The Agent needs to have its own certificate informing the CSP that the app can be trusted to reliably report this information as well as user consent to proceed. The application MUST provide information to the CSP to prevent use of the code by anyone other than the patient or guardian. The following is an example of one implementation of the user experience in providing proof that they are entitled to access the EHR record on themselves.

**User Consent**

Note – must have notification address email or SMS phone number

The following is a non-normative example of what might be displayed to the user. This is only used to verify that this is the user that is identified in the activation code.



## CSP Validation of Record Locator DOI by the EHR

The CSP then sends the information from the user to the EHR to validate the identity assurance.

## Confirmation from the EHR about the DOI to the CSP

The EHR sends a confirmation about the level if identity assurance base to the CSP.

## Registration Certificate returned to the user Device

This certificate from the CSP is what makes the phone a credential with IAL2 and AAL2 assurance. The user is informed by their agent app that the binding of their credential to IAL2 and AAL2 has been successful, or given instructions on how to remedy the problem.

## Open Issues:

1. How does the user agent determine which CSP to call
2. How should we specify anti-hammering requirements

# Contents of messages

## Contents of Distributed Identity Assurance (DIA)

|  |  |  |  |
| --- | --- | --- | --- |
| **Distributed Assurance Transaction Details** Contents of the Distributed Identity Assurance (DIA) from the EHR | | | |
| Field Name | Definition | Guidance | Required |
| doi | The document object identifier  e. g. com.regence.{locator} | This field follows the Google app name convention rather than the internet standard | MUST |
| iat | Date that the DIA was created | This field is partially intended to create a unique Id for the doc. It is calculated at printing time. | MUST |
| **Request Registration (Binding) Certificate from the CSP**  Consent to create a binding | | | |
| Field Name | Definition | Guidance | Required |
| csp  iss | Name of the csp | Can be used to get the public key of the CSP | MUST |
| roles | The role that the binding will support | For example if level 2 the roles are:  Ial2 aal2 | MAY |
| doi | The document object identifier. | Assigned by the EHR | MUST |
| iat | Date that the DIA was created | This field is partially intended to create a unique Id for the doc. It is calculated at printing time. | MUST |
| sub | Subject Identifier per OpenID  e.g. for authn –IAL2 | Created by the user, or perhaps in cooperation between the user and the CSP for use in medical context. | MUST |
| trust\_framewrok | The URL(s) of the relevant organization in change | US Healthcare Assurance Framework | MUST |
| Contact info | Used by the CSP to create notification to the user | e.g. email, SMS phone number, physical mailing address | MUST |
| identification info | Attributes supplied by the user to prove presence at the creation of the request (expandable) | For example, a full birthdate could be sufficient as anti-hammering technology is required | MAY |
| exp | The date this consent expires | In unix epoch time | MAY |
| ssa | Software statement  For aal2 – also user gesture | Tells the CSP what level of protection is provided for user secrets | MUST |
| nonce | Unguessable string | Anti-hammering is also in place so the demands on this are small | MUST |
| Jwks  ?jwk set | Patient public key(s)  By ref or inclusion -keyid | there will probably one or two options to select from | MUST |
| signature | use patient private key | This proves that the patient has control of the device | MUST |
| **CSP Validation of Record Locator DOI to EHR** | | | |
| Field Name | Definition | Guidance | Required |
| doi | document identifier | at the EHR | MUST |
| iat | date code issued to patient | used to assure which document is used | MUST |
| trust\_framewrok | The url(s) of the relevant organization in change | US Healthcare Assurance Framework | MUST |
| evidence | Array of strings listing the evidence | Driver’s license, insurance card … | MAY |
| personal info. | Information supplied by the user to prove presence at the creation of the request | For example, a full birthdate could be sufficient as anti-hammering technology is required | MAY |
| signature | Using key of the CSP | protection from alteration | MUST |
| encryption | by key of EHR |  | MUST |
| **Confirmation from the EHR about the DOI to the CSP** | | | |
| Field Name | Definition | Guidance | Required |
| doi | document identifier | at the EHR | MUST |
| iat | date code issued to patient | used to assure which document is used | MUST |
| trust\_framewrok | The url(s) of the relevant organization in change | US Healthcare Assurance Framework | MUST |
| evidence | Array of strings listing the evidence | Driver’s license, insurance card … | MAY |
| Status | Of the subject’s record at this EHR | Active – inactive – died | MUST |
| signature | Using key of the CSP | protection from alteration | MUST |
| encryption | by key of EHR |  | MUST |
| **Registration Metadata returned by the CSP to the user Device** | | | |
| Field Name | Definition | Guidance | Required |
| cert | Signed cert | The structure below | MUST |
| ??? | Whatever is needed that is not in the certificate | Point of contact in CSP | ??? |
| **Registration Certificate returned by the CSP to the user Device** | | | |
| Field Name | Definition | Guidance | Required |
| csp  iss | Name of the csp  Needs to enable validation & policy | Can be used to get the public key of the CSP | MUST |
| sub | identifier of the subject | chose by the user | MUST |
| doi | Might be too much info for cert | tbd | MAY |
| JWKS | public key set of user | for signing | MUST |
| identity | from EHR | perhaps docs used in proofing by name | MUST |
| trust\_framewrok | The url(s) of the relevant organization in change | US Healthcare Assurance Framework | MUST |
| evidence | Array of strings listing the evidence | Driver’s license, insurance card … | MAY |
| iat | issued date for cert | Created by the CSP | MUST |
| Exp | Expiry data for cert | From CSP | MUST |
| Signature | Using CSP private key | Protection from alteration – proof of acceptance of the claims | MUST |

Table 1: Distributed Assurance fields

TK - Data structure will be provided after the schema is fixed

Figure 3: Distributed Assurance data structure

## Presentation and Delivery

Although an Assurance can be provisioned in any manner that is feasible or expected based on the context, an Assurance MUST be provided to the PII Principal in a human-readable format either on screen, or delivered to the PII Principal, or both. A JSON encoded CR MAY also be delivered to the PII Principal.

NOTE: Issues such as language translation, localization, human-readable layout and formatting, and delivery mechanisms are out-of-scope for this document.

# Distributed Assurance - JSON

## JSON Fields

This specification uses “named object” data types to describe the principal concepts within the Distributed Assurance and allows for extension by implementers.

See the JSON schema for object implementation.

| **JSON name** | **CR name** | **Data Type** | **Format/Example** |
| --- | --- | --- | --- |
| version | Version | string |  |

Table 2: Distributed Assurance JSON fields

## JSON Schema

{

TBD

{

# Conformance

tbd

# Considerations

Assurance is measure used by digital systems to evaluate claims about identity, security and privacy. This spec addresses protocol interoperability of assurance information as the primary objective. The following considerations are needed for a complete solution and are offered here as suggested guidelines.

With each ecosystem policy and a Distributed Assurance implementation, there are different UX, legal, privacy, and security-related considerations. This document uses an example from health care but that would not apply in other federations.

## Distributed Assurance User Experience

A Distributed Assurance combines the protection offered for the user’s private key credential with the distributed Identity assurance of the DOI which both are part of the trusted ecosystem to build an authentication app that the user can carry with them as they navigate both the real-world and the digital-world of their ecosystem.

### Identifiers (non-normative)

It is the objective of the user's agent to store authentication credentials and other user claims needed to provide the user with the level of assurance for identity and authentication needed for demanding web sites. Each persona that the user creates with have the level of support that it requires. For example the following personas might be appropriate for Fred if he want to have a normal persona, as well as a high assurance medical persona and to act as guardian for his under 13 year-old daughter and a parent who has arranged for him to act as guardian.

1. Fred-personal = a common self-issued persona with no special assurances
2. Fred-medical = a high assurance self-issued persona with level 2 identity and authentication assurances for access to his own medical records.
3. Daughter-annie = a high assurance self-issued persona with level 2 identity and authentication assurances and access to Annie's medical records.
4. Grandfather-john = a high assurance self-issued persona with level 2 identity and authentication assurances for access to Fred's father's medical records.

While many use case will have multiple personas as shown above, there is no reason that a single persona user agent should not be produced as well. In any case each persona must be able to handle identity assurances from any compliant provider, and record locators at any number of other compliant providers. Acting as identifier provider for the user, the agent in the user's device may also crate any number of subject identifiers spanning a range of networks including decentralized identifiers (DIDs) defined over block chains; but none of this complexity need to be normally visible to the user. It can all be hidden inside user-opaque digital protocols.

## Privacy Considerations

The major objective of this section is to assure that solution can be configured by most developers to meet the privacy requirements of the evolving privacy legislation as well as the IDEF guidelines. TK link to IDEF.

In this document, sensitive data collection is indicated. NOTE: In multiple jurisdictions, there are categories listed as sensitive personal information. If you use, collect or disclose sensitive personal information these have legal requirements, require explicit consent and can have jurisdiction-specific legal notice requirements to be informed. For example, user information about sexually transmitted diseases would need particular care before being released.

For example the sensitivity levels are described in [Trustworthy Healthcare Provider](https://wiki.idesg.org/wiki/index.php/Trustworthy_Healthcare_Provider) document.

## Security Considerations

The major objective of this section is to assure that solution can be configured by most developers to meet the security requirements of the NIST 800-63-3B level AAL2 as well as the IDEF guidelines. The challenge is to encode the authentication assurances in a manner that can be reliably tested by the CSP and passed on to the relying parties.

### Private Key Protection

All user private keys must be enabled for AAL2 protection if requested. To prevent a stolen user device from allowing unauthorized use of private key, the user must be authenticated to the device before the private (or secret) keys are made available for use.

#### Android key protection (non-normative and may change over time)

The best working api for creating a Private Key in Android appears to be KeyGenParameterSpec added in Android M (API 23) as described in this post.<https://doridori.github.io/android-security-the-forgetful-keystore/#sthash.NijopPIO.dpbs> It is required that the Android device is protected with a device-lock set to prevent access without the code. This is a good security feature, but requires a minimum level of Android M. Fingerprint support is not solid until Android N.

#### Apple Key Protection (non-normative and may change over time)

### Protection for messages transmitted by any actor

The transmission of a JSON Distributed Assurance MUST enable validation of the integrity and authenticity of the receipt using the following specifications:

* JSON Web Token (JWT) [RFC 7519]
* JSON Web Encryption (JWE) [RFC 7516]
* JSON Web Signature (JWS) [RFC 7515]
* FHIR JSON Signature rules in [HL7 section 2.23.0.17.2](https://www.hl7.org/fhir/datatypes.html#signature) which contains options which are consistent with this specification. Note that detached signature are used in the listed messages, but not for the messages in this specification which are significantly smaller.

Notices

# Acknowledgements

The Distributed Assurance effort has been developed in the Kantara Community, supported by people who have invested in making this specification open and free to use. It is free so that people can have a common way to see their data control and sharing. If you wish to provide feedback, you may join the Kantara Working Group, and then email us tbd

In addition to Kantara, we wish to thank the following contributors to the Distributed Assurance effort:

list of names

The Distributed Assurance specification effort has been developed with the support of many communities, as noted in our acknowledgments section, and leverages best of breed standards, legal regulation and technical practices in its design and development, as noted in the references section.

# Works Cited

Group, K. I. (2019, 4 30). *Kantara Identity Assurance Framework: IAF-1050 2 Glossary and Overview* . Rtrieved from Kantara Initiative: https://kantarainitiative.org/confluence/display/IAWG/IAF+1050+-+Glossary+and+Overview?preview=/113869056/113869057/Kantara%20IAF-1050%20Overview%20Glossary%20v0.5.4.pdf

9, May 2015, <https://tools.ietf.org/html/rfc7519>  **[RFC 2119]** Bradner, S., “*Key words for use in RFCs to Indicate Requirement Levels*”, BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997 <http://www.rfc-editor.org/info/rfc2119>

**[RFC 4122]** P. Leach, M. Mealling, R. Salz, “*A Universally Unique IDentifier (UUID) URN Namespace*”, RFC 4122, 10.17487/RFC4122, July 2005, https://tools.ietf.org/html/rfc4122

**[RFC 7159]** Bray, T., Ed., “*The JavaScript Object Notation (JSON) Data Interchange Format*”, RFC 7159, DOI 10.17487/RFC7159, March 2014, <http://www.rfc-editor.org/info/rfc7159>

**[RFC 7515]** M. Jones, J. Bradley, N. Sakimura, “*JSON Web Signature (JWS)*”, RFC 7515, May 2015, <https://tools.ietf.org/html/rfc7515>

**[RFC 7516]** M. Jones, J. Hildebrand, “*JSON Web Encryption (JWE)*”, RFC 7516, May 2015, <https://tools.ietf.org/html/rfc7516>

**[RFC 7519]** M. Jones, J. Bradley, N. Sakimura, “*JSON Web Token (JWT)*”, RFC 7519, DOI 10.17487/RFC751

1. Categories of Data

(Explainers/Examples)

1. Example Distributed Assurance messages
   1. Human-readable Distributed Assurance– Simple

* 1. Human-readable Distributed Assurance– Fancy
  2. JSON Contents

{

}

# Revision history

|  |  |  |
| --- | --- | --- |
| **Version** | **Date** | **Summary of Substantive Changes** |
| DRAFT 2 | 2020-02-04 | Reference to KIAF 1050 Glossary for definitions.  Eliminated the concept of PCP in normative parts  Added reference to HL7 FHIR signature section  Rearranged section 7 to match IDEF categories  Fill out security considerations  Change federation name to The US Healthcare Assurance Framework and include in json  Clarified doi to be recommend as Record Locator  Add reference to Consent Receipt  Clarified Abstract and IPR policy  Add fields from OpenID Identity Proofing |
| 1.0.0 DRAFT 1 | 2019-12-21 | Initial v1.0 draft |