NIST Special Publication
NIST 800-63C-4 2pd

Digital Identity Guidelines

Federation and Assertions

Second Public Draft

David Temoshok
Justin P. Richer
Yee-Yin Choong
James L. Fenton
Naomi Lefkovitz
Andrew Regenscheid

This publication is available free of charge from:
https://doi.org/10.6028/NIST.SP.800-63c-4.2pd

NATIONAL INSTITUTE OF
STANDARDS AND TECHNOLOGY
U.S. DEPARTMENT OF COMMERCE

NIST Special Publication
NIST 800-63C-4 2pd

Digital Identity Guidelines

Federation and Assertions

Second Public Draft

David Temoshok
Naomi Lefkovitz
Applied Cybersecurity Division
Information Technology Laboratory

Yee-Yin Choong
Information Access Division
Information Technology Laboratory

Andrew Regenscheid
Computer Security Division
Information Technology Laboratory

Justin P. Richer
Bespoke Engineering

James L. Fenton
Altmode Networks

This publication is available free of charge from:
https://doi.org/10.6028/NIST.SP.800-63c-4.2pd

August 2024

U.S. Department of Commerce
Gina M. Raimondo, Secretary

National Institute of Standards and Technology
Laurie E. Locascio, NIST Director and Under Secretary of Commerce for Standards and Technology

There may be references in this publication to other publications currently under development by NIST in accordance with its assigned statutory responsibilities. The information in this publication, including concepts and methodologies, may be used by federal agencies even before the completion of such companion publications. Thus, until each publication is completed, current requirements, guidelines, and procedures, where they exist, remain operative. For planning and transition purposes, federal agencies may wish to closely follow the development of these new publications by NIST.

Organizations are encouraged to review all draft publications during public comment periods and provide feedback to NIST. Many NIST cybersecurity publications, other than the ones noted above, are available at https://csrc.nist.gov/publications.

Authority

This publication has been developed by NIST in accordance with its statutory responsibilities under the Federal Information Security Modernization Act (FISMA) of 2014, 44 U.S.C. § 3551 et seq., Public Law (P.L.) 113-283. NIST is responsible for developing information security standards and guidelines, including minimum requirements for federal information systems, but such standards and guidelines shall not apply to national security systems without the express approval of appropriate federal officials exercising policy authority over such systems. This guideline is consistent with the requirements of the Office of Management and Budget (OMB) Circular A-130.

Nothing in this publication should be taken to contradict the standards and guidelines made mandatory and binding on federal agencies by the Secretary of Commerce under statutory authority. Nor should these guidelines be interpreted as altering or superseding the existing authorities of the Secretary of Commerce, Director of the OMB, or any other federal official. This publication may be used by nongovernmental organizations on a voluntary basis and is not subject to copyright in the United States. Attribution would, however, be appreciated by NIST.

NIST Technical Series Policies

Copyright, Fair Use, and Licensing Statements
NIST Technical Series Publication Identifier Syntax

Publication History

Approved by the NIST Editorial Review Board on YYYY-MM-DD [will be added upon final publication]

How to Cite this NIST Technical Series Publication

Temoshok D, Richer JP, Choong YY, Fenton JL, Lefkovitz N, Regenscheid A, Galluzzo R (2024) Digital Identity Guidelines: Federation and Assertions. (National Institute of Standards and Technology, Gaithersburg, MD), NIST Special Publication (SP) 800-63C-4 2pd. https://doi.org/10.6028/NIST.SP.800-63c-4.2pd

Author ORCID iDs

David Temoshok: 0000-0001-6195-0331
Justin P. Richer: 0000-0003-2130-5180
Yee-Yin Choong: 0000-0002-3889-6047
James L. Fenton: 0000-0002-2344-4291
Naomi Lefkovitz: 0000-0003-3777-3106
Andrew Regenscheid: 0000-0002-3930-527X

Public Comment Period

August 21, 2024 - October 7, 2024

Submit Comments

mailto:dig-comments@nist.gov

Additional Information

Additional information about this publication is available at https://csrc.nist.gov/pubs/sp/800/63/c/4/2pd, including related content, potential updates, and document history.

All comments are subject to release under the Freedom of Information Act (FOIA).

Abstract

This guideline focuses on the use of federated identity and the use of assertions to implement identity federations. Federation allows a given credential service provider to provide authentication attributes and (optionally) subscriber attributes to a number of separately-administered relying parties. Similarly, relying parties may use more than one credential service provider. The guidelines are not intended to constrain the development or use of standards outside of this purpose. This publication supersedes NIST Special Publication (SP) 800-63C.

Keywords

assertions; authentication; credential service provider; digital authentication; electronic authentication; electronic credentials; federations.

Reports on Computer Systems Technology

The Information Technology Laboratory (ITL) at the National Institute of Standards and Technology (NIST) promotes the U.S. economy and public welfare by providing technical leadership for the Nation’s measurement and standards infrastructure. ITL develops tests, test methods, reference data, proof of concept implementations, and technical analyses to advance the development and productive use of information technology. ITL’s responsibilities include the development of management, administrative, technical, and physical standards and guidelines for the cost-effective security and privacy of other than national security-related information in federal information systems. The Special Publication 800-series reports on ITL’s research, guidelines, and outreach efforts in information system security, and its collaborative activities with industry, government, and academic organizations.

Note to Reviewers

In December 2022, NIST released the Initial Public Draft (IPD) of SP 800-63, Revision 4. Over the course of a 119-day public comment period, the authors received exceptional feedback from a broad community of interested entities and individuals. The input from nearly 4,000 specific comments has helped advance the improvement of these Digital Identity Guidelines in a manner that supports NIST’s critical goals of providing foundational risk management processes and requirements that enable the implementation of secure, private, equitable, and accessible identity systems. Based on this initial wave of feedback, several substantive changes have been made across all of the volumes. These changes include but are not limited to the following:

1. Updated text and context setting for risk management. Specifically, the authors have modified the process defined in the IPD to include a context-setting step of

defining and understanding the online service that the organization is offering and intending to potentially protect with identity systems.

2. Added recommended continuous evaluation metrics. The continuous improvement section introduced by the IPD has been expanded to include a set of recommended metrics for holistically evaluating identity solution performance. These are recommended due to the complexities of data streams and variances in solution deployments.
3. Expanded fraud requirements and recommendations. Programmatic fraud management requirements for credential service providers and relying parties now address issues and challenges that may result from the implementation of fraud checks.
4. Restructured the identity proofing controls. There is a new taxonomy and structure for the requirements at each assurance level based on the means of providing the proofing: Remote Unattended, Remote Attended (e.g., video session), Onsite Unattended (e.g., kiosk), and Onsite Attended (e.g., in-person).
5. Integrated syncable authenticators. In April 2024, NIST published interim guidance for syncable authenticators. This guidance has been integrated into SP 800-63B as normative text and is provided for public feedback as part of the Revision 4 volume set.
6. Added user-controlled wallets to the federation model. Digital wallets and credentials (called “attribute bundles” in SP 800-63C) are seeing increased attention and adoption. At their core, they function like a federated IdP, generating signed assertions about a subject. Specific requirements for this presentation and the emerging context are presented in SP 800-63C-4.

The rapid proliferation of online services over the past few years has heightened the need for reliable, equitable, secure, and privacy-protective digital identity solutions. Revision 4 of NIST Special Publication SP 800-63, *Digital Identity Guidelines*, intends to respond to the changing digital landscape that has emerged since the last major revision of this suite was published in 2017, including the real-world implications of online risks. The guidelines present the process and technical requirements for meeting digital identity management assurance levels for identity proofing, authentication, and federation, including requirements for security and privacy as well as considerations for fostering equity and the usability of digital identity solutions and technology.

Based on the feedback provided in response to the June 2020 Pre-Draft Call for Comments, research into real-world implementations of the guidelines, market innovation, and the current threat environment, this draft seeks to:

- Address comments received in response to the IPD of Revision 4 of SP 800-63
- Clarify the text to address the questions and issues raised in the public comments

- Update all four volumes of SP 800-63 based on current technology and market developments, the changing digital identity threat landscape, and organizational needs for digital identity solutions to address online security, privacy, usability, and equity

NIST is specifically interested in comments and recommendations on the following topics:

1. Federation and Assertions

    - Is the concept of user-controlled wallets and attribute bundles sufficiently and clearly described to support real-world implementations? Are there additional requirements or considerations that should be added to improve the security, usability, and privacy of these technologies?

2. General

    - What specific implementation guidance, reference architectures, metrics, or other supporting resources could enable more rapid adoption and implementation of this and future iterations of the Digital Identity Guidelines?

    - What applied research and measurement efforts would provide the greatest impacts on the identity market and advancement of these guidelines?

Reviewers are encouraged to comment and suggest changes to the text of all four draft volumes of the SP 800-63-4 suite. NIST requests that all comments be submitted by 11:59pm Eastern Time on October 7th, 2024. Please submit your comments to dig-comments@nist.gov. NIST will review all comments and make them available on the NIST Identity and Access Management website. Commenters are encouraged to use the comment template provided on the NIST Computer Security Resource Center website for responses to these notes to reviewers and for specific comments on the text of the four-volume suite.

Call for Patent Claims

This public review includes a call for information on essential patent claims (claims whose use would be required for compliance with the guidance or requirements in this Information Technology Laboratory (ITL) draft publication). Such guidance and/or requirements may be directly stated in this ITL Publication or by reference to another publication. This call also includes disclosure, where known, of the existence of pending U.S. or foreign patent applications relating to this ITL draft publication and of any relevant unexpired U.S. or foreign patents.

ITL may require from the patent holder, or a party authorized to make assurances on its behalf, in written or electronic form, either:

a) assurance in the form of a general disclaimer to the effect that such party does not hold and does not currently intend holding any essential patent claim(s); or

b) assurance that a license to such essential patent claim(s) will be made available to applicants desiring to utilize the license for the purpose of complying with the guidance or requirements in this ITL draft publication either:

   i. under reasonable terms and conditions that are demonstrably free of any unfair discrimination; or

   ii. without compensation and under reasonable terms and conditions that are demonstrably free of any unfair discrimination.

Such assurance shall indicate that the patent holder (or third party authorized to make assurances on its behalf) will include in any documents transferring ownership of patents subject to the assurance, provisions sufficient to ensure that the commitments in the assurance are binding on the transferee, and that the transferee will similarly include appropriate provisions in the event of future transfers with the goal of binding each successor-in-interest.

The assurance shall also indicate that it is intended to be binding on successors-in-interest regardless of whether such provisions are included in the relevant transfer documents.

Such statements should be addressed to: mailto:dig-comments@nist.gov.

Table of Contents

List of Tables

List of Figures

Preface

This publication and its companion volumes, [SP800-63], [SP800-63A], and [SP800-63B], provide technical guidelines to organizations for the implementation of digital identity services.

This document, SP 800-63C, provides requirements to identity providers (IdPs) and relying parties (RPs) of federated identity systems. Federation allows a given IdP to provide authentication attributes and (optionally) subscriber attributes to a number of separately-administered RPs through the use of federation protocols and assertions. Similarly, RPs can use more than one IdP as sources of identities.

Acknowledgments

The authors would like to thank their fellow collaborators on the current revision of this special publication, Christine Abruzzi, Ryan Galluzzo, Sarbari Gupta, Connie LaSalle, and Diana Proud-Madruga, as well as Kerrianne Buchanan for her contributions and review. The authors would like to also acknowledge the past contributions of Donna F. Dodson, Elaine M. Newton, Ray A. Perlner, W. Timothy Polk, Emad A. Nabbus, Paul A. Grassi, Ellen M. Nadeau, Kristen K. Greene, Mary F. Theofanos, Sarah K. Squire, Jamie M. Danker, Kaitlin Boeckl, Kat Megas, Ben Piccarreta, and Danna Gabel O’Rourke.

1. Introduction

This section is informative.

Federation is a process that enables the subscriber account defined in [SP800-63A] to be used with an RP that does not verify one of the authenticators bound to the subscriber account. Instead, a service known as an identity provider, or IdP, makes the subscriber account available through a federation protocol to the relying party, or RP. The IdP sends a verifiable statement, called an assertion, about the subscriber account to the RP, triggered by an authentication event of the subscriber. The RP verifies the assertion provided by the IdP and creates an authenticated session with the subscriber, granting the subscriber access to the RP's functions.

The IdP works in one of two modes:

- As a verifier for authenticators bound to the subscriber account as described in [SP800-63B] (see details in Sec. 4), or
- As a subscriber-controlled device onboarded by the CSP, often known as a digital wallet (see details in Sec. 5).

The federation process allows the subscriber to obtain services from multiple RPs without the need to hold or maintain separate authenticators at each RP, a process sometimes known as *single sign-on*. The federation process also is generally the preferred approach to authentication when the RP and the subscriber account are not administered together under a common security domain, since the RP does not need to verify an authenticator in the subscriber account. Even so, federation can be still applied within a single security domain for a variety of benefits including centralized account management and technical integration.

The federation process can be facilitated by additional parties acting in other roles, such as a federation authority to facilitate the trust agreements in place and federation proxies to facilitate the protocol connections.

1.1. Notations

This guideline uses the following typographical conventions in text:

- Specific terms in **CAPITALS** represent normative requirements. When these same terms are not in **CAPITALS**, the term does not represent a normative requirement.
  - The terms " **SHALL** " and " **SHALL NOT** " indicate requirements to be followed strictly in order to conform to the publication and from which no deviation is permitted.
  - The terms " **SHOULD** " and " **SHOULD NOT** " indicate that among several possibilities, one is recommended as particularly suitable without mentioning

or excluding others, that a certain course of action is preferred but not necessarily required, or that (in the negative form) a certain possibility or course of action is discouraged but not prohibited.

- The terms “ **MAY** ” and “ **NEED NOT** ” indicate a course of action permissible within the limits of the publication.
- The terms “ **CAN** ” and “ **CANNOT** ” indicate a possibility and capability—whether material, physical, or causal—or, in the negative, the absence of that possibility or capability.

1.2. Document Structure

This document is organized as follows. Each section is labeled as either normative (i.e., mandatory for compliance) or informative (i.e., not mandatory).

- Section 1 provides an introduction to the document. This section is *informative*.
- Section 2 describes requirements for Federation Assurance Levels. This section is *normative*.
- Section 3 describes general requirements for federation systems. This section is *normative*.
- Section 4 describes requirements for general-purpose IdPs. This section is *normative*.
- Section 5 describes requirements for subscriber-controlled wallets. This section is *normative*.
- Section 6 provides security considerations. This section is *informative*.
- Section 7 provides privacy considerations. This section is *informative*.
- Section 8 provides usability considerations. This section is *informative*.
- Section 9 provides equity considerations. This section is *informative*.
- Section 10 provides additional example scenarios. This section is *informative*.
- References contains a list of publications referred to from this document. This section is *informative*.
- Appendix A contains a selected list of abbreviations used in this document. This appendix is *informative*.
- Appendix B contains a glossary of selected terms used in this document. This appendix is *informative*.
- Appendix C contains a summarized list of changes in this document’s history. This appendix is *informative*.

2. Federation Assurance Level (FAL)

This section is normative.

This section defines *federation assurance levels* (FALs) and the requirements for securing federation transactions at each FAL. In order to fulfill the requirements for a a given FAL, the federation transaction **SHALL** meet or exceed all requirements listed for that FAL.

Each FAL is characterized by a set of requirements that increase the security and complexity as the FAL increases. These requirements are listed here and expanded in other sections of this document:

Audience Restriction
The assertion presented in the federation protocol is targeted to a specific RP and the RP can confirm that it is the intended audience of the assertion.

Injection Protection
The RP is strongly protected from an attacker presenting an assertion in circumstances outside a current federation transaction request. (See Sec. 3.10.1 for details on injection protection.)

Trust Agreement Establishment
The agreement to participate in a federation transaction for the purposes of creating an authenticated session for the subscriber at the RP. (See Sec. 3.4 for details of the trust agreement.)

Identifier and Key Establishment
The IdP and RP have exchanged identifiers and key material to allow for the verification of assertions and other artifacts during future federation transactions. (See Sec. 3.5 for details of key establishment.)

Presentation
The assertion can be presented to the RP either on its own (as a bearer assertion) or in concert with an authenticator presented by the subscriber.

Table 1 provides a non-normative summary of aspects for each FAL. Each successive level subsumes and fulfills all requirements of lower levels (e.g., a federation process at FAL3 can be accepted at FAL2 or FAL1 since FAL3 satisfies all the requirements of these lower levels). Combinations not found in Table 1 are possible, and agencies can choose to implement stronger protections in one or more areas of requirements at a given FAL.

Table 1. Federation Assurance Levels

Requirement	**FAL1**	**FAL2**	**FAL3**
Audience Restriction	Multiple RPs allowed per assertion, Single RP per assertion recommended	Single RP per assertion	Single RP per assertion
Injection Protection	Recommended for all transactions	Required; transaction begins at the RP	Required; transaction begins at the RP
Trust Agreement Establishment	Subscriber-driven or A priori	A priori	A priori
Identifier and Key Establishment	Dynamic or Static	Dynamic or Static	Static
Presentation	Bearer Assertion	Bearer Assertion	Holder-of-Key Assertion or Bound Authenticator

While many different federation implementation options are possible, the FAL is intended to provide clear guidance representing increasingly secure deployment options. See [SP800-63] for details on how to choose the most appropriate FAL.

> Note: In these guidelines, assertions, attribute bundles, and other elements of the federation protocol are protected by asymmetric digital signatures or symmetric MACs. When either asymmetric or symmetric cryptography is specifically required, the terms “sign” and “signature” will be qualified as appropriate to indicate the requirement. When either option is possible, the terms “sign” and “signature” are used without a qualifier.

2.1. Common FAL Requirements

At all FALs, all federation transactions **SHALL** comply with the requirements in Sec. 3 to deliver an assertion to the RP and create an authenticated session at the RP. Examples of assertions used in federation protocols include the ID Token in OpenID Connect [OIDC] and the Security Assertion Markup Language [SAML] Assertion format.

At all FALs, the RP needs to trust the IdP to provide valid assertions representing the subscriber’s authentication event and **SHALL** validate the assertion.

IdPs and RPs **SHALL** employ appropriately tailored security controls from the moderate baseline security controls defined in [SP800-53] or an equivalent federal (e.g., [FEDRAMP]) or industry standard that the organization has determined for the information systems, applications, and online services that these guidelines are used to protect. IdPs and RPs **SHALL** ensure that the minimum assurance-related controls for the appropriate systems, or equivalent, are satisfied. Additional security controls are discussed in Sec. 3.10.

If no FAL is specified by the trust agreement or federation transaction, the requirements of this section still apply.

An IdP or RP can be capable of operating at multiple FALs simultaneously, depending on use case and needs. For example, an IdP could provide FAL3 federation transactions to a high-risk RP while providing FAL2 to an RP with a lower risk profile. Similarly, an RP could require FAL2 for normal actions but require the subscriber to re-authenticate with FAL3 for higher impact or more sensitive actions. This capability extends to other dimensions, as an IdP could simultaneously have access to subscriber accounts that have been proofed at any IAL and allow authentication at any AAL. However, an RP talking to that IdP could have restrictions on the lowest IAL and AAL it is willing to accept for access. As a consequence, it is imperative that the trust agreement establish the xALs allowed and required for different use cases.

2.2. Federation Assurance Level 1 (FAL1)

FAL1 provides a basic level of protection for federation transactions, allowing for a wide range of use cases and deployment decisions.

At FAL1, the IdP **SHALL** sign the assertion using approved cryptography. The RP **SHALL** validate the signature using the key associated with the expected IdP. The signature protects the integrity of the assertion contents and allows for the IdP to be verified as the source of the assertion.

All assertions at FAL1 **SHALL** be audience-restricted to a specific RP or set of RPs, and the RP **SHALL** validate that it is one of the targeted RPs for the given assertion.

At FAL1, the trust agreement **MAY** be established by the subscriber during the federation transaction. Note that at FAL1, it is still possible for the trust agreement to be established a priori by the RP and IdP.

At FAL1, the federation protocol **SHOULD** apply injection protection as discussed in Sec. 3.10.1. The federation transaction **SHOULD** be initiated by the RP.

2.3. Federation Assurance Level 2 (FAL2)

FAL2 provides a high level of protection for federation transactions, providing protections against a variety of attacks against federated systems. All the requirements for FAL1 apply at FAL2 except where overridden by more specific or stringent requirements here.

At FAL2, the assertion **SHALL** be strongly protected from injection attacks, as discussed in Sec. 3.10.1. The federation transaction **SHALL** be initiated by the RP.

At FAL2, the assertion **SHALL** audience restricted to a single RP.

At FAL2, an a priori trust agreement **SHALL** be established prior to the federation transaction taking place.

IdPs operated by or on behalf of federal agencies that present assertions at FAL2 or higher **SHALL** protect keys used for signing or encrypting those assertions with mechanisms validated at [FIPS140] Level 1 or higher.

2.4. Federation Assurance Level 3 (FAL3)

FAL3 provides a very high level of protection for federation transactions, establishing very high confidence that the subscriber asserted by the IdP is the subscriber present in the authenticated session. All the requirements at FAL1 and FAL2 apply at FAL3 except where overridden by more specific or stringent requirements here.

At FAL3, the RP **SHALL** verify that the subscriber is in control of an authenticator in addition to the assertion. This authenticator is either identified in a holder-of-key assertion as described in Sec 3.14 or is a bound authenticator as described in Sec. 3.15.

At FAL3, the trust agreement **SHALL** be established such that the IdP can identify and trust the RP to abide by all aspects of the trust agreement prior to any federation transaction taking place. To facilitate this, the key material used to authenticate the RP and IdP to each other is associated with the identifiers for the RP and IdP in a static fashion using a trusted mechanism. For example, a public key file representing the RP is uploaded to the IdP during a static registration process, and the RP downloads the IdP’s public key from a URL indicated in the trust agreement. Alternatively, the trust agreement can dictate that the RP and IdP can upload their respective public keys to a federation authority and then download each other’s keys from that same trusted authority.

IdPs operated by or on behalf of federal agencies that present assertions at FAL3 **SHALL** protect keys used for signing or encrypting those assertions with mechanisms validated at [FIPS140] Level 1 or higher.

2.5. Requesting and Processing xALs

Since an IdP is capable of asserting the identities of many different subscribers with a variety of authenticators using a variety of federation parameters, the IAL, AAL, and FAL could vary across different federation transactions, even to the same RP.

IdPs **SHALL** support a mechanism for RPs to specify a set of minimum acceptable xALs as part of the trust agreement and **SHOULD** support the RP specifying a more strict minimum set at runtime as part of the federation transaction. When an RP requests a particular xAL, the IdP **SHOULD** fulfill that request, if possible, and **SHALL** indicate the resulting xAL in the assertion. For example, if the subscriber has an active session that was authenticated at AAL1, but the RP has requested AAL2, the IdP needs to prompt the subscriber for AAL2 authentication to step up the security of the session at the IdP during the subscriber's interaction at the IdP, if possible. The IdP sends the resulting AAL as part of the returned assertion, whether it is AAL1 (the step-up authentication was not met) or AAL2 (the step-up authentication was met successfully).

The IdP **SHALL** inform the RP of the following information for each federation transaction:

- The IAL of the subscriber account being presented to the RP, or an indication that no IAL claim is being made
- The AAL of the currently active session of the subscriber at the IdP, or an indication that no AAL claim is being made
- The FAL of the federation transaction

The RP gets this xAL information from a combination of the terms of the trust agreement as described in Sec. 3.4 and information included in the assertion as described in Sec. 4.9 and Sec. 5.8. If the xAL is unchanging for all messages between the IdP and RP, the xAL information **SHALL** be included in the terms of the trust agreement between the IdP and RP. If the xAL could be within a range of possible values specified by the trust agreement, the xAL information **SHALL** be included as part of the assertion contents.

The IdP **MAY** indicate that no claim is made to the IAL or AAL for a given federation transaction. In such cases, no default value is assigned to the resulting xAL by the RP. That is to say, a federation transaction without an IAL declaration in either the trust agreement or the assertion is functionally considered to have "no IAL" and the RP cannot assume the account meets "IAL1", the lowest numbered IAL described in this suite.

The RP **SHALL** determine the minimum IAL, AAL, and FAL it is willing to accept for access to any offered functionality. An RP **MAY** vary its functionality based on the IAL, AAL, and FAL of a specific federated authentication. For example, an RP can allow federation transactions at AAL2 for common functionality (e.g., viewing the status of a dam system) but require AAL3 be used for higher risk functionality (e.g., changing the flow rates of a dam system). Similarly, an RP could restrict management functionality to only certain subscriber accounts which have been identity proofed at IAL2, while allowing federation transactions from all subscriber accounts regardless of IAL.

In a federation process, only the IdP has direct access to the details of the subscriber account, which determines the applicable IAL, and the authentication event at the IdP, which determines the applicable AAL. Consequently, the IdP declares the IAL, AAL, and intended FAL for each federation transaction.

The RP **SHALL** ensure that it meets its obligations in the federation transaction for the FAL declared in the assertion. For example, the RP needs to ensure the presentation method meets the injection protection requirements at FAL2 and above, and that the appropriate bound authenticator is presented at FAL3.

3. Common Federation Requirements

This section is normative.

A federation transaction serves to allow the subscriber to establish an authenticated session with the RP based on a subscriber account known to the IdP. The federation transaction can also provide the RP with a set of identity attributes within the authenticated session. The authenticated session can then be used by the RP for:

- logging in the subscriber to access functionality at the RP,
- identifying the subscriber based on presented attributes, and
- processing the subscriber attributes presented in the federation transaction.

A federation transaction requires relatively complex multiparty protocols that have subtle security and privacy requirements. When evaluating a particular federation protocol, profile, or deployment structure, it is often instructive to break it down into its component relationships and evaluate the needs for each of these:

- the subscriber to the CSP,
- the CSP to the IdP,
- the subscriber to the IdP,
- the IdP to the RP, and
- the subscriber to the RP.

In addition, the subscriber often interacts with the CSP, IdP, and RP through a user agent like a web browser. The user agent is therefore often involved in the federation process, but it is not necessary for all types of applications and interactions. As such, the actions of the subscriber described throughout these guidelines can optionally be performed through a user agent. Where necessary, requirements on the user agent are called out directly.

Each party in a federation protocol bears specific responsibilities and expectations that must be fulfilled in order for the federated system to function as intended.

The subscriber account is augmented by the IdP with federation-specific items, including but not limited to the following:

- One or more external subject identifiers, for use with a federation protocol
- A set of access rights, detailing which RPs can access which attributes of the subscriber account (such as allowlists and saved runtime decisions by the subscriber)
- Federated account usage information
- Additional attributes collected by or assigned by the IdP to the account

A subset of these attributes is made available to the RP through the federation process, either in the assertion or through an identity API (see Sec 3.11.3). These attributes are often used in determining access privileges for attribute-based access control (ABAC) or facilitating a transaction (e.g., providing a shipping address). The details of authorization and access control are outside the scope of these guidelines.

To keep and manage these attributes, the RP often maintains an *RP subscriber account* for the subscriber. The RP subscriber account also contains information local to the RP itself, as described in Sec. 3.7.

Federation transactions take place across three dimensions:

Trust Agreements:
The establishment of a policy decision that allows the CSP, IdP, and RP to connect for the purposes of federation. This policy is governed by a trust agreement, which establishes the permission to connect.

Associating Keys and Identifiers:
The association of keys and identifiers for the CSP, IdP, and RP that take part in the federation transaction. This process enables the parties to identify each other securely for future exchanges.

Federation Protocol:
The verification of the subscriber’s identity by the IdP and subsequent issuance of an assertion to the RP. This results in the passing of subscriber attributes to the RP and establishing an authenticated session for the subscriber at the RP.

These dimensions all need to be fulfilled for a federation process to be complete. The exact order in which that happens, and which parties are involved in which steps, can vary depending on deployment models and other factors.

The requirements for IdPs in this section apply to both general-purpose IdPs as discussed in Sec. 4 and subscriber-controlled wallets as discussed in Sec. 5.

3.1. Roles

3.1.1. Credential Service Provider (CSP)

The CSP collects and verifies attributes from the subscriber and stores them in a subscriber account. The CSP also binds one or more authenticators to the subscriber account, allowing the subscriber to authenticate directly to systems capable of verifying an authenticator.

3.1.2. Identity Provider (IdP)

The IdP provides a bridge between the subscriber account (as established by the CSP) and the RP that the subscriber is accessing. An IdP can be deployed as a service for multiple subscriber accounts or as a component controlled by a single subscriber.

The IdP establishes an authentication event with the subscriber, either through the verification of an authenticator (for general-purpose IdPs) or presentation of an activation factor (for subscriber-controlled wallets). The IdP creates assertions to represent the authentication event.

The IdP makes identity attributes of the subscriber available within the assertion or through an identity API (see Sec. 3.11.3).

In some systems, this is also known as the offering party (OP).

3.1.3. Relying Party (RP)

The RP processes assertions from the IdP and provides the service that the subscriber is trying to access. Unlike in a direct authentication model, the RP does not provide the verifier function to authenticators tied to the subscriber account.

In some systems, this is also known as the service provider (SP).

3.2. Functions

3.2.1. Trust Agreement Management

The trust agreement (see Sec. 3.4) can be managed through a dedicated party, known as a *federation authority*. The federation authority facilitates the onboarding and management of parties fulfilling different roles and functions within a trust agreement. This management provides a transitive trust to other parties in the agreement.

For example, an RP can enter a trust agreement with a federation authority and decide that any IdP approved by that federation authority is suitable for its purposes. This trust can hold true whether or not the IdP was covered by the trust agreement at the time the RP joined. Federation authorities are used in multilateral trust agreements as discussed in Sec. 3.4.2.

3.2.2. Authorized Party

The *authorized party* in a trust agreement is the organization, person, or entity that is responsible for the specific release decisions covered by the trust agreement, including the release of subscriber attributes. The trust agreement stipulates who the expected authorized party is, as well as the parameters under which a request could be automatically granted, automatically denied, or require a runtime decision from an individual. For public-facing scenarios, the authorized party is expected to be the subscriber. For enterprise scenarios, the authorized party is expected to be the agency.

If the authorized party is the operator of the IdP, consent to release attributes is decided for all subscribers and established by an allowlist as described in Sec. 4.6.1.1, allowing for the disclosure of identity attribute without direct decisions and involvement by the subscriber. A trust agreement can alternatively stipulate that an individual, such as the subscriber, is to be prompted at runtime for consent to disclose certain attributes to the RP as discussed in Sec. 4.6.1.3. If specified by the trust agreement, it is also possible for an individual other than a subscriber to act as the authorized party. For example, an administrator of a system being prompted to release attribute information on behalf of a subscriber as part of a provisioning API.

Examples of different authorized parties are found in Sec 10.10.

3.2.3. Proxied Federation

A federation *proxy* acts as an intermediary between the IdP and RP for all communication in the federation protocol. The proxy functions as an RP on the upstream side and an IdP on the downstream side, as shown in Fig. 1. When communicating through a proxy, the upstream IdP and downstream RP communicate with the proxy using a standard federation protocol, and the subscriber takes part in two separate federation transactions. As a consequence, all normative requirements that apply to IdPs and RPs **SHALL** apply to proxies in their respective roles on each side. Additionally, it is possible for a proxy to act as an upstream IdP to another proxy downstream, and so on in a chain.

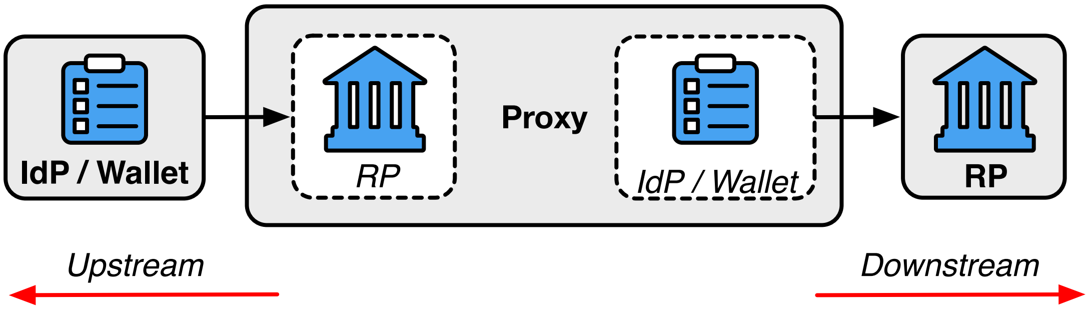

Fig. 1. Federation Proxy

The role of the proxy is limited to the federation protocol; it is not involved in establishment or facilitation of a trust agreement between the upstream IdP and downstream RP. The same party can operate a federation authority as well as a proxy to facilitate federation transactions, but this function is separate from their role in managing the trust agreement. Just like other members of a federation system, the proxy can be involved in separate trust agreements with each of the upstream and downstream components, or a single trust agreement can apply to all parties such as in a multilateral agreement.

The federated identifier (see Sec. 3.3) of an assertion from a proxy **SHALL** indicate the proxy as the issuer of the assertion. The downstream RP receives and validates the assertion generated by the proxy, as it would an assertion from any other IdP. This assertion is based on the assertion the proxy receives from the upstream IdP. The contents of the assertion from the upstream IdP can be handled in several ways, depending on the method of proxying in use:

- The proxy can create an all-new assertion with no information from the assertion from the upstream IdP carried in it. This pattern is useful for blinding the downstream RP, so that the RP does not know which upstream IdP the subscriber originally came from.
- The proxy can copy attributes from the assertion from the upstream IdP into the assertion from the proxy. This pattern is useful for carrying identity attributes in the assertion to the downstream RP.
- The proxy can include the entire assertion from the upstream IdP in the assertion from the proxy. This pattern allows the RP to independently validate the assertion from the upstream IdP as well as the assertion from the proxy.

A proxied federation model can provide several benefits. Federation proxies can simplify technical integration between the RP and IdP by providing a common interface for integration. Additionally, to the extent a proxy effectively blinds the RP and IdP from each other, it can provide some business confidentiality for organizations that want to guard their subscriber lists from each other. Proxies can also mitigate some of the privacy risks described in Sec. 3.9, though other risks arise from their use since an additional party is now involved in handling subscriber information. For example, if an attacker is able to compromise the proxy, the attacker need not target the IdP or RP directly in order to gain access to subscriber attributes or activity since all of that information flows through the proxy. Additionally, the proxy can perform additional profiling of the subscriber beyond what the IdP and RP can do, since the proxy brokers the federation transactions between the parties and binds the subscriber account to either side of the connection.

See Sec. 7.5 for further information on blinding techniques, their uses, and limitations.

The FAL of the connection between the proxy and the downstream RP is considered as the lowest FAL along the entire path, and the proxy **SHALL** accurately represent this to the downstream RP. For example, if the connection between the upstream IdP and the proxy is FAL1 and the connection between the proxy and the downstream RP otherwise meets the requirements of FAL2, the connection between the proxy and the downstream RP is still considered FAL1. Likewise, if the connection between the upstream IdP and the proxy is FAL2 and the connection between the proxy and the downstream RP is only FAL1, the overall connection through the proxy is considered FAL1.

3.2.4. Fulfilling Roles and Functions of a Federation Model

The roles in a federation transaction can be connected in a variety of ways, but several common patterns are anticipated by these guidelines. The expected trust agreement structure and connection between components will vary based on which pattern is in use.

Different roles and functions can be fulfilled by separate parties who integrate with each other. For example, a CSP can provide attributes of the subscriber account to an IdP that is not operated by the same party or agency as the CSP.

It is also possible for a single party to fulfil multiple roles within a given federation agreement. For example, if the CSP provides the IdP as part of its identity services, the CSP can provision the subscriber accounts at the IdP as part of the subscriber account establishment process. Similarly, the RP can also be in the same security and administrative domain as the IdP, but still use federation technology to connect for technical, deployment, and account management benefits.

The same is true for other functions in the overall federation system, such as a federation authority and proxy. While the roles may seem similar, they are fundamentally distinct and do not need to be connected: a federation authority facilitates establishment of a trust agreement between parties, and a proxy facilitates connection of the federation protocol by acting as an RP to the upstream IdP and as an IdP to the downstream RP. The same entity can fulfill both the federation authority and proxy functions in the system, providing both a means of establishing trust agreements and a means of establishing technical connections between IdPs and RPs.

3.3. Federated Identifiers

The subscriber **SHALL** be identified in the federation transaction using a federated identifier unique to that subscriber. A federated identifier is the logical combination of a subject identifier, representing a subscriber account, and an issuer identifier, representing the IdP. The subject identifier is assigned by the IdP, and the issuer identifier is assigned to the IdP usually through configuration.

The multi-part federated identifier pattern is required because different IdPs manage their subject identifiers independently, and could therefore potentially collide in their choices of subject identifiers for different subjects. Therefore, it is imperative that an RP never process the subject identifier without taking into account which IdP issued that subject identifier. For most use cases, the federated identifier is stable for the subscriber across multiple sessions and is independent of the authenticator used, allowing the RP to reliably identify the subscriber across multiple authenticated sessions and account changes. However, it is also possible for the federated identifier and its associated use at the RP to be ephemeral, providing some privacy enhancement. Federated identifiers, and their constituent parts, are intended to be machine-readable and not managed by or exposed to the subscriber, unlike a username or other human-facing identifier.

Federated identifiers **SHALL** contain no plaintext personally-identifiable information (PII), such as usernames, email addresses, or employee numbers, etc.

3.3.1. Pairwise Pseudonymous Identifiers (PPI)

In some circumstances, it is desirable to prevent the subscriber account from being easily linked at multiple RPs through use of a common subject identifier. The use of a pairwise pseudonymous identifier (PPI) allows an IdP to provide multiple distinct federated identifiers to different RPs for a single subscriber account. Use of a PPI prevents different RPs from colluding together to track the subscriber using the federated identifier.

3.3.1.1. General Requirements

When using pairwise pseudonymous identifiers within the assertions generated by the IdP for the RP, the IdP **SHALL** generate a different federated identifier for each RP as described in Sec. 3.3.1.2 or set of RPs as described in Sec. 3.3.1.3.

Some identity attributes such as names, physical address, phone numbers, email addresses, and others can be used to identify a subscriber outside of a federation transaction. When PPIs are used alongside these kinds of identifying attributes, it may still be possible for multiple colluding RPs to re-identify a subscriber by correlation across systems. For example, if two independent RPs each see the same subscriber identified with a different PPI, the RPs could still determine that the subscriber is the same person by comparing the name, email address, physical address, or other identifying attributes carried alongside the PPI in the respective assertions. Where PPIs are used alongside identifying attributes, privacy policies **SHALL** be established to prevent correlation of subscriber data consistent with applicable legal and regulatory requirements.

Note that in a proxied federation model (see Sec. 3.2.3), the upstream IdP may be unable to generate a PPI for the downstream RP, since the proxy could blind the IdP from knowing which RP is being accessed by the subscriber. In such situations, the PPI is generally established between the IdP and the federation proxy. The proxy, acting as an IdP, can provide a PPI to the downstream RP. Depending on the protocol, the federation proxy may need to map the PPI back to the associated identifiers from upstream IdPs in order to allow the identity protocol to function. In such cases, the proxy will be able to track and determine which PPIs represent the same subscriber at different RPs. The proxy **SHALL NOT** disclose the mapping between the PPI and any other identifiers to a third party or use the information for any purpose other than those allowed for transmission of subscriber information defined in Sec. 3.9.1.

3.3.1.2. Pairwise Pseudonymous Identifier Generation

The PPI **SHALL** contain no identifying information about the subscriber (e.g., username, email address, employee number, etc.). The PPI **SHALL** be difficult to guess by a party having access to information about the subscriber, having at least 112 bits of entropy as stated in [SP800-131A]. PPIs can be generated randomly and assigned to subscribers by the IdP or could be derived from other subscriber information if the derivation is done in an irreversible, unguessable manner (e.g., using a keyed hash function with a secret key as discussed in [SP800-131A]).

Unless the PPI is designated as shared by the trust agreement, the PPI **SHALL** be disclosed to only a single RP.

3.3.1.3. Shared Pairwise Pseudonymous Identifiers

The same shared PPI **SHALL** be used for a specific set of RPs if all the following criteria are met:

- The trust agreement stipulates a shared PPI for a specific set of RPs;
- The authorized party consents to and is notified of the use of a shared PPI;
- Those RPs have a demonstrable relationship that justifies an operational need for the correlation, such as a shared security domain or shared legal ownership; and
- All RPs in the set of a shared PPI consent to being correlated in such a manner (i.e., one RP cannot request to have another RP's PPI without that other RP's knowledge and consent).

The RPs **SHALL** conduct a privacy risk assessment to consider the privacy risks associated with requesting a shared PPI. See Sec. 7.2 for further privacy considerations.

The IdP **SHALL** ensure that only intended RPs are included in the set; otherwise, a rogue RP could learn of the shared PPI for a set of RPs by fraudulently posing as part of that set.

The sector identifier feature of [OIDC] provides a mechanism to calculate a shared PPI for a group of RPs. In this protocol, the identifiers of the RPs are all listed at a URL that can be fetched by the IdP over an authorized protected channel. The shared PPI is calculated by taking into account the sector identifier URL along with other inputs to the algorithm, such that all RPs listed in the sector identifier URL's contents receive the same shared PPI.

3.4. Trust Agreements

All federation transactions **SHALL** be defined by one or more trust agreements between the applicable parties.

The trust agreement **SHALL** establish a trust relationship between the RP and:

- The CSP responsible for provisioning and managing the subscriber account,
- The IdP responsible for providing assertions and attributes, or
- Both the CSP and IdP.

Trust agreements establish the terms for federation transactions between the parties they affect, including things like the allowed xALs and the intended purposes of identity attributes exchanged in the federation transaction. The trust agreement **SHALL** establish usability and equity requirements for the federation transaction. The trust agreement **SHALL** disclose details of the proofing process used at the CSP, including any compensating controls and exception handling processes.

All trust agreements **SHALL** define a specific population of subscriber accounts that the agreement is applicable to. The exact means of defining this population are out of scope of this document. In many cases, the population is defined as the full set of subscriber accounts that the CSP manages and makes available through an IdP. In other cases, the population is a demarcated subset of accounts available through an IdP. It is also possible for an RP to have a distinct trust agreement established with an IdP for a single subscriber account, such as in a subscriber-driven trust agreement.

During the course of a single federation transaction, it is important for the policies and expectations of all parties be unambiguous for all parties involved. Therefore, there **SHOULD** be only one set of trust agreements in effect for a given transaction. This will usually be determined by the unique combination of CSP, IdP, and RP participating in the transaction. However, these agreements could vary in other ways, such as different populations of subscribers being governed by different trust agreements.

The existence of a trust agreement between parties does not preclude the existence of other agreements for each party in the agreement to have with other parties. For example, an IdP can have independent agreements with multiple RPs simultaneously, and an RP can likewise have independent agreements with multiple IdPs simultaneously. The IdP and RP need not disclose the existence or terms of trust agreements to parties outside of or not covered by the agreement in question.

Trust agreements **SHALL** establish terms regarding expected and acceptable IALs and AALs in connection with the federated relationship.

Trust agreements **SHALL** define necessary mechanisms and materials to coordinate redress and issues between the different participants in the federation, as discussed in Sec. 3.4.3.

Establishment of a trust agreement is required for all federation transactions, even those in which the roles and applications exist within a single security domain or shared legal ownership. In such cases, the establishment of the trust agreement can be an internal process and does not need to involve a formal agreement. Even in such cases, it is still required for the IdP to document and disclose the trust agreement to the subscriber upon request.

Even though subscribers are not generally a party directly involved in the trust agreement’s terms, subscribers are affected by the terms of the trust agreement and the resulting federation transactions. As such, the terms of the trust agreement need to be made available to subscribers in clear and understandable language. The means by which the subscriber can access these terms, and the party responsible for informing the subscriber, varies based on the means of establishment of the trust agreement and the terms of the trust agreement itself. Additionally, the subscriber's user agent is not usually party to the trust agreement, unless it is acting in one of the roles of the federation transaction.

3.4.1. Bilateral Trust Agreements

In a bilateral trust agreement, the establishment of the trust agreement occurs directly between the federated parties, and the trust agreement is not managed or facilitated by a separate party. Bilateral trust agreements allow for a point-to-point connection to be established between organizations wishing to provide federated identity access to services. Bilateral connections can take many forms, including large enterprise applications with static contracts and subscriber-driven dynamic connections to previously unknown RPs. In all cases, the CSP, IdP, and RP manage their policies regarding the federated connection directly.

Bilateral trust agreements impose no additional requirements beyond those needed to establish the trust agreement itself.

3.4.2. Multilateral Trust Agreements

In a multilateral trust agreement, the federated parties look to a *federation authority* to assist in establishing the trust agreement between parties. In this model, the federation authority facilitates the inclusion of CSPs, IdPs, and RPs under the trust agreement.

When onboarding a party in any role, the federation authority conducts vetting on that party to verify its compliance with the tenets of the trust agreement. The level of vetting is unique to the use cases and models employed within the federation, and details are outside the scope of this document. This vetting is depicted in Fig. 2.

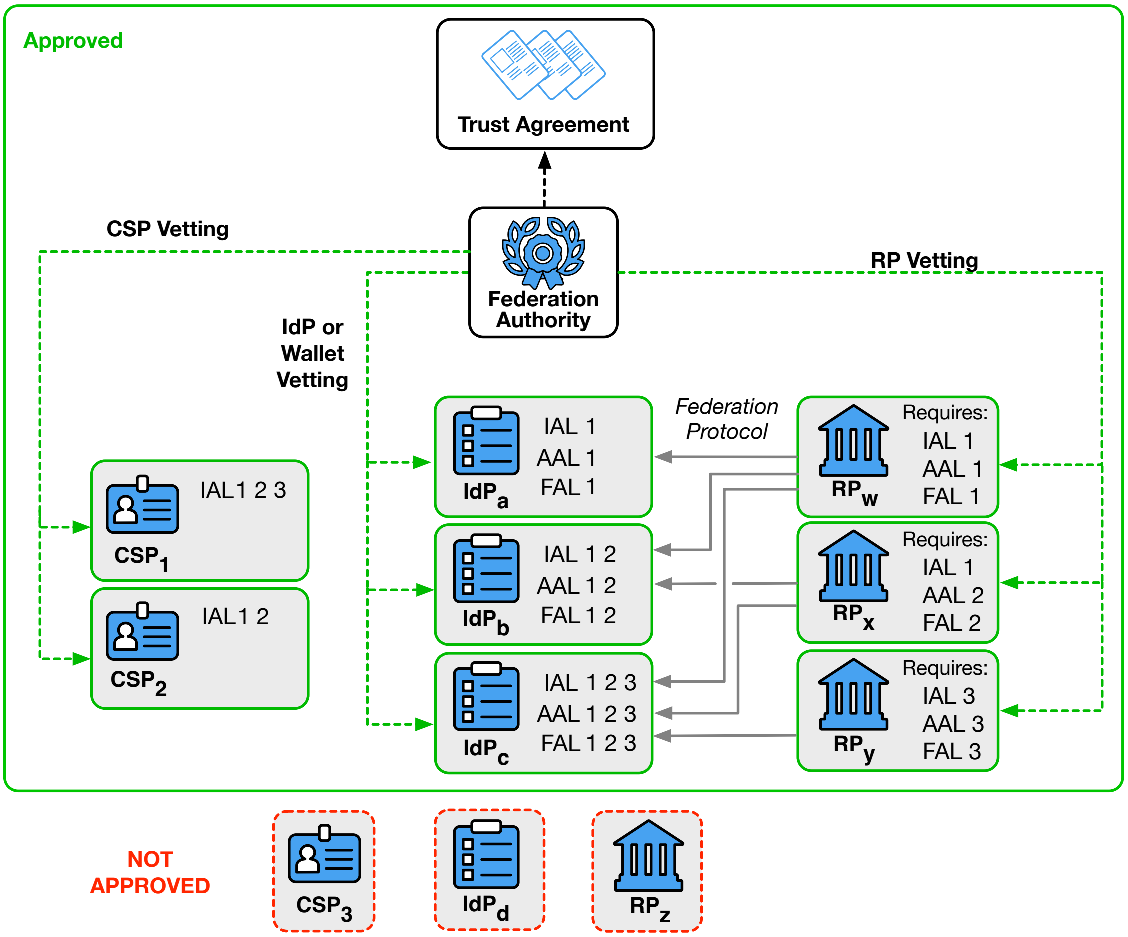

Fig. 2. Federation Authority

The trust agreement **SHALL** enumerate the required practices for vetting all parties, and **SHALL** indicate the party or parties responsible for performing the vetting process.

Vetting of CSPs, IdPs, and RPs **SHALL** establish, as a minimum, that:

- CSPs are performing identity proofing of subscriber accounts in accordance with [SP800-63A]

- CSPs onboard subscriber accounts to IdPs in a secure fashion in adherence to the requirements in Sec. 4.1 or Sec. 5.4 as applicable
- Authenticators used for authenticating the subscriber at the IdP or onboarding a subscriber-controlled wallet are used in accordance with [SP800-63B]
- Assertions generated by IdPs adhere to the requirements in Sec. 4.9 or Sec. 5.8.
- RPs adhere to requirements for handling subscriber attribute data, such as retention, aggregation, and disclosure to third parties.
- RP and IdP systems use approved profiles of federation protocols.

The federation authority **MAY** provide a programmatic means for parties under the trust agreement to verify membership of other parties under the trust agreement. For example, a federation authority could provide a discovery API that provides the vetted capabilities of an IdP for providing identities to RPs within the system, or it could provide a signed attestation for RPs to present to IdPs during a registration step.

Federation authorities **SHALL** periodically re-evaluate members for compliance, in terms disclosed in the trust agreement.

When information needs to be shared between CSPs, such as during suspicion of fraud on a subscriber account, the federation authority can define the policies that apply for the transfer of this information. While sharing information in this way can be used to mitigate fraud, there are also substantial privacy concerns. The federation authority **SHALL** include all information sharing between parties other than for identity purposes in its privacy risk assessment.

A federation authority **MAY** incorporate other multilateral trust agreements managed by other federation authorities in its trust agreement, creating an interfederation agreement. For example, IdP1 has been vetted under a multilateral agreement with FA1, and RP2 has been vetted under a multilateral agreement with FA2. In order to facilitate connection between IdP1 and RP2, a new federation authority FA3 can provide a multilateral agreement that accepts IdPs from FA1 and RPs from FA2. If IdP1 and RP2 accept the authority of FA3, the federation connection can continue under the auspices of this interfederation agreement.

3.4.3. Redress Requirements

Federation transactions occur between multiple parties that are often controlled by multiple entities, and different stages of the federation transaction can lead to different situations in which a subscriber would need to seek redress from the different parties.

As the recipient of a subscriber’s identity attributes, the RP is the subscriber’s primary view into the federated system, and in some instances the subscriber may be unaware that an IdP is involved with their use of the RP. Therefore it falls to the RP to provide the subscriber with a clear and accessible method of contacting the RP to request redress. For matters that involve the RP subscriber account (including any attributes stored in the account), RP functionality, bound authenticators, RP allowlists, and other items under the RP's control, the RP **SHALL** provide clear and accessible means of redress to the subscriber. For matters that involve the IdP or CSP, the RP **SHALL** provide the subscriber with a means of initiating the redress process with the IdP or CSP, as appropriate.

For matters involving the use of the subscriber account in federation transactions, including attribute values and derived attribute values made available over federation transactions, IdP functionality, holder-of-key authenticators, IdP allowlists, and other items in the IdP's control, the IdP **SHALL** provide clear and accessible means of redress to the subscriber. For matters that also involve a particular RP, the IdP **SHALL** provide the subscriber with a means of initiating the redress process with the RP. For matters involving the subscriber account that has been made available to the IdP, the IdP **SHALL** provide the subscriber with a means of initiating the redress process with the CSP.

For matters involving the subscriber account, including identity attributes and authenticators in the subscriber account, the CSP **SHALL** provide the subscriber with a clear and accessible means of redress.

See Sec. 3.6 of [SP800-63] for more requirements on providing redress.

3.5. Identifiers and Cryptographic Key Management for CSPs, IdPs, and RPs

While a trust agreement establishes permission to federate, it does not facilitate the secure connection of parties in the federation. In order to communicate over a federation protocol, the CSP, IdP, and RP need to be able to identify each other in a secure fashion, with the ability to associate identifiers with cryptographic keys and related security artifacts. In this way, an RP can ensure that an assertion is coming from the intended IdP, or that an attribute bundle is coming from the intended CSP. Likewise, an IdP can ensure that it is sending an assertion to the intended RP.

The process of an RP establishing cryptographic keys and identifiers for an IdP or CSP is known as *discovery*. The process of the IdP establishing cryptographic keys and identifiers for the RP is known as *registration*. Both the discovery and registration processes can happen prior to any federation transaction happening, or inline as part of the transaction itself. Both the discovery and registration processes can happen directly between parties or be facilitated through use of a third party service. Different federation protocols and processes have different processes for establishing these cryptographic keys and identifiers, but the end result is that each party can properly identify others as necessary within the protocol.

The discovery and registration processes **SHALL** be established in a secure fashion as defined by the trust agreement governing the transaction. Protocols requiring the transfer of cryptographic key information **SHALL** use an authenticated protected channel to exchange cryptographic key information needed to operate the federated relationship, including any shared secrets or public keys. Any symmetric keys used in this relationship **SHALL** be unique to a pair of federation participants.

CSPs, IdPs (including subscriber-controlled wallets), and RPs **MAY** have multiple cryptographic keys and identifiers to serve different purposes within a trust agreement, or to serve different trust agreements. For example, an IdP could use one set of assertion signing keys for all FAL1 and FAL2 transactions, but use a separately managed set of cryptographic keys for FAL3 transactions, stored in a higher security container.

When domain names, URIs, or other structured identifiers are used to identify parties, wildcards **SHALL NOT** be used. For example, if an RP is deployed at "www.example.com", "service.example.com", and "gateway.example.com", then each of these identifiers would have to be registered for the RP. A wildcard of "*.example.com" cannot be used, as it would unintentionally allow access to "user.example.com" and "unknown.example.com" under the same RP identifier.

3.5.1. Cryptographic Key Rotation

Over time, it can be desirable or necessary to update the cryptographic key associated with a CSP, IdP, or RP. The allowable update process for any cryptographic keys and identifiers **SHALL** be defined by the trust agreement and **SHALL** be executed using an authenticated protected channel, as in the initial cryptographic key establishment.

For example, if the IdP is identified by a URL, the IdP could publish its current public key set at a location underneath that URL. RPs can then fetch the public key from the known location as needed, getting updated public keys as they are made available.

3.5.2. Cryptographic Key Storage

CSPs, IdPs (including subscriber-controlled wallets), and RPs **SHALL** store all private and shared keys used for signing, encryption, and any other cryptographic operations in a secure fashion. Key storage is subject to applicable [FIPS140] requirements of the FAL at which the key is being used, including applicable tamper resistance requirements.

Some circumstances, such as reaching FAL3 with a subscriber-controlled wallet, require the cryptographic keys to be stored in a non-exportable manner. To be considered non-exportable, key storage **SHALL** either be a separate piece of hardware or an embedded processor or execution environment, e.g., secure element, trusted execution environment (TEE), or trusted platform module (TPM). These hardware modules or embedded processors are separate from a host processor such as the CPU on a laptop or mobile device. Non-exportable key storage **SHALL** be designed to prohibit the export of the secret keys to the host processor and **SHALL NOT** be capable of being reprogrammed by the host processor to allow the secret keys to be extracted.

3.5.3. Software Attestations

Software and device attestation can be used to augment the establishment of cryptographic keys and identifiers, especially in dynamic and distributed systems. Attestations in this usage are cryptographically-bound statements that a particular piece of software, device, or runtime system meets a set of agreed-upon parameters. The attestation is presented by the software in the context of establishing its identity, and allows the receiver to verify the request with a higher degree of certainty than they would otherwise.

For example, a specific distribution of subscriber-controlled wallet software can be signed by its distributor, allowing RPs to recognize individual instances of that software. Alternatively, an RP could be issued an attestation from a federation authority, allowing IdPs to recognize the RP as part of the federation.

When attestations are required by the trust agreement or requested as part of the federation protocol, received attestations **SHALL** be validated by the receiver.

See [RFC7591] Sec. 2.3 for more information about *software statements*, which are a means for OAuth and OpenID Connect RPs to communicate a signed set of software attributes during dynamic client registration.

3.6. Authentication and Attribute Disclosure

Once the IdP and RP have entered into a trust agreement and have completed registration, the federation protocol can be used to pass subscriber attributes from the IdP to the RP.

A subscriber's attributes **SHALL** be transmitted between IdP and RP only for federation transactions or support functions such as identification of compromised subscriber accounts as discussed in Sec. 3.9. A subscriber's attributes **SHALL NOT** be transmitted for any other purposes, even when parties are allowlisted.

A subscriber's attributes **SHALL NOT** be used by the RP for purposes other than those stipulated in the trust agreement. A subscribers attributes **SHALL** be stored and managed in accordance with Sec. 3.10.3.

The subscriber **SHALL** be informed of the transmission of attributes to an RP. In the case where the authorized party is the organization, the organization **SHALL** make available to the subscriber the list of approved RPs and the associated sets of attributes sent to those RPs. In the case where the authorized party is the subscriber, the subscriber **SHALL** be prompted prior to release of attributes using a runtime decision at the IdP as described in Sec. 4.6.1.3.

3.7. RP Subscriber Accounts

It is common for an RP to keep a record representing a subscriber local to the RP itself, known as the *RP subscriber account*. The RP subscriber account can contain things like access rights at the RP as well as a cache of identity attributes for the subscriber. An active RP subscriber account is bound to one or more federated identifiers from the RP's trusted IdPs. Successful authentication of one of these federated identifiers through a federation protocol allows the subscriber to access the information and functionality protected by the RP subscriber account.

An RP subscriber account is *provisioned* when the RP has associated a set of attributes about the subscriber with a data record representing the subscriber account at the RP. The RP subscriber account **SHALL** be bound to at least one federated identifier, and a given federated identifier is bound to only one RP subscriber account at a given RP. The provisioning can happen prior to authentication or as a result of the federated authentication process, depending on the deployment patterns as discussed in Sec. 4.6.3. Prior to being provisioned, the RP subscriber account does not exist and has no associated data record at the RP.

An RP subscriber account is *terminated* when the RP removes all access to the account at the RP. Termination **SHALL** include removal of all federated identifiers and bound authenticators from the RP subscriber account (to prevent future federation transactions) as well as removal of attributes and information associated with the account in accordance with Sec. 3.10.3. An RP **MAY** terminate an RP subscriber account independently from the IdP for a variety of reasons, regardless of the current validity of the subscriber account from which it is derived.

The RP subscriber account can be provisioned at the RP without an authenticated session, but an authenticated session can only be created on a provisioned account. See Sec. 3.8 for more information.

3.7.1. Account Linking

A single RP subscriber account **MAY** be associated with more than one federated identifier. This practice is sometimes known as *account linking*. If the RP allows a subscriber to manage multiple accounts in this way, the RP **SHALL** require an authenticated session with the subscriber account for all management and linking functions. This authenticated session **SHOULD** require one existing federated identifier before linking the new federated identifier to the RP subscriber account. An RP **MAY** offer a means of recovery of an RP subscriber account with no current means of access.

When a federated identifier is removed from an RP subscriber account, the RP **SHALL** disallow access to the RP subscriber account from the removed federated identifier.

The RP **SHALL** document its practices and policies that it enacts when an RP subscriber account reaches a state of having zero associated federated identifiers, no means of access, and no means of recovery. In such cases, the RP subscriber account **SHOULD** be terminated and information associated with the account in accordance with Sec. 3.10.3.

The RP **SHALL** provide notice to the subscriber when:

- A new federated identifier is added to an existing RP subscriber account
- A federated identifier is removed from an RP subscriber account, but the account is not terminated

For additional considerations on providing notice to a subscriber about account management events, see Sec. 4.6 of [SP800-63B].

The RP **MAY** associate different access rights to the same account depending on which federated account is used to access the RP. The means by which an RP determines authorization and access is out of scope of these guidelines.

3.7.2. Account Resolution

If the RP has access to existing information about a set of subscribers, and this information is not associated with a federated identifier, the RP performs a process known as *account resolution* to determine which set of subscriber information to associate with a new RP subscriber account.

An RP performing account resolution **SHALL** ensure that the attributes requested from the IdP are sufficient to uniquely resolve the subscriber within the RP's system before linking the federated identifier with the RP subscriber account and granting access. The intended use of each attribute by the RP is detailed in the trust agreement, including whether the attribute is used for account resolution in this manner.

An RP performing account resolution **SHALL** perform a risk assessment to ensure that the resolution process does not associate an RP subscriber account's information with a federated identifier not belonging to the subscriber.

A similar account resolution process is also used when the RP verifies an authenticator used in a holder-of-key assertion for the first time. In this case, the RP **SHALL** ensure that the attributes carried with the authenticator uniquely resolve to the subscriber.

3.7.3. Alternative Authentication Processes

The RP **MAY** allow a subscriber to access their RP subscriber account using direct authentication processes by allowing the subscriber to add and remove authenticators in the RP subscriber account. The RP **SHALL** follow the requirements in [SP800-63B] in managing all alternative authenticators.

Since the RP is using the direct authentication model discussed in [SP800-63], there is no federation transaction and therefore no FAL assigned.

If the RP allows this kind of access, the RP **SHALL** disclose in the trust agreement:

- The process for adding and removing alternative authenticators in the RP subscriber account
- Any restrictions on authenticators the subscriber can use to access the RP
- The AAL required for access to the subscriber account without a federation transaction
- The circumstances under which the RP will require the subscriber to authenticate with their IdP, such as a period of time since last federation transaction

For additional considerations on providing notice to a subscriber about authenticator management events, see Sec. 4.6 of [SP800-63B].

While it is possible for a bound authenticator to be used as an alternative authenticator for direct access to the RP, these uses are distinct from each other and an RP needs to determine that the use of a given authenticator can be used in one or both scenarios.

3.8. Authenticated Sessions at the RP

The end goal of a federation transaction is creating an authenticated session between the subscriber and the RP, backed by a verified assertion from the IdP. This authenticated session can be used to allow the subscriber access to functions at the RP (i.e., logging in), identifying the subscriber to the RP, or processing attributes about the subscriber carried in the federation transaction. An authenticated session **SHALL** be created by the RP only when the following conditions are true:

- The RP has processed and verified a valid assertion
- The assertion is from the expected IdP for a transaction
- The IdP that issued the assertion is the IdP identified in the federated identifier of the assertion
- The assertion is associated with an RP subscriber account (which may be ephemeral)
- The RP subscriber account has been provisioned at the RP through the method specified in the trust agreement

If the assertion is a holder-of-key assertion at FAL3, the authenticator indicated in the assertion **SHALL** be verified before the RP subscriber account is associated with an authenticated session, as discussed in Sec. 3.14. If the assertion also requires authentication with a bound authenticator at FAL3, a bound authenticator **SHALL** be verified before the RP subscriber account is associated with an authenticated session, as discussed in Sec. 3.15.

The authenticated session **MAY** be ended by the RP at any time.

See [SP800-63B] Sec. 5 for more information about session management requirements for both IdPs and RPs. For additional session requirements with general purpose IdPs, see Sec. 4.7.

3.9. Privacy Requirements

The ultimate goal of a subscriber is to interact with and use the RP. Federation involves the transfer of personal attributes from a third party that is not otherwise involved in a transaction — the IdP. Federation also potentially gives the IdP broad visibility into subscriber activities and status. Accordingly, there are specific privacy requirements associated with federation which do not exist in direct authentication.

When the RP requests a federation transaction from the IdP, this request and the subsequent processing of the federation transaction reveals to the IdP where the subscriber is logging in. Over time, the IdP could build a profile of subscriber transactions based on this knowledge of which RPs a given subscriber is using. This aggregation could enable new opportunities for subscriber tracking and use of profile information that do not align with subscribers' privacy interests.

If the same subscriber account is asserted to multiple RPs, and those RPs communicate with each other, the colluding RPs could track a subscriber's activity across multiple applications and security domains. The IdP **SHOULD** employ technical measures, such as the use of pairwise pseudonymous identifiers described in Sec. 3.3.1 or privacy-enhancing cryptographic protocols, to provide disassociability and discourage subscriber activity tracking and profiling between RPs.

The following requirements apply specifically to federal agencies acting as an IdP, an RP, or both:

1. The agency **SHALL** consult with their Senior Agency Official for Privacy (SAOP) to conduct an analysis determining whether the requirements of the Privacy Act are triggered by the agency that is acting as an IdP, by the agency that is acting as an RP, or both (see Sec. 7.4).
2. The agency **SHALL** publish or identify coverage by a System of Records Notice (SORN) as applicable.
3. The agency **SHALL** consult with their SAOP to conduct an analysis determining whether the requirements of the E-Government Act are triggered by the agency that is acting as an IdP, the agency that is acting as an RP, or both.
4. The agency **SHALL** publish or identify coverage by a Privacy Impact Assessment (PIA) as applicable.
5. The agency **SHALL** conduct a privacy risk assessment regarding the sharing of subscriber identity information between the IdP and RP.

If the RP subscriber account lifecycle process gives the RP access to attributes through a provisioning API as discussed in Sec. 4.6.3, additional privacy measures **SHALL** be implemented to account for the difference in RP subscriber account lifecycle. The IdP **SHALL** minimize the attributes made available to the RP through the provisioning API. The IdP **SHALL** limit the population of subscriber accounts available via the provisioning API to the population of subscribers authorized to use the RP by the trust agreement. To prevent RP retention of identity attributes for accounts that have been terminated at the IdP, the IdP **SHALL** use the provisioning API to de-provision RP subscriber accounts for terminated subscriber accounts.

Trust agreements **SHOULD** require identity attributes be shared only when the subscriber opts in, using a runtime decision as discussed in Sec. 4.6.1.3.

3.9.1. Transmitting Subscriber Information

The IdP **SHALL** limit transmission of subscriber information to only that which is necessary for functioning of the system. These functions include the following:

- identity proofing, authentication, or attribute assertions (collectively “identity service”); or
- in the case of a specific subscriber request to transmit the information

The IdP **MAY** additionally transmit the subscriber’s information in the following cases, if stipulated and disclosed by the trust agreement:

- fraud mitigation related to the identity service,
- to respond to a security incident related to the identity service, or
- to comply with law or legal process.

If an IdP discloses information on subscriber activities at an RP to any party, or processes the subscriber’s attributes for any purpose other than these cases, the IdP **SHALL** implement measures to maintain predictability and manageability commensurate with the privacy risk arising from the additional processing. Measures **MAY** include providing clear notice, obtaining subscriber consent, or enabling selective use or disclosure of attributes. When an IdP uses consent measures for this purpose, the IdP **SHALL NOT** make consent for the additional processing a condition of the identity service.

An RP **MAY** disclose information on subscriber activities to the associated IdP in the following cases, if stipulated and disclosed by the trust agreement:

- fraud mitigation related to the identity service,
- to respond to a security incident related to the identity service, or
- to comply with law or legal process.

See [NISTIR8062] for additional information on privacy engineering and risk management.

3.10. Security Controls

The IdP and RP **SHALL** employ appropriately tailored security controls from the moderate baseline security controls defined in [SP800-53] or equivalent federal (e.g., [FEDRAMP]) or industry standard that the organization has determined for the information systems, applications, and online services that these guidelines are used to protect. The IdP and RP **SHALL** ensure that the minimum assurance-related controls for the appropriate systems, or equivalent, are satisfied.

3.10.1. Protection from Injection Attacks

An *injection attack* in the context of a federated protocol consists of an attacker attempting to force an RP to accept or process an assertion or assertion reference in order to gain access to the RP or deny a legitimate subscriber access to the RP. The attacker does this by taking an assertion or assertion reference and injecting it into a vulnerable RP. If the attacker is able to do this successfully, the attacker can trick an RP into binding the attacker's session to the federated identifier in the assertion. The attackers assertion could be either stolen from a legitimate subscriber or manufactured to perpetrate the attack.

Protection from injection attacks is recommended at all FALs, and this protection is required at FAL2 and above. In all cases, the RP needs to take reasonable steps to prevent an attacker from presenting an injected assertion or assertion reference based on the nature of the RP software, the capabilities of the federation protocol in use, and the needs of the overall system. Both [OIDC] and [SAML] provide mechanisms for injection protection including nonces sent from the RP during the request, RP authentication for back-channel communications, and methods for the RP to start the federation transaction and track its state throughout the process. Different mechanisms provide different degrees of protection and are applicable in different circumstances. While the details of specific protections will vary based on the federation protocol and technology in use, common best practices such as the following can be used to limit the attack surface:

- The use of back channel assertion presentation as discussed in Sec. 4.11.1, which prevents an attacker from presenting the assertion directly to the RP.
- The use of an unguessable value to tie the unauthenticated session at the RP with the request to the back channel, which prevents an attacker from injecting an assertion reference from one session to another.
- Requiring the RP to authenticate to the IdP during an assertion request, preventing the attacker from faking a request from the RP to begin a federation process.
- Prohibition of IdP-initiated federation processes, which prevent the RP from accepting unsolicited assertions and assertion references from the IdP. This prohibition does not include processes in which an external party (such as the IdP

or a federation authority) signals the RP to start a federation process with the IdP, allowing the RP to begin the federation transaction and securely await a response within that transaction.

- The use of a signed front channel response from the IdP with an RP-provided nonce covered by the signature, which prevents the attacker from injecting an assertion reference from one session to another.
- The use of platform APIs for front-channel communication, as opposed to HTTP redirects.

Injection attacks are particularly dangerous when combined with phishing attacks. When combined, the attacker can either trick the subscriber into generating a valid assertion for the attacker to inject into the attacker's session, or the attacker can trick the subscriber into injecting the attacker's assertion into the subscriber's session at the RP.

3.10.2. Protecting Subscriber Information

Communications between the IdP and the RP **SHALL** be protected in transit using an authenticated protected channel. Communications between the subscriber and either the IdP or the RP (usually through a user agent) **SHALL** be made using an authenticated protected channel.

Note that the IdP may have access to information that may be useful to the RP in enforcing security policies, such as device identity, location, system health checks, and configuration management. If so, it may be a good idea to pass this information along to the RP within the bounds of the subscriber's privacy preferences described in Sec. 7.2.

Additional attributes about the user **MAY** be included outside of the assertion itself by use of authorized access to an identity API as discussed in Sec. 3.11.3. Splitting user information in this manner can aid in protecting user privacy and can allow for limited disclosure of identifying attributes on top of the essential information in the authentication assertion itself.

When derived attribute values are available and fulfill the RP's needs, the RP **SHOULD** request derived attribute values rather than full attribute values as described in Sec. 7.3. The IdP **SHOULD** support derived attribute values to the extent the underlying federation protocol allows.

3.10.3. Storing Subscriber Information

The IdP and RP **SHALL** delete personal identity information in the subscriber account and RP subscriber account (respectively) upon account termination, unless required otherwise by legal action or policy. Whenever personal identity information is stored in a subscriber account or RP subscriber account, whether the account is active or not, the IdP and RP **SHALL** determine and use appropriate controls to ensure secure storage of the personal identity information.

For example, the RP could record the federated identifier in access and audit logs, which logs are retained even after the account has been terminated. However, all identity attributes and personal information are removed from the RP's own storage.

When the RP uses an ephemeral provisioning mechanism as described in Sec. 4.6.3, the RP **SHALL** remove all subscriber attributes at the termination of the session, unless required by legal action or policy.

3.11. Identity Attributes

Identity attributes representing the subscriber are sent to the RP during a federation transaction. These attributes take on multiple aspects, which can be combined in different ways.

Bundling:
Attributes **SHALL** be either *unbundled* (presented directly by the IdP) or *bundled* into a package that is cryptographically signed by the CSP, as described in Sec. 3.11.1.

Derivation:
Attributes **SHALL** be either *attribute values* (e.g., a date of birth) or *derived attribute values* (e.g., an indication of age of majority).

Presentation:
Attributes **SHALL** be either presented in the assertion (and therefore covered by the assertion's signature) or made available as part of a protected identity API.

Trust agreements **SHALL** record the validation practices for all attributes made available under the trust agreement (e.g., whether the attribute is from an authoritative or credible source, self-asserted by the subscriber, assigned by the IdP, etc.).

3.11.1. Attribute Bundles

> Note: Attribute bundles are often referred to elsewhere as *credentials* by other protocols and specifications, but usage of this term would be in conflict with its use within these guidelines for a different concept. Consequently, the term attribute bundle is used within these guidelines instead.

As an alternative to sending attributes directly from the IdP, attributes can be collected into bundles that are signed by the CSP. These attribute bundles can be independently verified by the RP. This pattern is commonly used by a subscriber-controlled wallet.

Some examples of technologies used to bundle attributes are Selective Disclosure JSON Web Tokens [SD-JWT] and the mDoc security object defined in [ISOIEC18013-5].

The presentation of an attribute bundle **SHALL** be protected by the IdP in the same manner as non-bundled attributes. That is to say, attribute bundles presented in an assertion are covered by the signature of the assertion, and attribute bundles made available by an identity API are protected by the limited access controls to that API.

Attribute bundles include one or more attribute values and derived attribute values. Attribute bundles are carried in the assertion from the IdP, the subscriber attributes within the bundle need not be fully disclosed to all RPs on every transaction and instead **MAY** be selectively disclosed to the RP. An attribute bundle using selective disclosure technology can effectively limit which attributes an RP can read from the attribute bundle. The RP can still verify the signature of the attribute bundle as a whole, confirming its source as the CSP, without the IdP having to disclose all of the contents of the attribute bundle to the RP.

The RP **SHALL** validate the signature covering the attribute bundle itself as well as the signature of the assertion as a whole. The RP **SHALL** ensure that the attribute bundle is able to be presented by the IdP that created the assertion containing the attribute bundle, such as by verifying that the public key used to sign the assertion is included in the signature of the attribute bundle.

3.11.2. Derived Attribute Values

For some use cases, knowing the actual value of an identity attribute is not strictly necessary for the RP to function, but a value derived from the identity attribute is sufficient instead. For example, if the RP needs to know if the subscriber is above the age of majority, the RP could request the subscriber’s birth date and calculate the majority age question from this value. However, doing so reveals more specific information to the RP than it truly needed. Instead, if the IdP can calculate whether the subscriber’s age meets the definitions for majority at the time of the RP’s request and return a simple boolean for this derivation instead of the birth date value itself. The RP can then continue its processing without needing to see the underlying value.

Derived attribute values increase the privacy of a system since they allow a more focused release of information to the RP. While some federation systems allow the RP to dynamically query for an arbitrary derived attribute value at request time, many common use cases can be accommodated by the IdP pre-calculating common derived attribute values and offering them as alternatives to the full attribute value.

3.11.3. Identity APIs

Attributes about the subscriber, including profile information, **MAY** be provided to the RP through a protected API known as the *identity API*. The RP is granted limited access to the identity API during the federation transaction, in concert with the assertion. For example, in OpenID Connect, the UserInfo Endpoint provides a standardized identity API for fetching attributes about the subscriber. This API is protected by an OAuth 2.0 Access Token, which is issued to the RP along with OpenID Connect's assertion, the ID Token.

By making attributes available at an identity API, the IdP no longer has to use the assertion to convey as much information to the RP. This not only means that sensitive attributes do not have to be carried in the assertion itself, it also makes the assertion smaller and easier to process by the RP. The contents of the assertion can then be limited to essential fields (e.g., unique subject identifiers) and information about the immediate authentication event being asserted.

Identity APIs also make it possible for the RP to help manage when subscriber attributes are transmitted from the IdP. The RP often caches attributes provided by the IdP in an RP subscriber account, discussed in Sec. 3.10.1, and the RP can record when these attributes were last received from the IdP. The RP can request subscriber attributes only when needed to update the RP subscriber account, instead of receiving them on every federation transaction in the assertion. The IdP can aid this decision by indicating in the assertion the time at which any of the subscriber attributes available to the RP were updated at the IdP. This approach is particularly helpful when a subscriber's attributes are stable over time, allowing the RP to function without fetching them on every request.

All possible use of identity APIs, including which provisioning models are available through the API, **SHALL** be recorded and disclosed as part of the trust agreement. Access to the identity API **SHALL** be time limited by the trust agreement. Access to the identity API **SHOULD** be limited to the duration of the federation transaction plus time necessary for synchronization of attributes, as discussed in Sec. 4.6.4. Since the time limitation is separate from the validity time window of the assertion and the lifetime of the authenticated session at the RP, access to an identity API by the RP without an associated valid assertion **SHALL NOT** be sufficient for the establishment of an authenticated session at the RP.

A given identity API deployment is expected to be capable of providing attributes for all subscribers for whom the IdP can create assertions. However, when access to the identity API is granted within the context of a federation transaction, the attributes provided by an identity API **SHALL** be associated with only the single subscriber identified in the associated assertion. If the identity API is hosted by the IdP, the returned attributes **SHALL** include the subject identifier for the subscriber. This allows the RP to positively correlate the assertion's subject to the returned attributes. Note that when access to an identity API is provided as part of pre-provisioning of RP subscriber accounts as discussed in Sec. 4.6.3, the RP is usually granted blanket access to the identity API outside the context of the federation transaction and these requirements do not apply. For pre-provisioning use cases, the privacy considerations **SHALL** be evaluated and recorded as part of the trust agreement. If the identity API is hosted externally, the requirements in Sec. 3.11.3.1 apply.

3.11.3.1. External Identity APIs

While most identity APIs used in federation protocols are hosted as part of the IdP, it is also possible for the IdP to grant access to identity APIs hosted directly by attribute providers. These services provide attributes about the subscriber in addition to those made available directly from the IdP.

When the IdP grants access to an external attribute provider, the IdP is making an explicit statement that the information returned from the attribute provider is associated with the subscriber identified in the associated assertion. For the purposes of the trust agreement, the IdP is the responsible party for the accuracy and content of the identity API and its association with the represented subscriber account.

The attributes returned by the attribute provider are assumed to be independent of those returned directly from the IdP, and as such **MAY** use different identifiers, formats, or schemas.

For example, an IdP could provide access to a subscriber's medical license information as part of the federation process. Instead of the IdP asserting the license status directly, the IdP provides the RP access to a record for the subscriber at a medical licensure agency by providing a link to an API containing the record representing the subscriber as well as a credential allowing limited access to this API. The RP can then make a strong association between the current subscriber and the license record, even though the license record will likely use a different subject identifier and would otherwise be not correlatable by the RP. The trust agreement would list the medical licensure agency as an additional attribute provider to the IdP. The IdP remains responsible for providing this linked data.

Before accepting attributes from an external identity provider and associating them with the RP subscriber account, the RP **SHALL** verify that the attributes in question are allowed to be provided by the external attribute provider under the auspices of the trust agreement.

3.12. Assertion Protection

Assertions **SHALL** include a set of protections to prevent attackers from manufacturing valid assertions or reusing captured assertions at disparate RPs. The protections required are dependent on the details of the use case being considered, and specific protections are listed here.

3.12.1. Assertion Identifier

Assertions SHALL be sufficiently unique to permit unique identification by the target RP. Assertions MAY accomplish this by use of an embedded nonce, issuance timestamp, assertion identifier, or a combination of these or other techniques.

3.12.2. Signed Assertion

Assertions SHALL be cryptographically signed by the issuer (IdP). The RP SHALL validate the digital signature or MAC of each such assertion based on the issuer's key. This signature SHALL cover the entire assertion, including its identifier, issuer, audience, subject, and expiration.

The assertion signature SHALL either be a digital signature using asymmetric keys or a MAC using a symmetric key shared between the RP and issuer. Shared symmetric keys used for this purpose by the IdP SHALL be independent for each RP to which they send assertions, and are normally established during registration of the RP. Public keys for verifying digital signatures SHALL be transferred to the RP in a secure manner, and MAY be fetched by the RP in a secure fashion at runtime, such as through an HTTPS URL hosted by the IdP. Approved cryptography SHALL be used.

3.12.3. Encrypted Assertion

The contents of the assertion can be encrypted to protect their exposure to untrusted third parties, such as a user agent. This protection is especially relevant when the assertion contains PII of the subscriber—excluding opaque identifiers such as the subject identifier. Subject identifiers are meaningless outside of their target systems, unlike other possible identifiers such as SSN, email address, or driver's license number. Therefore, subject identifiers are excluded as a qualifier for encrypting the assertion. A trust agreement MAY require encryption of assertion contents in other situations.

When the entire assertion is encrypted, the encryption protects the contents of the assertion from being read by unintended parties, ensuring that only the targeted RP is able to process the assertion. While most assertion formats support encryption of the entire assertion, some assertion formats allow for only the PII portions of the assertion to be encrypted, providing selective disclosure of sensitive information to the RP without encrypting the entire assertion.

When encrypting assertions, the IdP SHALL encrypt the contents of the assertion using either the RP's public key or a shared symmetric key. Shared symmetric keys used for this purpose by the IdP SHALL be independent for each RP to which they send assertions, and are normally established during registration of the RP. Public keys for encryption SHALL be transferred over an authenticated protected channel and MAY be fetched by the IdP at runtime, such as through an HTTPS URL hosted by the RP.

All encryption of assertions **SHALL** use approved cryptography applied to the federation technology in use. For example, a SAML assertion can be encrypted using XML-Encryption, or an OpenID Connect ID Token can be encrypted using JSON Web Encryption (JWE). When used with back-channel presentation, an assertion can also be encrypted with a mutually-authenticated TLS connection, so long as there are no intermediaries between the IdP and RP that interrupt the TLS channel.

3.12.4. Audience Restriction

Assertions **SHALL** use audience restriction techniques to allow an RP to recognize whether or not it is the intended target of an issued assertion. All RPs **SHALL** check that the audience of an assertion contains an identifier for their RP to prevent the injection and replay of an assertion generated for one RP at another RP.

In order to limit the places that an assertion could successfully be replayed by an attacker, IdPs **SHOULD** issue assertions designated for only a single audience. Restriction to a single audience is required at FAL2 and above.

3.13. Bearer Assertions

A bearer assertion can be presented on its own as proof of the identity of the party presenting it. No other proof beyond validation of the assertion is required. Similarly, a bearer assertion reference can be presented own its own to the RP and used by the RP to fetch an assertion. If an attacker can capture or manufacture a valid assertion or assertion reference representing a subscriber and can successfully present that assertion or reference to the RP, then the attacker could be able to impersonate the subscriber at that RP.

Note that mere possession of a bearer assertion or reference is not always enough to impersonate a subscriber. For example, if an assertion is presented in the back-channel federation model (described in Sec. 4.11.1), additional controls can be placed on the transaction (such as identification of the RP and assertion injection protections) that help further protect the RP from fraudulent activity.

3.14. Holder-of-Key Assertions

A holder-of-key assertion as in Fig. 3 **SHALL** include a unique identifier for an authenticator that can be verified independently by the RP, such as the public key of a certificate controlled by the subscriber. The RP **SHALL** verify that the subscriber possesses the authenticator identified by the assertion.

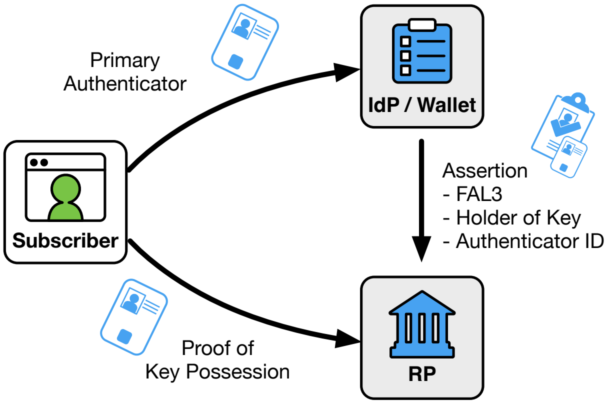

Fig. 3. Holder-of-Key Assertions

The authenticator identified in a holder-of-key assertion **MAY** be distinct from the primary authenticator the subscriber uses to authenticate to the IdP. The authenticator identified in a holder-of-key assertion **SHALL** be phishing resistant. When the RP encounters an authenticator in a holder-of-key assertion for the first time, the RP **SHALL** ensure that the authenticator can be uniquely resolved to the RP subscriber account, as discussed in Sec. 3.7.2.

A holder-of-key assertion **SHALL NOT** include an unencrypted private or symmetric key to be used as an authenticator.

When the RP uses an ephemeral provisioning mechanism as described in Sec. 4.6.3, the IdP **SHOULD** use a unique pairwise identifier for each authorization request to the RP to prevent the RP from storing or correlating information.

A more complete example is found in Sec 10.6, which shows the use of a mutual TLS connection to provide the proof of possession of a certificate on a smart card that is listed by the assertion.

Since the authenticators used in holder-of-key assertions are presented to multiple parties, and these authenticators often contain identity attributes, there are additional privacy considerations to address as discussed in Sec. 7.

3.15. Bound Authenticators

A bound authenticator as shown in Fig. 4 is an authenticator bound to the RP subscriber account and managed by the RP. The IdP **SHALL** include an indicator in the assertion when the assertion is to be used with a bound authenticator at FAL3. The unique identifier for the authenticator (such as its public key) **SHALL** be stored in the RP subscriber account. The RP needs to have a reliable basis for evaluating the characteristics of the bound authenticator; one such basis is the inclusion of a signed attestation, as discussed in Sec. 3.2.4 of [SP800-63B].

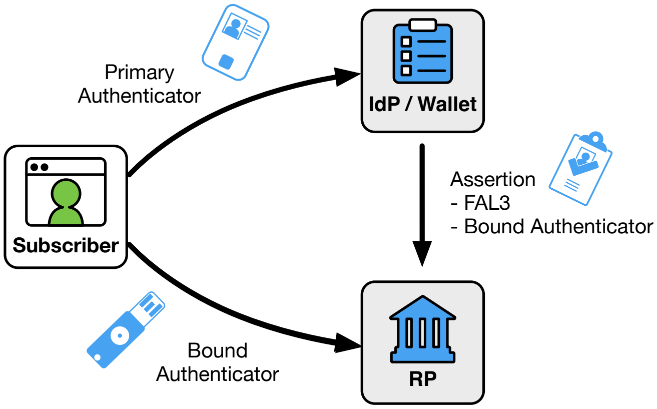

Fig. 4. Bound Authenticators

A bound authenticator **SHALL** be unique per subscriber at the RP such that two subscribers cannot present the same authenticator for their separate RP subscriber accounts. All bound authenticators **SHALL** be phishing resistant. Consequently, subscriber-chosen values such as a password cannot be used as bound authenticators. The RP **SHALL** accept authentication from a bound authenticator only in the context of processing an FAL3 assertion for a federation transaction. While it's possible for the same authenticator to also be used for direct authentication to the RP, such use is not considered a bound authenticator and the RP **SHALL** document these as distinct use cases.

Before an RP can successfully accept an FAL3 assertion, the RP subscriber account **SHALL** include a reference to a bound authenticator that is to be verified during the FAL3 transaction. These authenticators can be provided by either the RP or the subscriber, with slightly different requirements applying to the initial binding of the authenticator to the RP subscriber account in each case.

The RP **SHALL** send a notification to the subscriber via a mechanism that is independent of the transaction binding the new authenticator (e.g., an email to an address previously associated with the subscriber), and **SHOULD** notify the IdP using a shared signaling system (see Sec. 4.8), if any of the following events occur:

- A new bound authenticator is added to the RP subscriber account.
- An existing bound authenticator is removed from the RP subscriber account.

For additional considerations on providing notice to a subscriber about authenticator management events, see Sec. 4.6 of [SP800-63B].

3.15.1. RP-Provided Bound Authenticator Issuance

For RP-provided authenticators, the administrator of the RP **SHALL** issue the authenticator to the subscriber directly for use with an FAL3 federation transaction. The administrator of the RP **SHALL** store a unique identifier for the bound authenticator in the RP subscriber account, such as the public key of the authenticator.

The administrator of the RP **SHALL** determine through independent means that the identified subject of the RP subscriber account is the party to which the authenticator is issued.

For example, consider an RP that has a collection of cryptographic authenticators that it has purchased for use with FAL3 authentication. These authenticators are each provisioned to a specific RP subscriber account, but are held in a controlled environment by the administrator of the RP. To issue the authenticator, the RP could use an in-person process in which the administrator of the RP has the subscriber authenticate to an RP-controlled kiosk using an FAL3 federation transaction from the IdP. The administrator then hands the subscriber the bound authenticator indicated by the RP subscriber account and has them authenticate to the kiosk using that. The subscriber is now in possession of a bound authenticator supplied by the RP, which can be used to reach FAL3 for future transactions. Alternatively, the administrator of the RP could send the authenticator to a verified address for the subscriber and have the subscriber verify receipt through an activation process. Since the use of the bound authenticator still requires a valid assertion from the IdP, interception of the authenticator alone is not sufficient for accessing the RP subscriber account.

3.15.2. Subscriber-Provided Bound Authenticator Binding Ceremony

The RP **MAY** provide a process for associating subscriber-provided authenticators to the RP subscriber account on a trust-on-first-use basis. This process is known as a *binding ceremony* and has additional requirements beyond a typical FAL3 federation process. This is similar to the subscriber-provided authenticator binding process discussed in Sec. 4.1.3 of [SP800-63B].

If no bound authenticators are associated with the RP subscriber account, the RP **SHALL** perform a binding ceremony to establish the connection between the authenticator, the subscriber, and the RP subscriber account as shown in Fig. 5. The RP **SHALL** first establish an authenticated session using federation with an assertion that meets all the other requirements of FAL3, including an indication that the assertion is intended for use at

FAL3 with a bound authenticator. The subscriber **SHALL** immediately be prompted to present and authenticate with the proposed authenticator. Upon successful presentation of the authenticator, the RP **SHALL** store a unique identifier for the authenticator (such as its public key) and associate this with the RP subscriber account associated with the federated identifier. If the subscriber fails to successfully authenticate to the RP using an appropriate authenticator, the binding ceremony fails. The binding ceremony session **SHALL** have a timeout of five minutes or less and **SHALL NOT** be used as an authenticated session for any other purpose as described in Sec. 3.8. Upon successful completion of the binding ceremony, the RP **SHALL** immediately request a new assertion from the IdP at FAL3, including prompting the subscriber for the newly-bound authenticator.

An RP **MAY** allow a subscriber to bind multiple subscriber-provided authenticators at FAL3. If this is the case, and the RP subscriber account has one or more existing bound authenticators, the binding ceremony makes use of the existing ability to reach FAL3. The subscriber **SHALL** first be prompted to authenticate to the RP with an existing bound authenticator to reach FAL3. Upon successful authentication, the RP **SHALL** immediately prompt the subscriber to authenticate to the RP using the newly-bound authenticator.

In addition to an RP determining a bound authenticator is no longer viable, a subscriber could choose to stop using a bound authenticator for a variety of reasons, such as the authenticator being lost, compromised, or no longer usable due to technology and platform changes. In such cases, an RP **MAY** allow a subscriber to remove a subscriber-provided bound authenticator from their RP subscriber account, thereby removing the ability to use that authenticator for FAL3 sessions. When a bound authenticator is removed, the RP **SHALL** terminate all current FAL3 sessions for the subscriber and **SHALL** require reauthentication at FAL3 of the subscriber from the IdP. The RP **SHALL NOT** prompt the subscriber to authenticate with the authenticator being removed, since the subscriber will often not have access to the authenticator in question during the unbinding process, particularly in cases where the authenticator is lost or compromised.

This option is particularly helpful in situations where the subscriber already has access to an appropriate authenticator that the RP wants to allow them to use for FAL3 transactions. For example, a subscriber could have a single-factor cryptographic authenticator which uses name-based phishing resistance as described in Sec. 3.2.5.2 of [SP800-63B]. With such a device, the IdP and RP would see different keys when the authenticator is used in each location, meaning the bound authenticator cannot be easily verified by the IdP. Furthermore, since the RP did not issue the authenticator, the RP does not know the authenticator’s key ahead of time, nor does it know which subscriber account to associate to the key. Instead, the RP can use a binding ceremony as described here to allow the subscriber to use this device as a bound authenticator at FAL3. A more complete example is found in Sec 10.7.

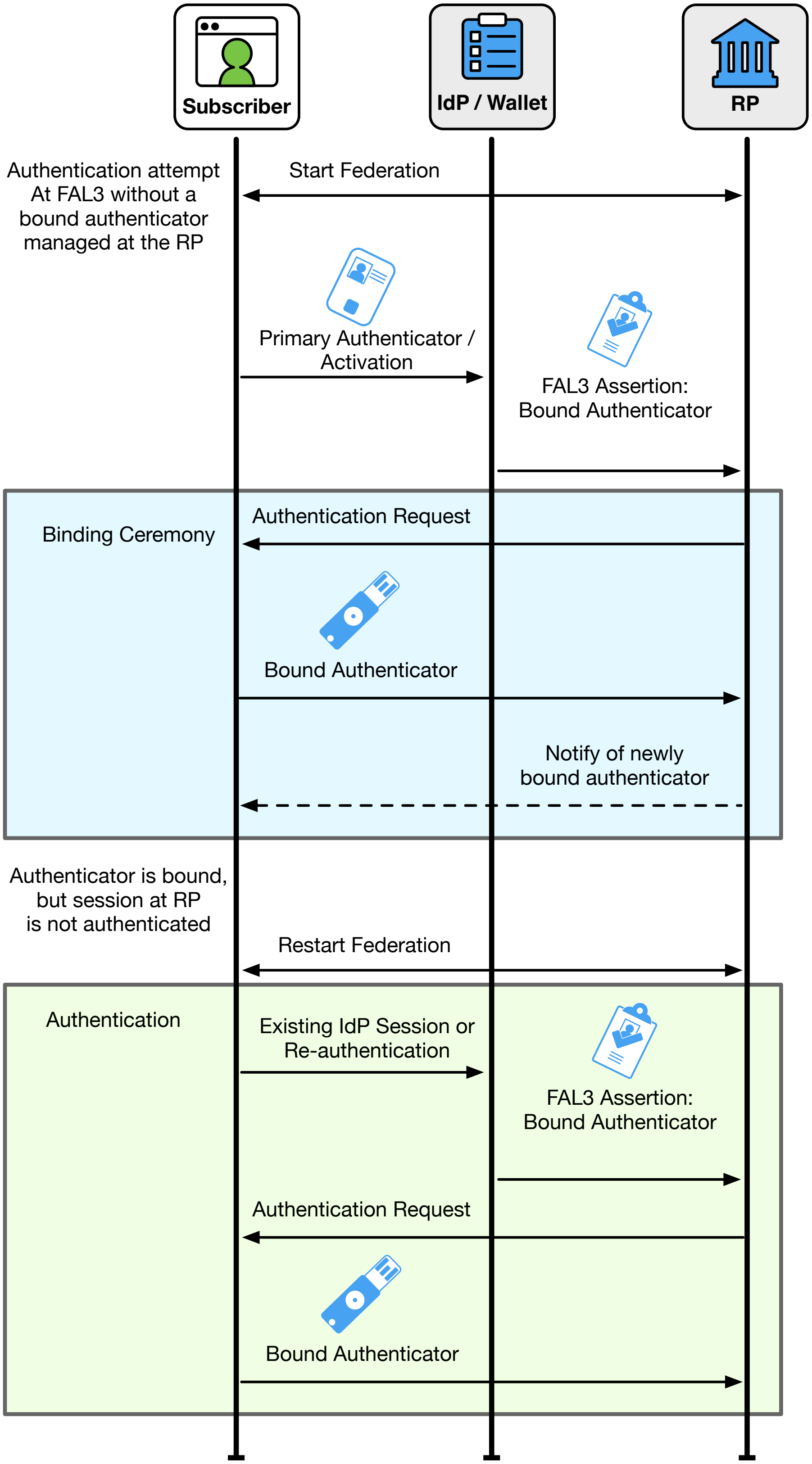
Subscriber
IdP / Wallet
RP
Authentication attempt At FAL3 without a bound authenticator managed at the RP
Start Federation
Primary Authenticator / Activation
FAL3 Assertion: Bound Authenticator
Binding Ceremony
Authentication Request
Bound Authenticator
Notify of newly bound authenticator
Authenticator is bound, but session at RP is not authenticated
Restart Federation
Authentication
Existing IdP Session or Re-authentication
FAL3 Assertion: Bound Authenticator
Authentication Request
Bound Authenticator

Fig. 5. Subscriber-Provided Bound Authenticator Binding Ceremony

3.16. RP Requirements for Processing Holder-of-Key Assertions and Bound Authenticators

When the RP receives an assertion associated with a bound authenticator, the subscriber proves possession of the bound authenticator directly to the RP. The primary authentication at the IdP and the federated authentication at the RP are processed separately. While the subscriber could use the same authenticator during the primary authentication at the IdP and as the bound authenticator at the RP, there is no assumption that these will be the same.

The following requirements apply to all assertions associated with a bound authenticator:

1. The subscriber **SHALL** prove possession of the bound authenticator to the RP, in addition to presentation of the assertion itself.
2. For a holder-of-key assertion, a reference to a given authenticator found within an assertion **SHALL** be trusted at the same level as all other information within the assertion, as stipulated in the trust agreement.
3. The RP **SHALL** process and validate the assertion in addition to the bound authenticator.
4. Failure to authenticate with the bound authenticator **SHALL** result in an error at the RP.

4. General-Purpose IdPs

This section is normative.

When the IdP is hosted on a service and not on the subscriber’s device, or when the IdP represents multiple subscribers, the IdP is known as a *general-purpose IdP* and the following requirements apply.

Digital wallets that are deployed to networked systems and not to subscriber devices are considered general-purpose IdPs for the purposes of these guidelines.

4.1. IdP Account Provisioning

In order to make subscriber accounts available through an IdP, the subscriber accounts need to be provisioned at the IdP. The means by which the subscriber account is provisioned to the IdP **SHALL** be disclosed in the trust agreement.

Due to the requirement for the IdP to be able to authenticate the subscriber, the IdP is often a service of the CSP, where the IdP has some level of access to the attributes and authenticators in the subscriber account. Such IdPs are generally in the same security domain as the IdAM that houses the subscriber account. In other cases, one or more authenticators in the subscriber account can be verified outside of the security domain, such as authenticators tied to a common PKI.

The IdP augments the subscriber account with federation-specific attributes, such as a subject identifier. The IdP can collect additional attributes, subject to the privacy and storage requirements enumerated by the trust agreement.

Once the subscriber account is provisioned to the IdP, the CSP is no longer an active participant in the federation process. Consequently, even if the RP fetches attributes through an identity API hosted by the CSP, the identity API is considered a function of the IdP and not the CSP for the purposes of these guidelines.

4.2. Federation Transaction

A federation transaction involving a general-purpose IdP establishes the subscriber account at the IdP and culminates in an authenticated session for the subscriber at the RP. This process is shown in Fig. 6.

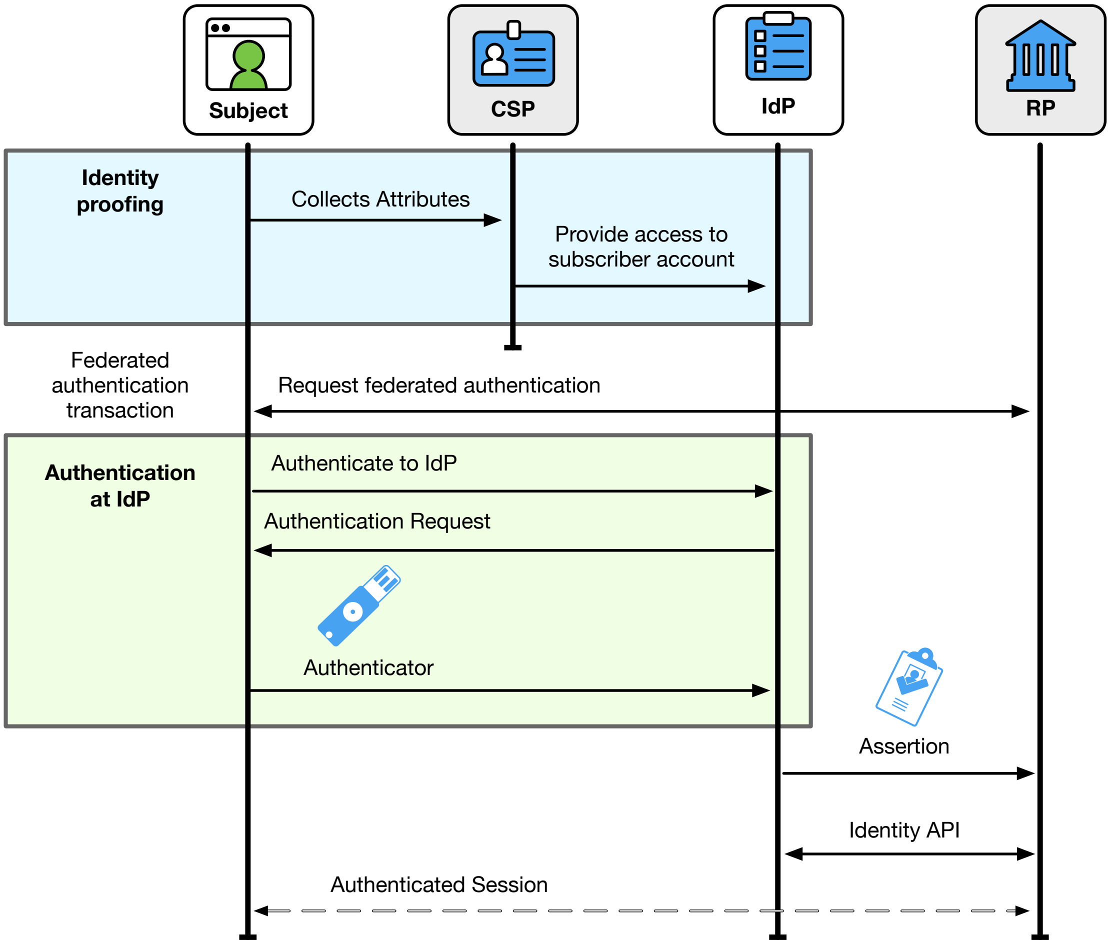

Fig. 6. Federation Overview

A federation transaction is a multi-stage process:

1. Before federation can occur, the subscriber account is established by the CSP. This account binds the identity attributes collected by the CSP to a set of authenticators used by the subscriber.
2. The subscriber account is provisioned at the IdP. The IdP augments the subscriber account with federation-specific attributes, such as a subject identifier.
3. The IdP and RP perform discovery and registration to establish the cryptographic keys and identifiers needed for information to be securely exchanged between

the parties in the federation protocol. While there may have been an existing policy decision representing a permission to connect (through an apriori trust agreement), this step entails a connection and integration at the technical level. This stage can occur before any subscriber tries to access the RP or as a response to a subscriber's attempt to use an IdP at an RP.

4. The IdP and RP begin a federated authentication transaction to authenticate a subscriber to the RP. As part of this, the set of attributes that is to be passed to the RP is selected from a subset of what the RP has requested, what is allowed by the trust agreement, and what is permitted by the authorized party. If necessary, the authorized party is prompted at runtime to approve the release of attributes.
5. The subscriber authenticates to the IdP using an authenticator bound to the subscriber account.
6. The IdP creates an assertion to represent the results of the authentication event. The assertion is based on terms established by the trust agreement, the request from the RP, the capabilities of the IdP, the subscriber account known to the IdP, and the attributes permitted by the authorized party.
7. The assertion is passed to the RP across the network.
8. The RP processes this assertion from the IdP and establishes an authenticated session with the subscriber. Optionally, the RP receives identity attributes from the IdP representing the subscriber account, either in the assertion or through an identity API.

In all transactions, the parties involved enter into a trust agreement, described in Sec. 3.4. This agreement establishes which parties are fulfilling which roles, and its execution represents initial permission for the systems in question to connect. The list of available subscriber identity attributes is established in this step, though the decision of which attributes are released to a given RP for a given transaction is finalized during the federation transaction itself.

In a federated identity transaction, the IdP is the source of identity and authentication attributes for the RP. The normal flow of information for a federation transaction is from the IdP to the RP. Due to the directional nature of this information flow, the IdP is considered to be *upstream* of the RP and the RP is considered to be *downstream* of the IdP. It is also possible for additional information to flow back up from the RP, particularly through use of shared signals as discussed in Sec. 4.8.

4.3. Trust Agreements

Trust agreements **SHALL** be established either:

- as the result of an agreement by the federated parties, prior to the federation transaction, or
- as the result of decision or action by the subscriber, during the federation transaction.

4.3.1. Apriori Trust Agreement Establishment

When the trust agreement is established by the federated parties prior to the federation transaction, the trust agreement **SHALL** establish the following terms, which **MAY** vary per IdP and RP relationship:

- The set of subscriber attributes the CSP makes available to the IdP as part of the subscriber account
- The set of subscriber attributes the IdP can make available to the RP
- The attribute storage policy of the IdP for the subscriber account, including any available means for the subscriber to request deletion
- Any additional attribute sources that the IdP receives applicable subscriber attributes from
- What if any identity APIs are made available by the IdP, either directly or through an external provider, and which subscriber attributes are available at these APIs
- The population of subscriber accounts that the IdP can create assertions for
- Any additional uses of subscriber information, beyond providing the identity service
- The set of subscriber attributes that the RP will request (a subset of the attributes made available)
- The purpose for each attribute requested by the RP
- The attribute storage policy of the RP for the RP subscriber account, including any available means for the subscriber to request deletion
- The use of any shared signaling between the IdP and RP
- The authorized party responsible for decisions regarding the release of subscriber attributes to the RP (e.g., the IdP organization, the subscriber, etc.)
- The means of informing subscribers about attribute release to the RP
- The xALs available from the IdP
- The xALs required by the RP

The terms of the trust agreement **SHALL** be available to the operators of the RP and the IdP upon its establishment. The terms of the trust agreement **SHALL** be made available to subscribers upon request to the IdP or RP.

The IdP and RP **SHALL** each assess their respective redress mechanisms for their efficacy in achieving a resolution of complaints or problems and disclose the results of this assessment as part of the trust agreement. See Sec. 3.4.3 for additional requirements and considerations for redress mechanisms.

If FAL3 is allowed within the trust agreement, the trust agreement **SHALL** stipulate the following terms regarding holder-of-key assertions and bound authenticators (see Sec. 3.14 and Sec. 3.15):

- The means by which holder-of-key assertions can be verified by the RP (such as a common trusted PKI system)
- The means by which the RP can associate holder-of-key assertions with specific RP subscriber accounts (such as attribute-based account resolution or pre-provisioning)
- Whether bound authenticators are supplied by the RP or by the subscriber
- Documentation of the binding ceremony used for any subscriber-provided bound authenticators

Runtime decisions at the IdP, as described in Sec. 4.6.1.3, **MAY** be used to further limit which subscriber attributes are sent between parties in the federated authentication process (e.g., a runtime decision could opt to not disclose an email address even though this attribute was included in the terms of the trust agreement).

The IdP and RP **SHALL** exchange only the minimum data necessary to achieve the function of the system.

The trust agreement **SHALL** be reviewed periodically to ensure it is still fit for purpose, and to avoid unnecessary data exchange and over-collection of subscriber data.

4.3.2. Subscriber-driven Trust Agreement Establishment

When the trust agreement is established as the result of a subscriber's decision, such as a subscriber starting a federation transaction between an RP and their IdP who have no established agreement, the trust agreement is anchored by the subscriber. Consequently, the following terms **SHALL** be disclosed to the subscriber upon request to the IdP and to the RP during the runtime decision at the IdP as described in Sec. 4.6.1.3:

- The set of subscriber attributes the CSP makes available to the IdP
- Any additional attribute sources that the IdP receives applicable subscriber attributes from

- What if any identity APIs are made available by the IdP, either directly or through an external provider, and which subscriber attributes are available at these APIs
- The set of subscriber attributes the IdP can make available to the RP
- The attribute storage policy of the IdP for the subscriber account, including any available means for the subscriber to request deletion
- The use of any shared signaling between the IdP and RP
- The population of subscriber accounts that the IdP can create assertions for
- Any additional uses of subscriber information, beyond providing the identity service
- The xALs available from the IdP

The IdP **SHALL** assess its redress mechanisms for their efficacy in achieving a resolution of complaints or problems and disclose the results of this assessment to the subscriber. See Sec. 3.4.3 for additional requirements and considerations for redress mechanisms.

The release of subscriber attributes **SHALL** be managed using a runtime decision at the IdP, as described in Sec. 4.6.1.3. The authorized party **SHALL** be the subscriber.

The following terms of the trust agreement **SHALL** be disclosed to the subscriber during the runtime decision:

- The set of subscriber attributes that the RP will request (a subset of the attributes made available by the IdP)
- The purpose for each attribute requested by the RP
- The attribute storage policy of the RP for the RP subscriber account, including any available means for the subscriber to request deletion
- The xALs required by the RP

Note that all information disclosed to the subscriber needs to be conveyed in a manner that is understandable and actionable, as discussed in Sec. 8.

4.4. Discovery and Registration

To perform a federation transaction with a general-purpose IdP, the RP **SHALL** associate the assertion signing keys and other relevant configuration information with the IdP's identifier, as stipulated by the trust agreement. If these are retrieved over a network connection, request and retrieval **SHALL** be made over a secure protected channel from a location associated with the IdP's identifier by the trust agreement. In many federation protocols, this is accomplished by the RP fetching the public keys and configuration data from a URL known to be controlled by the IdP or offered on the IdP's behalf. It is also possible for the RP to be configured directly with this information in a static fashion, whereby the RP's administrator enters the IdP information directly into the RP software.

Additionally, the RP **SHALL** register its information either with the IdP or with an authority the IdP trusts, as stipulated by the trust agreement. In many federation protocols, the RP is assigned an identifier during this stage, which the RP will use in subsequent communication with the IdP.

In all of these requirements, the IdP **MAY** use a trusted third party to facilitate its discovery and registration processes, so long as that trusted third party is identified in the trust agreement. For example, a consortium could make use of a hosted service that collects the configuration records of IdPs and RPs directly from participants. Instead of going to the IdP directly for its discovery record, an RP would instead go to this service. The IdP would in turn go to this service to find the identifiers and configuration information for RPs that are needed to connect.

4.4.1. Manual Registration

At all FALs, the cryptographic keys and identifiers of the RP and IdP can be exchanged in a manual process, whereby the administrator of the RP submits the RP's configuration to the IdP (either directly or through a trusted third party) and receives the identifier to use with that IdP. The RP administrator then configures the RP with this identifier and any additional information needed for the federation transaction to continue.

As this is a manual process, the registration happens prior to the federation transaction beginning.

This process **MAY** be facilitated by some level of automated tooling, whereby the manual configuration points the systems in question to a trusted source of information that can be updated over time. If such automation is used, the trust agreement **SHALL** enumerate the allowable terms of the cryptographic key distribution and assignment, including allowable cache lifetimes.

4.4.2. Dynamic Registration

At FAL1 and FAL2, the cryptographic keys and identifiers of the RP can be exchanged in a dynamic process, whereby the RP software presents its configuration to the IdP (either directly or through a trusted third party) and receives the identifier to use with that IdP. This process is specific to the federation protocol in use but requires machine-readable configuration data to be made available over the network. All transmission of configuration information **SHALL** be made over a secure protected channel to endpoints associated with the IdP's identifier by the trust agreement.

IdPs **SHOULD** consider the risks of information leakage to multiple RP instances and take appropriate countermeasures, such as issuing PPIs to dynamically registered RPs as discussed in Sec. 3.3.1.

Dynamic registration **SHOULD** be augmented by attestations about the RP software and device, as discussed in Sec. 3.5.3.

[OIDC-Registration] defines a protocol for dynamic registration of RPs at an OpenID Connect IdP.

4.5. Subscriber Authentication at the IdP

In a federation context, the IdP acts as the verifier for the authenticator bound to the subscriber account, as described in [SP800-63B]. Verification of the authenticator creates an authentication event which begins the authenticated session at the IdP. This authenticated session serves as the basis of the IdP’s claim that the subscriber is present.

The IdP **SHALL** require the subscriber to have an authenticated session before any of the following events:

- Approval of attribute release
- Creation and issuance of an assertion
- Establishment of a subscriber-driven trust agreement.

Additional requirements for session management and reauthentication are discussed in Sec. 4.7.

4.6. Authentication and Attribute Disclosure

The decision of whether a federation transaction proceeds **SHALL** be determined by the authorized party stipulated by the trust agreement. The decision can be calculated in a variety of ways, including:

- an allowlist, which determines the circumstances under which the system can allow the federation transaction to proceed in an automated fashion;
- a blocklist, which determines the circumstances under which the system will not allow the federation transaction to proceed; and
- a runtime decision, which allows the authorized party to decide if the transaction can proceed and under what precise terms. Note that a runtime decision can be stored and applied to future transactions.

The applicability of an allowlist, blocklist, or runtime decision can be influenced by aspects of the federation transaction, including the identity of the IdP and RP, the subscriber attributes requested, the xAL required, and other factors. These decisions can be facilitated by risk management systems, federation authorities, and local system policies.

For a non-normative example of an RP that has been allowlisted at an IdP for a set of subscribers to facilitate single-sign-on for an enterprise application, see Sec. 10.5.

The IdP **SHALL** provide effective mechanisms for redress of subscriber complaints or problems (e.g., subscriber identifies an inaccurate attribute value). See Sec. 3.4.3 for additional requirements and considerations for redress mechanisms.

4.6.1. IdP-Controlled Decisions

4.6.1.1. IdP Allowlists of RPs

In an a priori trust agreement, IdPs **MAY** establish allowlists of RPs authorized to receive authentication and attributes from the IdP without a runtime decision from the subscriber. When placing an RP on its allowlist, the IdP **SHALL** confirm that the RP abides the terms of the trust agreement. The IdP **SHALL** determine which identity attributes are passed to the allowlisted RP upon authentication. IdPs **SHALL** make allowlists available to subscribers as described in Sec. 7.2.

IdP allowlists **SHALL** uniquely identify RPs through the means of domain names, cryptographic keys, or other identifiers applicable to the federation protocol in use. Any entities that share an identifier **SHALL** be considered equivalent for the purposes of the allowlist. Allowlists **SHOULD** be as specific as possible to avoid unintentional impersonation of an RP.

IdP allowlist entries for an RP **SHALL** indicate which attributes are included as part of an allowlisted decision. If additional attributes are requested by the RP, the request **SHALL** be either:

- subject to a runtime decision of the authorized party to approve the additional attributes requested,
- redacted to only the attributes in the allowlist entry, or
- denied outright by the IdP.

IdP allowlists **MAY** include other information, such as the xALs under which the allowlist entry is applied. For example, an IdP could use an allowlist entry to bypass a consent screen for an FAL1 transaction but require confirmation of consent from the subscriber during an FAL3 transaction.

4.6.1.2. IdP Blocklists of RPs

IdPs **MAY** establish blocklists of RPs not authorized to receive authentication assertions or attributes from the IdP, even if requested to do so by the subscriber. If an RP is on an IdP’s blocklist, the IdP **SHALL NOT** produce an assertion targeting the RP in question under any circumstances.

IdP blocklists **SHALL** uniquely identify RPs through the means of domain names, cryptographic keys, or other identifiers applicable to the federation protocol in use. Any entities that share an identifier **SHALL** be considered equivalent for the purposes of the blocklist. For example, a wildcard domain identifier of “*.example.com” would match the domains “www.example.com”, “service.example.com”, and “unknown.example.com” equally. All three of these sites would blocked by the same blocklist entry.

4.6.1.3. IdP Runtime Decisions

Every RP that is in a trust agreement with an IdP but not on an allowlist with that IdP **SHALL** be governed by a default policy in which runtime authorization decisions will be made by an authorized party identified by the trust agreement. Since the runtime decision occurs during the federation transaction, the authorized party is generally a person and, in most circumstances, is the subscriber; however, it is possible for another party such as an administrator to be prompted on behalf of the subscriber. Note that in a subscriber-driven trust agreement, a runtime decision with the subscriber is the only allowable means to authorize the release of subscriber attributes.

When processing a runtime decision, the IdP prompts the authorized party interactively during the federation transaction. The authorized party provides consent to release an authentication assertion and specific attributes to the RP. The IdP **SHALL** provide the authorized party with explicit notice and prompt them for positive confirmation before any attributes about the subscriber are transmitted to the RP. At a minimum, the notice **SHOULD** be provided by the party in the position to provide the most effective notice and obtain confirmation, consistent with Sec. 7.2. The IdP **SHALL** disclose which attributes will be released to the RP if the transaction is approved. If the federation protocol in use allows for optional or selective attribute disclosure at runtime, the authorized party **SHALL** be given the option to decide whether to transmit specific attributes to the RP without terminating the federation transaction entirely.

If the authorized party is the subscriber, the IdP **SHALL** provide mechanisms for the subscriber to view the attribute values and derived attribute values to be sent to the RP. To mitigate the risk of unauthorized exposure of sensitive information (e.g., shoulder surfing), the IdP **SHALL**, by default, mask sensitive information displayed to the subscriber. For more details on masking, see Sec. 8 on usability considerations.

An IdP **MAY** employ mechanisms to remember and re-transmit the same set of attributes to the same RP, remembering the authorized party’s decision. This mechanism is associated with the subscriber account as managed by the IdP. If such a mechanism is provided, the IdP **SHALL** allow the authorized party to revoke such remembered access at a future time.

4.6.2. RP-Controlled Decisions

4.6.2.1. RP Allowlists of IdPs

RPs **MAY** establish allowlists of IdPs from which the RP will accept authentication and attributes without a runtime decision from the subscriber to use the IdP. In practice, many RPs interface with only a single IdP, and this IdP is allowlisted as the only possible entry for that RP. When placing an IdP in its allowlist, the RP **SHALL** confirm that the IdP abides by the terms of the trust agreement. Note that this confirmation can be facilitated by a federation authority or be undertaken directly by the RP.

RP allowlists **SHALL** uniquely identify IdPs through the means of domain names, cryptographic keys, or other identifiers applicable to the federation protocol in use.

RP allowlist entries **MAY** be applied based on aspects of the subscriber account (such as the xALs required for the transaction). For example, an RP could use a runtime decision for FAL1 transactions but require an allowlisted IdP for FAL3 transactions.

4.6.2.2. RP Blocklists of IdPs

RPs **MAY** also establish blocklists of IdPs that the RP will not accept authentication or attributes from, even when requested by the subscriber. A blocklisted IdP can be otherwise in a valid trust agreement with the RP, for example if both are under the same federation authority.

RP blocklists **SHALL** uniquely identify IdPs through the means of domain names, cryptographic keys, or other identifiers applicable to the federation protocol in use.

4.6.2.3. RP Runtime Decisions

Every IdP that is in a trust agreement with an RP but not on an allowlist with that RP **SHALL** be governed by a default policy in which runtime authorization decisions will be made by the authorized party indicated in the trust agreement. In this mode, the authorized party is prompted by the RP to select or enter which IdP to contact for authentication on behalf of the subscriber. This process can be facilitated through the use of a discovery mechanism allowing the subscriber to enter a human-facing identifier such as an email address. This process allows the RP to programmatically select the appropriate IdP for that identifier. Since the runtime decision occurs during the federation transaction, the authorized party is generally a person and, in most circumstances, is the subscriber.

The RP **MAY** employ mechanisms to remember the authorized party's decision to use a given IdP. Since this mechanism is employed prior to authentication at the RP, the manner in which the RP provides this mechanism (e.g., a browser cookie outside the authenticated session) is separate from the RP subscriber account described in Sec. 3.10.1. If such a mechanism is provided, the RP **SHALL** allow the authorized party to revoke such remembered options at a future time.

4.6.3. Provisioning Models for RP subscriber accounts

The lifecycle of the provisioning process for an RP subscriber account varies depending on factors including the trust agreement discussed in Sec. 3.4 and the deployment pattern of the IdP and RP. However, in all cases, the RP subscriber account **SHALL** be provisioned at the RP prior to the establishment of an authenticated session at the RP in one of the following ways:

Just-In-Time Provisioning

An RP subscriber account is created automatically the first time the RP receives an assertion with an unknown federated identifier from an IdP. Any identity attributes learned during the federation process, either within the assertion or through an identity API as discussed in Sec. 3.11.3, **MAY** be associated with the RP subscriber account. Accounts provisioned in this way are bound to the federated identifier in the assertion used to provision them. This is the most common form of provisioning in federation systems, as it requires the least coordination between the RP and IdP. However, in such systems, the RP **SHALL** be responsible for managing any cached attributes it might have. See Fig. 7.

Pre-provisioning

An RP subscriber account is created by the IdP pushing the attributes to the RP or the RP pulling attributes from the IdP. Pre-provisioning of accounts generally occurs in bulk through a provisioning API as discussed in Sec. 4.6.5, as the provisioning occurs prior to the represented subscribers authenticating through a federation transaction. Pre-provisioned accounts **SHALL** be bound to a federated identifier at the time of provisioning. Any time a particular federated identifier is seen by the RP, the associated account can be logged in as a result. This form of provisioning requires infrastructure and planning on the part of the IdP and RP, but these processes can be facilitated by automated protocols. Additionally, the IdP and RP must keep the set of provisioned accounts synchronized over time as discussed in Sec. 4.6.4. See Fig. 8.

In this model, the RP also receives attributes about subscribers who have not yet interacted with the RP (and who may never do so). This is in contrast to other models, where the RP receives information only about the subset of subscribers that use the RP, and then only after the subscriber uses the RP for the first time. The privacy considerations of the RP having access to this information prior to a federation transaction **SHALL** be accounted for in the trust agreement.

Ephemeral

An RP subscriber account is created when processing the assertion, but then the RP subscriber account is terminated when the authenticated session ends. This process is similar to a just-in-time provisioning, but the RP keeps no long-term record of the account when the session is complete, in accordance with Sec. 3.10.3. This form of provisioning is useful for RPs that fully externalize access rights to the IdP, allowing the RP to be more simplified with less internal state. However, this pattern is not common because even the simplest RPs tend to have a need to track state within the application or at least keep a record of actions associated with the federated identifier. See Fig. 9.

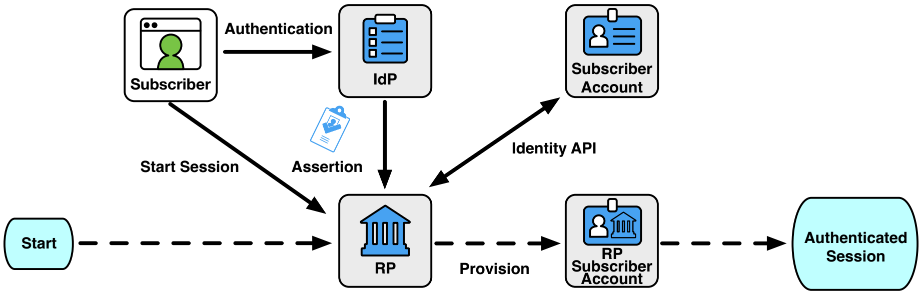

Fig. 7. Just-In-Time Provisioning

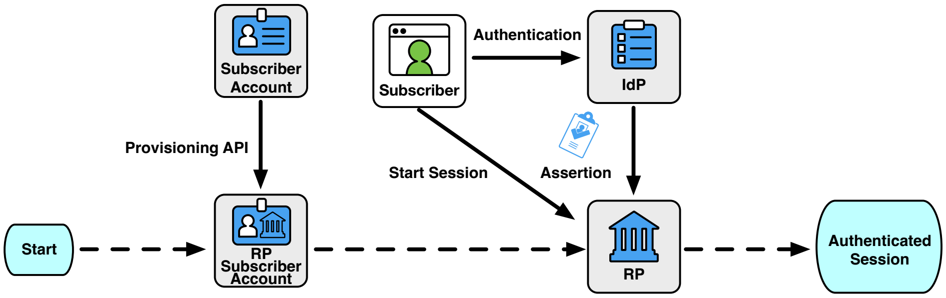

Fig. 8. Pre-Provisioning

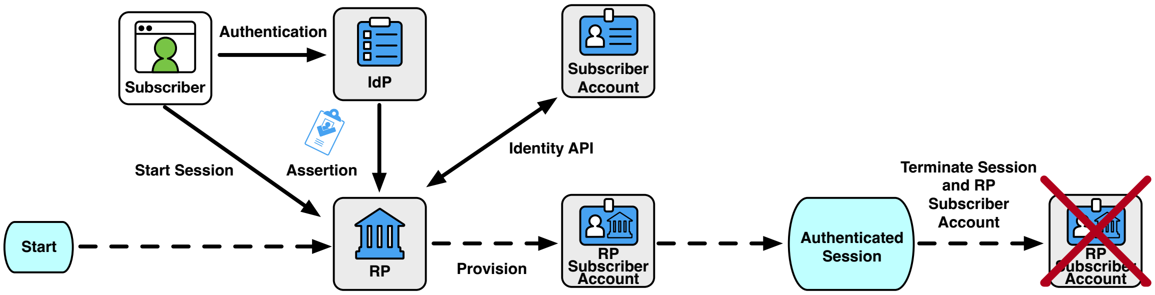

Fig. 9. Ephemeral Provisioning

Other

Other RP subscriber account provisioning models are possible but the details of such models are outside the scope of these guidelines. The details of any alternative provisioning model **SHALL** be included in the privacy risk assessments of the IdP and RP.

All organizations **SHALL** document their provisioning models as part of their trust agreement.

4.6.4. Attribute Synchronization

In a federated process, the IdP and RP each have their own stores of identity attributes associated with the subscriber account. The IdP has a direct view of the subscriber account's attributes, but the RP subscriber account is derived from a subset of those attributes that are presented during the federation transaction. Therefore, it is possible for the IdP's and RP's attribute stores to diverge from each other over time.

From the RP's perspective, the IdP is the trusted source for any attributes that the IdP asserts as being associated with the subscriber account at the IdP. The provenance of the IdP's attributes, and their validation process, is stipulated in the trust agreement.

However, the RP **MAY** additionally collect, and optionally verify, other attributes to associate with the RP subscriber account, as discussed in Sec. 4.6.6. Sometimes, these attributes can even override what is asserted by the IdP. For example, if an IdP asserts a full display name for the subscriber, the RP can allow the subscriber to provide an alternative preferred name for use at the RP.

The IdP **SHOULD** signal downstream RPs when the attributes of a subscriber account available to the RP have been updated, and the RP **MAY** respond to this signal by updating the attributes in the RP subscriber account. This synchronization can be accomplished using shared signaling as described in Sec. 4.8, through a provisioning API as described in Sec. 4.6.5, or by providing a signal in the assertion (e.g., a timestamp indicating when relevant attributes were last updated) allowing the RP to determine that its cache is out of date. If the RP is granted access to an identity API as in Sec. 3.11.3, the IdP **SHOULD** allow the RP access to the API for sufficient time to perform synchronization operations after the federation transaction has concluded. For example, if the assertion is valid for five minutes, access to the identity API could be valid for 30 minutes to allow the RP to fetch and update attributes out of band.

The IdP **SHOULD** signal downstream RPs when a subscriber account is terminated, or when the subscriber account's access to an RP is revoked. This can be accomplished using shared signaling as described in Sec. 4.8 or through a provisioning API as described in Sec. 4.6.5. Upon receiving such a signal, the RP **SHALL** process the RP subscriber account as stipulated in the trust agreement. If the RP subscriber account is terminated, the RP **SHALL** remove all personal information associated with the RP subscriber account, in accordance with Sec. 3.10.3. If the reason for termination is suspicious or fraudulent activity, the IdP **SHALL** include this reason in its signal to the RP to allow the RP to review the account's activity at the RP for suspicious activity, if specified in the trust agreement with that RP.

4.6.5. Provisioning APIs

As part of some proactive forms of provisioning, the RP can be given access to subscriber attributes through a general-purpose identity API known as a *provisioning API*. This type of API allows an IdP to push attributes for a range of subscriber accounts, and sometimes allows an RP to query the attributes of these subscriber accounts directly. Since access to the API is granted outside the context of a federation transaction, access to the provisioning API for a given subscriber does not indicate to the RP that a given subscriber has been authenticated.

The attributes in the provisioning API available to a given RP **SHALL** be limited to only those necessary for the RP to perform its functions, including any audit and security purposes as discussed in Sec. 3.9.1. As part of establishing the trust agreement, the IdP **SHALL** document when an RP is given access to a provisioning API including at least the following:

- the purpose for the access using the provisioning model;
- the set of attributes made available to the RP;
- whether the API functions as a push to the RP, a pull from the RP, or both; and
- the population of subscribers whose attributes are made available to the RP.

Access to the provisioning API **SHALL** occur over a mutually authenticated protected channel. The exact means of authentication varies depending on the specifics of the API and whether it is a push model (where the IdP connects to the RP) or a pull model (where the RP connects to the IdP).

A provisioning API **SHALL NOT** be made available under a subscriber-driven trust agreement. The IdP **SHALL NOT** make a provisioning API available to any RP outside of an established trust agreement. The IdP **SHALL** provide access to a provisioning API only as part of a federated identity relationship with an RP to facilitate federation transactions with that RP and related functions such as signaling revocation of the subscriber account. The IdP **SHALL** revoke an RP's access to the provisioning API once access is no longer required by the RP for its functioning purposes or when the trust agreement is terminated.

Any provisioning API provided to the RP **SHALL** be under the control and jurisdiction of the IdP. External attribute providers **MAY** be used as information sources by the IdP to provide attributes through this provisioning API, but the IdP is responsible for the content and accuracy of the information provided by the referenced attribute providers.

When a provisioning API is in use, the IdP **SHALL** signal to the RP when a subscriber account has been terminated. When receiving such a signal, the RP **SHALL** remove the binding of the federated identifier from the account and **SHALL** terminate the account if necessary (e.g., there are no other federated identifiers linked to this account or the trust agreement dictates such an action). The RP **SHALL** remove all personal information sourced from the provisioning API in accordance with Sec. 3.10.3.

4.6.6. Collection of Additional Attributes by the RP

The RP **MAY** collect and maintain additional attributes from the subscriber beyond those provided by the IdP. For example, the RP could collect a preferred display name directly from the subscriber that is not provided by the IdP. The RP could also have a separate agreement with an attribute provider that gives the RP access to an identity API not associated with the IdP. For example, the RP could receive a state license number from the IdP, but use a separate attribute verification API to check if a particular license number is currently valid. The assertion from the IdP binds the license to the subscriber, but the attribute verification API provides additional information beyond what the IdP can share or be authoritative for.

These attributes are governed separately from the trust agreement since they are collected by the RP outside of a federation transaction. All attributes associated with an RP subscriber account, regardless of their source, **SHALL** be removed when the RP subscriber account is terminated, in accordance with Sec. 3.10.3.

The RP **SHALL** disclose to the subscriber the purpose for collection of any additional attributes. These attributes **SHALL** be used solely for the stated purposes of the RP's functionality and **SHALL NOT** have any secondary use, including communication of said attributes to other parties.

The RP **SHALL** provide an effective means of redress for the subscriber to update and remove these additionally-collected attributes from the RP subscriber account. See Sec. 3.4.3 for additional requirements and considerations for redress mechanisms.

The following requirement applies to federal agencies, regardless of whether they operate their own identity service or use an external CSP as part of their identity service:

- An RP **SHALL** disclose any additional attributes collected, and their use, as part of its System of Records Notice (SORN)

4.6.7. Time-based Removal of RP Subscriber Accounts

If an RP is using a just-in-time provisioning mechanism, the RP only learns of the existence of a subscriber account when that account is first used at the RP. If the IdP does not inform the RP of terminated subscriber accounts using shared signaling as described in Sec 4.8, an RP could accumulate RP subscriber accounts that are no longer accessible from the IdP. This poses a risk to the RP for holding personal information in the RP subscriber accounts. In such circumstances, the RP **MAY** employ a time-based mechanism to identify RP subscriber accounts for termination that have not been accessed after a period of time tailored to the usage patterns of the application. For example, an RP that is usually accessed on a weekly basis could set a timeout of 120 days since last access at the RP to mark the RP subscriber account for termination. An RP that expects longer gaps between access, such as a service used annually, should have a much longer time frame, such as five years.

When processing such an inactive account, the RP **SHALL** provide sufficient notice to the subscriber, about the pending termination of the account and provide the subscriber with an option to re-activate the account prior to its scheduled termination. Upon termination, the RP **SHALL** remove all personal information associated with the RP subscriber account, in accordance with Sec. 3.10.3.

4.7. Reauthentication and Session Requirements in Federated Environments

In a federated environment, the RP manages its sessions separately from any sessions at the IdP. The assertion is related to both sessions but its validity period is ultimately independent of them.

As shown in Fig. 10, an assertion is created during an authenticated session at the IdP, and processing an assertion creates an authenticated session at the RP. The validity time window of an assertion is used to manage the RP's processing of the assertion but does not indicate the lifetime of the authenticated session at the IdP or the RP. If a request comes to the IdP for a new federation transaction while the subscriber's session is still valid at the IdP, a new and separate assertion would be created with its own validity time window. Similarly, after the RP consumes the assertion, the validity of the RP's session is independent of the validity of the assertion, and in most cases the authenticated session at the RP will far outlive the validity of the assertion. Access granted to an identity API is likewise independent of the validity of the assertion or the lifetime of the authenticated session at the RP.

The IdP ending the subscriber's session at the IdP will not necessarily cause any sessions that subscriber might have at downstream RPs to end as well. The RP and IdP **MAY** communicate end-session events to each other, if supported by the federation protocol or through shared signaling (see Sec. 4.8).

At the time of a federated transaction request, the subscriber could have a pre-existing authenticated session at the IdP which **MAY** be used to generate an assertion to the RP. The IdP **SHALL** communicate to the RP any information the IdP has regarding the time of the subscriber's latest authentication event at the IdP, and the RP **MAY** use this information in making authorization and access decisions. Depending on the capabilities of the federation protocol in use, the IdP **SHOULD** allow the RP to request that the subscriber provide a fresh authentication at the IdP instead of using the existing

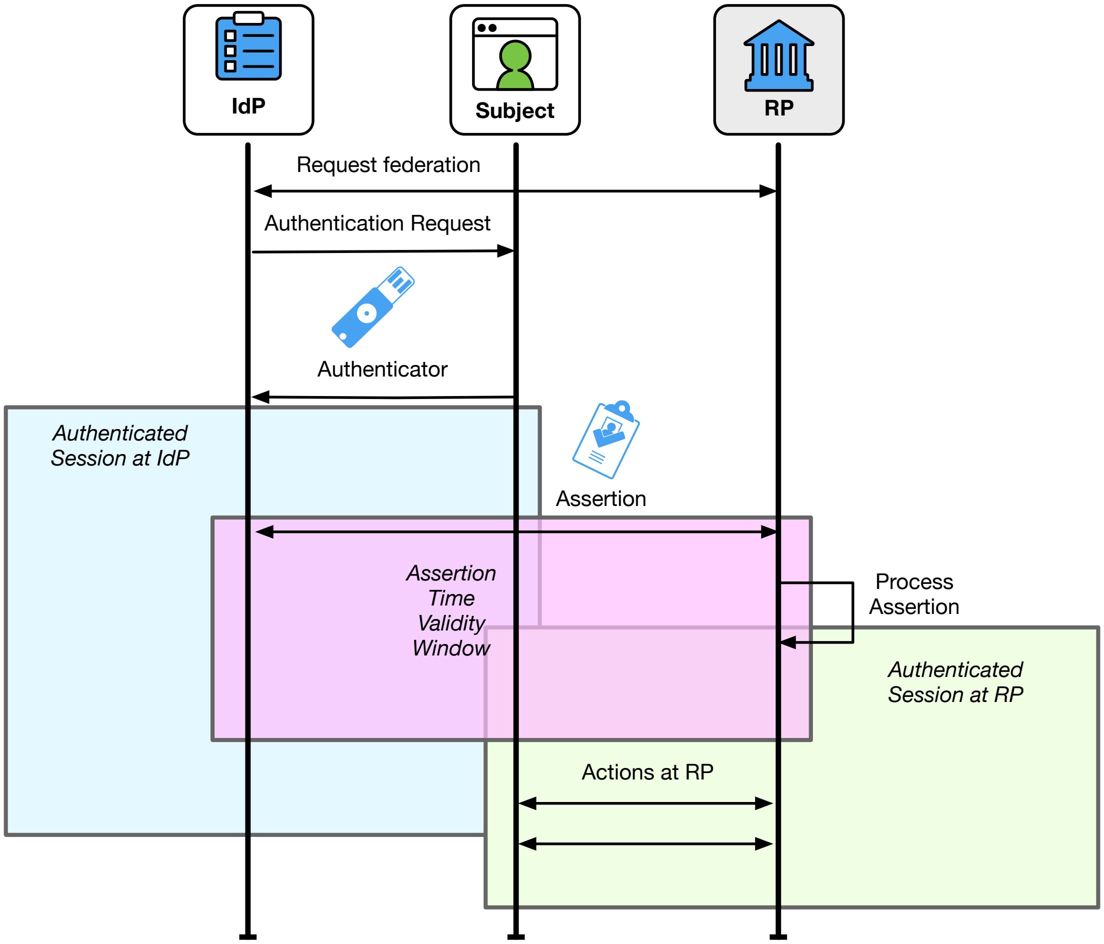

Fig. 10. Session Lifetimes

session at the IdP. For example, suppose the subscriber authenticates at the IdP for one transaction. Then, 30 min later, the subscriber starts a federation transaction at the RP. Depending on xAL requirements, the subscriber's existing session at the IdP can be used to avoid prompting the subscriber for their authenticators. The resulting assertion to the RP will indicate that the last time the subscriber had authenticated to the RP was 30 min in the past. The RP can then use this information to determine whether this is reasonable for the RP's needs, and, if possible within the federation protocol, request the IdP to prompt the subscriber for a fresh authentication event instead.

An RP requiring authentication through a federation protocol **SHALL** specify the maximum acceptable authentication age to the IdP, either through the federation protocol (if possible) or through the terms of the trust agreement. The authentication age represents the time since the last authentication event in the subscriber's session at the IdP, and the IdP **SHALL** reauthenticate the subscriber if they have not been authenticated within that time period. The IdP **SHALL** communicate the authentication event time to the RP to allow the RP to decide if the assertion is sufficient for authentication at the RP and to determine the time for the next reauthentication event.

If an RP is granted access to an identity API at the same time the RP receives an assertion, the lifetime of the access to the identity API is independent from the lifetime of the assertion. As a consequence, the RP's ability to successfully fetch additional attributes through an identity API **SHALL NOT** be used to establish a session at the RP. Likewise, inability to access an identity API **SHOULD NOT** be used to end the session at the RP.

When the RP is granted access to the identity API, the RP is often also granted access to other APIs at the same time, such as granting access to a subscriber's calendar and data storage while also logging in. It is common for this access to be valid long after the assertion has expired and possibly after the session with the RP has ended, allowing the RP to access these non-identity APIs on the subscriber's behalf while the subscriber is no longer present at the RP. Providing access to non-identity APIs is outside the scope of these guidelines.

The RP **MAY** terminate its authenticated session with the subscriber or restrict access to the RP's functions if the assertion, authentication event, or attributes do not meet the RP's requirements. For example, if an RP is configured to allow access to certain high-risk functionality only if the federation transaction was at FAL3, but the incoming assertion only meets the requirements for FAL2, the RP could decide to deny access to the high-risk functionality while allowing access to lower-risk functionality, or the RP could choose to terminate the session entirely.

See [SP800-63B] Sec. 5 for more information about session management requirements that apply to both IdPs and RPs.

4.8. Shared Signaling

In some environments, it is useful for the IdP and RP to send information to each other outside of the federation transaction. These signals can communicate important changes in state between parties that would not be otherwise known. The use of any shared signaling **SHALL** be documented in the trust agreement between the IdP and RP. Signaling from the IdP to the RP **SHALL** require an apriori trust agreement. Signaling from the RP to the IdP **MAY** be used in both apriori and subscriber-driven trust agreements.

Any use of shared signaling **SHALL** be documented and made available to the authorized party stipulated by the trust agreement. This documentation **SHALL** include the events under which a signal is sent, the information included in such a signal (including any attribute information), and any additional parameters sent with the signal. The use of shared signaling **SHALL** be subject to privacy review under the trust agreement.

The IdP **SHOULD** send a signal regarding the following changes to the subscriber account:

- The account has been terminated.
- The account is suspected of being compromised.
- Attributes of the account, including identifiers other than the federated identifier (such as email address or certificate common name), have changed.
- The possible range of IAL, AAL, or FAL for the account has changed.

If the RP receives a signal that an RP subscriber account is suspected of compromise, the RP **SHOULD** review actions taken by that account at the RP for suspicious activity.

The RP **SHOULD** send a signal regarding the following changes to the RP subscriber account:

- The account has been terminated.
- The account is suspected of being compromised.
- A bound authenticator is added by the RP.
- A bound authenticator is removed by the RP.

If the IdP receives a signal that a subscriber account is suspected of compromise, the IdP **SHALL** review actions taken by that account at the IdP for suspicious activity. If suspicious activity is confirmed at the IdP, the IdP **SHALL** signal any additional RPs the subscriber account was used for during the suspected time frame.

Additional signals from both the IdP and RP **MAY** be allowed subject to privacy and security review as part of the trust agreement.

4.9. Assertion Contents

An assertion is a packaged set of attribute values or derived attribute values about or associated with an authenticated subscriber that is passed from the IdP to the RP in a federated identity system. Assertions contain a variety of information, including: assertion metadata, attribute values and derived attribute values about the subscriber, information about the subscriber’s authentication at the IdP, and other information that the RP can leverage (e.g., restrictions and validity time window). While the assertion’s primary function is to authenticate the user to an RP, the information conveyed in the assertion can be used by the RP for a number of use cases — for example, authorization or personalization of a website. These guidelines do not restrict RP use cases nor the type of protocol or data payload used to federate an identity, provided that the chosen solution meets all mandatory requirements contained herein.

Assertions **SHALL** represent a discrete authentication event of the subscriber at the IdP and **SHALL** be processed as a discrete authentication event at the RP.

All assertions **SHALL** include the following attributes:

1. Subject identifier: An identifier for the party to which the assertion applies (i.e., the subscriber).
2. Issuer identifier: An identifier for the issuer of the assertion (i.e., the IdP).
3. Audience identifier: An identifier for the party intended to consume the assertion (i.e., the RP). An assertion can contain more than one audience identifier at FAL1.
4. Issuance time: A timestamp indicating when the IdP issued the assertion.
5. Validity time window: A period of time outside of which the assertion **SHALL NOT** be accepted as valid by the RP for the purposes of authenticating the subscriber and starting an authenticated session at the RP. This is usually communicated by means of an expiration timestamp for the assertion in addition to the issuance timestamp.
6. Assertion identifier: A value uniquely identifying this assertion, used to prevent attackers from replaying prior assertions.
7. Authentication time: A timestamp indicating when the IdP last verified the presence of the subscriber at the IdP through a primary authentication event.
8. Nonce: A cryptographic nonce, if one is provided by the RP.
9. Signature: Digital signature or message authentication code (MAC), including key identifier, covering the entire assertion.

All assertions **SHALL** contain sufficient information to determine the following aspects of the federation transaction:

1. The IAL of the subscriber account being represented in the assertion, or an indication that no IAL is asserted.
2. The AAL used when the subscriber authenticated to the IdP, or an indication that no AAL is asserted.
3. The IdP’s intended FAL of the federation process represented by the assertion.

At FAL3, the assertion **SHALL** include one of the following:

- The public key, key identifier, or other identifier for a holder-of-key assertion, or
- An indicator that verification of a bound authenticator is required to process this assertion.

Assertions **MAY** also include additional items, including the following information:

1. Attribute values and derived attribute values: Information about the subscriber.
2. Attribute bundles: Collections of attributes in a signed bundle from the CSP.
3. Attribute metadata: Additional information about one or more subscriber attributes, such as those described in [NISTIR8112].
4. Authentication event: Additional details about the authentication event, such as the class of authenticator used.

The RP **SHALL** validate the assertion by checking that all the following are true:

- *Signature validation*: ensuring that the signature of the assertion is valid and corresponds to a key belonging to the IdP sending the assertion.
- *Issuer verification*: ensuring that the assertion was issued by the IdP the RP expects it to be from.
- *Time validation*: ensuring that the expiration and issue times are within acceptable limits of the current timestamp.
- *Audience restriction*: ensuring that this RP is the intended recipient of the assertion.
- *Nonce*: ensuring that the cryptographic nonce included in the RP's request (if applicable) is included in the presentation.
- *Transaction terms*: ensuring that the IAL, AAL, and FAL represented by the assertion are allowable under the applicable trust agreement.

An RP **SHALL** treat subject identifiers as not inherently globally unique. Instead, the value of the assertion's subject identifier is usually in a namespace under the assertion issuer's control, as discussed in Sec. 3.3. This allows an RP to talk to multiple IdPs without incorrectly conflating subjects from different IdPs.

Assertions **MAY** include additional attributes about the subscriber. Section 3.9 contains privacy requirements for presenting attributes in assertions. The RP **MAY** be given limited access to an identity API as discussed in Sec. 3.11.3, either in the same response as the assertion is received or through some other mechanism. The RP can use this API to fetch additional identity attributes for the subscriber that are not included in the assertion itself.

The assertion's validity time window is the time between its issuance and its expiration. This window needs to be large enough to allow the RP to process the assertion and create a local application session for the subscriber, but should not be longer than necessary for such establishment. Long-lived assertions have a greater risk of being stolen or replayed; a short assertion validity time window mitigates this risk. Assertion validity time windows **SHALL NOT** be used to limit the session at the RP. See Sec. 4.7 for more information.

4.10. Assertion Requests

When the federation transaction is initiated by the RP, the RP's request for an assertion **SHALL** contain:

1. An identifier for the RP
2. A cryptographic nonce, to be returned in the assertion

The RP's request **SHOULD** additionally contain:

1. The set of identity attributes requested by the RP and their purpose of use at the RP; this is a subset of what is allowed by the trust agreement
2. The requirements for the authentication event at the IdP

Note that federation transactions are always initiated by the RP at FAL2 or higher.

4.11. Assertion Presentation

Depending on the specifics of the protocol, the RP and the IdP communicate with each other in two ways, which lends to two different ways in which an assertion can be passed from the IdP to the RP:

- The *back channel*, through a direct connection between the RP and IdP, not involving the subscriber directly; or
- The *front channel*, through a third party using redirects involving the subscriber and the subscriber's browser.

There are tradeoffs with each model, but each requires the proper validation of the assertion. Assertions **MAY** also be proxied to facilitate federation between IdPs and RPs using different presentation methods, as discussed in detail in Sec. 3.2.3.

4.11.1. Back-Channel Presentation

In the *back-channel* presentation model shown in Fig. 11, the subscriber is given an assertion reference to present to the RP, generally through the front channel. The assertion reference itself contains no information about the subscriber and **SHALL** be resistant to tampering and fabrication by an attacker. The RP presents the assertion reference to the IdP to fetch the assertion. How this is achieved varies form one protocol to the next. In the authorization code flow and some forms of the hybrid flow of [OIDC] the assertion (the ID Token) is presented in the back channel in exchange for the assertion reference (the authorization code). In the artifact binding profile of [SAML-Bindings], the SAML assertion is presented in the back channel.

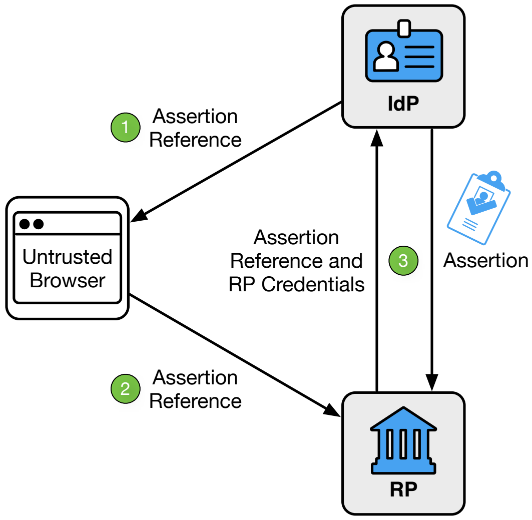

Fig. 11. Back-channel Presentation

As shown in Fig. 11, the back-channel presentation model consists of three steps:

1. The IdP sends an assertion reference to the subscriber through the front channel.
2. The subscriber sends the assertion reference to the RP through the front channel.
3. The RP presents the assertion reference and its RP credentials to the IdP through the back channel. The IdP validates the credentials and returns the assertion.

The assertion reference:

1. **SHALL** be limited to use by a single RP.
2. **SHALL** be single-use.
3. **SHALL** be time limited, and **SHOULD** have a validity time window of no more than five minutes.
4. **SHALL** be presented along with authentication of the RP to the IdP.
5. **SHALL NOT** be predictable or guessable by an attacker.

In this model, the RP directly requests the assertion from the IdP, minimizing chances of interception and manipulation by a third party (including the subscriber themselves). More network transactions are required in the back-channel method, but the information is limited to only those parties that need it. Since an RP is expecting to get an assertion only from the IdP directly as a result of its request, the attack surface is reduced. Consequently, it is more difficult to inject assertions directly into the RP and this presentation method is recommended for FAL2 and above. Since the IdP and RP are already directly connected, the back-channel presentation method facilitates the use of identity APIs, as described in Sec. 3.11.3.

Note that while it is technically possible for an assertion reference (which is single-audience) to result in a multi-audience assertion, this situation is unlikely. For this reason, back-channel presentation is practically limited to use with single-audience assertions.

Conveyance of the assertion reference from the IdP to the subscriber, as well as from the subscriber to the RP, **SHALL** be made over an authenticated protected channel. Conveyance of the assertion reference from the RP to the IdP, as well as the assertion from the IdP to the RP, **SHALL** be made over an authenticated protected channel.

The RP **SHALL** protect itself against injection of manufactured or captured assertion references by the use of cross-site scripting protection, rejecting assertion references outside of the correct stage of a federation transaction, or other accepted techniques discussed in Sec. 3.10.1. When assertion references are presented to the IdP, the IdP **SHALL** verify that the RP presenting the assertion reference is the same RP that made the assertion request resulting in the assertion reference. Examples for this are discussed in Sec 10.12 such as the authorization code flow of [OIDC] with additional security profiles such as [FAPI].

Note that in a federation proxy described in Sec. 3.2.3, the upstream IdP audience restricts the assertion reference and assertion to the proxy, and the proxy restricts any newly-created assertion references or assertions to the downstream RP.

4.11.2. Front-Channel Presentation

In the *front-channel* presentation model shown in Fig. 12, the IdP creates an assertion and sends it to the RP by means of a third party, such as the subscriber’s user agent. In the implicit flow and some forms of the hybrid flow of [OIDC], the assertion (the ID Token) is presented in the front channel. In the SAML Web SSO profile defined in [SAML-WebSSO], the SAML assertion is presented in the front channel.

Front-channel presentation methods expose the assertion to parties other than the IdP and RP, which increases the risk for leakage of PII and other information included in the assertion. Additionally, there is an increased attack surface for the assertion to be captured and replayed by an attacker. As a consequence, it is recommended to not use front-channel presentation when other mechanisms are available.

The RP **SHALL** use the assertion identifier ensure that a given assertion is presented at most once during the assertion’s validity time window.

The RP **SHALL** protect itself against injection of manufactured or captured assertion by the use of cross-site scripting protection, rejecting assertions outside of the correct stage of a federation transaction, or other accepted techniques discussed in Sec. 3.10.1.

Conveyance of the assertion from the IdP to the subscriber, as well as from the subscriber to the RP, **SHALL** be made over an authenticated protected channel.

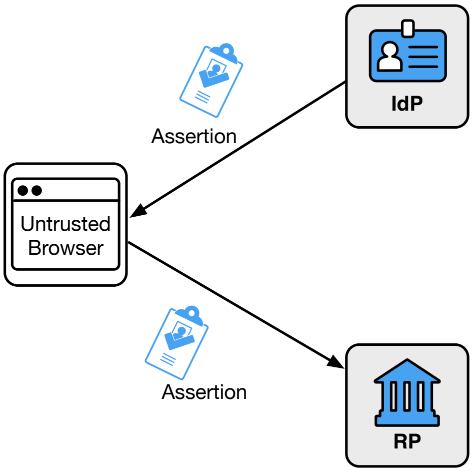

Fig. 12. Front-channel Presentation

With general-purpose IdPs, it is common for front-channel communications to be accomplished using HTTP redirects, where the contents of the assertion are made available as part of an HTTP request URL. Due to the nature of the HTTP ecosystem, these request URLs are sometimes available in unexpected places, such as access logs and browser history. These logs and other artifacts tend to live on long past the federation transaction and are available in other contexts, which increases the attack surface for reading the assertion. As a consequence, an IdP that uses HTTP redirects for front channel presentation of assertions that contain PII **SHALL** encrypt the assertion as discussed in Sec 3.12.3.

5. Subscriber-Controlled Wallets

This section is normative.

When the IdP runs on a device controlled by the subscriber, whether as a digital wallet or as a self-issued identity provider, the IdP is known as a *subscriber-controlled wallet* and the following requirements apply.

Subscriber-controlled wallets **SHALL** require the presentation of an activation factor in order to perform any actions requiring the use of the wallet's signing key, including onboarding of the wallet and release of attributes to an RP.

5.1. Wallet Activation

The subscriber-controlled wallet **SHALL** require presentation of an activation factor from the subscriber for the following actions:

- Providing proof of the signing key to the CSP during the provisioning process
- Signing the assertion for presentation to the RP

The subscriber-controlled wallet **SHOULD** require presentation of an activation factor before any other operations that involve use of the wallet's signing keys. The wallet **MAY** request reissuance of previously-issued attribute bundles without requiring subscriber involvement.

Submission of the activation factor **SHALL** be a separate operation from the unlocking of the host device (e.g., smartphone), although the same activation factor used to unlock the host device **MAY** be used in the activation operation. Agencies **MAY** relax this requirement for subscriber-controlled wallets managed by or on behalf of the CSP (e.g., via mobile device management) that are constrained to have short (agency-determined) inactivity timeouts and device activation factors meeting the above requirements. Additional discussion of activation factors for authenticators is found in Sec. 3.2.10 of [SP800-63B].

5.2. Federation Transaction

A federation transaction with a subscriber-controlled wallet establishes the subscriber's device as an IdP for the subscriber account and creates an authenticated session for the subscriber at the RP. The process is shown in Fig. 13.

A federation transaction with a subscriber-controlled wallet takes place over several steps:

1. The CSP identity proofs the subscriber and creates a subscriber account.
2. The CSP provisions the wallet to the subscriber account, which includes the subscriber verifying an authenticator in their subscriber account.

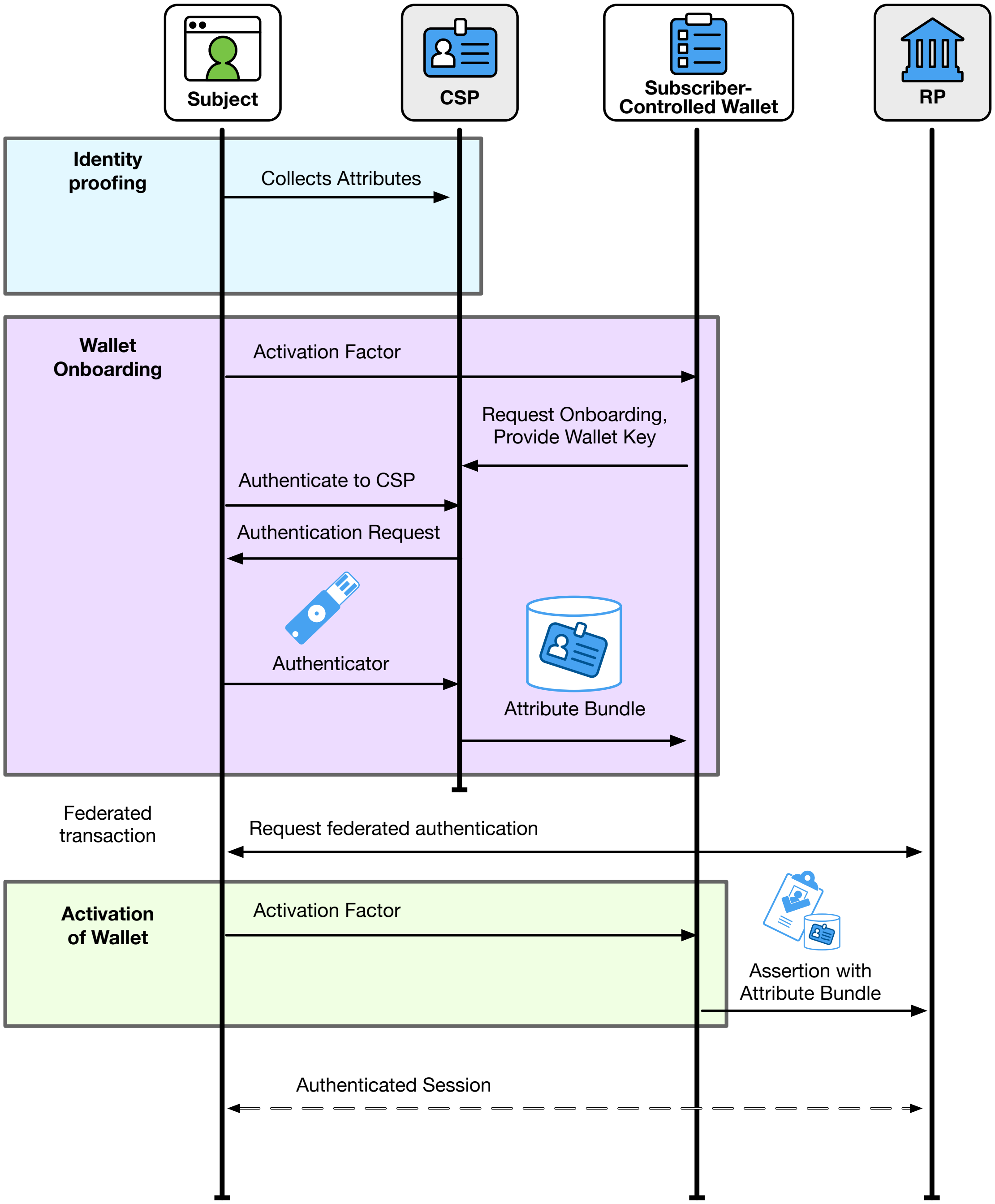

Fig. 13. Subscriber-Controlled Wallet

3. The wallet receives a signed attribute bundle from the CSP, allowing the wallet to act as an IdP.
4. The RP requests a federated authentication from the wallet, usually through subscriber action.
5. The subscriber activates the wallet through an authentication factor.
6. The wallet creates an assertion based on the attribute bundles available to the wallet.
7. The wallet presents the assertion to the RP.
8. The RP validates the assertion.
9. The RP creates an authenticated session for the subscriber.

5.3. Trust Agreements

The trust agreement for a transaction involving a subscriber-controlled wallet **SHALL** be established between the RP and the CSP. The trust agreement **MAY** be facilitated through use of a federation authority, as described in Sec. 3.4.2.

In most cases, the RP does not have a direct trust relationship with the wallet (acting as IdP), but instead trusts the wallet transitively through the wallet's established relationship with the CSP. This relationship can be verified by the means of attribute bundles, as described in Sec. 3.11.1. Even though the wallet is not usually involved in the process of establishing the trust agreement, the trust agreement between the RP and CSP can still be accomplished in either an a priori or subscriber-driven fashion.

The trust agreement **SHALL** include the following

- The set of subscriber attributes the CSP makes available to wallets in attribute bundles
- The set of subscriber attributes the wallet can make available to the RP
- The population of subscriber accounts that the CSP can represent
- The xALs available from the wallet

The release of subscriber attributes **SHALL** be managed using a runtime decision managed by the wallet, as described in Sec. 4.6.1.3. The authorized party **SHALL** be the subscriber.

The following terms **SHALL** be disclosed to the subscriber during the runtime decision:

- The set of subscriber attributes that the RP will request (a subset of the attributes made available)
- The purpose for each attribute requested by the RP

- The xALs required by the RP

Note that all information disclosed to the subscriber needs to be conveyed in a manner that is understandable and actionable, as discussed in Sec. 8.

If FAL3 is allowed within the trust agreement and authenticators other than the wallet itself are allowed for use at FAL3, the trust agreement **SHALL** stipulate the following terms regarding holder-of-key assertions and bound authenticators (see Sec. 3.14 and Sec. 3.15):

- Whether the wallet's presentation is considered sufficient for holder-of-key assertion requirements
- The means by which non-wallet holder-of-key assertions can be verified by the RP (such as a common trusted PKI system)
- The means by which the RP can associate non-wallet holder-of-key assertions with specific RP subscriber accounts (such as attribute-based account resolution or pre-provisioning)
- Whether bound authenticators are supplied by the RP or by the subscriber
- Documentation of the binding ceremony used for any subscriber-provided bound authenticators

5.4. Provisioning the Subscriber-Controlled Wallet

When the CSP provisions the subscriber-controlled wallet, the process **SHALL** include the following steps:

1. The subscriber authenticates to the CSP's provisioning system using one or more authenticators bound to the subscriber account.
2. The subscriber activates the wallet using an activation factor.
3. The wallet proves possession of its signing key to the CSP.
4. The CSP creates one or more attribute bundles that include subscriber attributes and the wallet's signing key (or a reference to that key).
5. The wallet stores the attribute bundle for later presentation to RPs.

The subscriber-controlled wallet **MAY** generate and use a different signing key for each provisioning request with the CSP.

The CSP **SHALL** create a unique attribute bundle for each requesting wallet.

5.4.1. Deprovisioning the Subscriber-Controlled Wallet

The CSP **SHALL** provide a means of deprovisioning a subscriber-controlled wallet. The deprovisioning process is used when the subscriber account is terminated, thereby rendering downstream federation actions invalid, or when the wallet needs to be terminated due to the device being lost, stolen, or compromised.

To accomplish this, the CSP **SHALL** issue attribute bundles with a limited time validity window, **SHALL** issue attribute bundles specific to each wallet. The CSP **SHOULD** provide a means to independently verify the status of attribute bundles (i.e., whether a specific bundle has been revoked by the CSP). If such a service is offered, the service **SHALL** be deployed in a privacy-preserving way such that the CSP is not alerted to the use of a specific attribute bundle at a specific RP.

5.5. Discovery and Registration

To perform a federation transaction with a subscriber-controlled wallet, the RP **SHALL** first determine the attribute bundle singing public key of the CSP through a secure process as stated by the trust agreement. In some systems, this is accomplished by retrieving the CSP's attribute bundle signing public keys from a URL known to be controlled by the CSP. In other systems, the RP is configured manually with the public key of the CSP before being deployed.

The RP learns the identifier and assertion signing public keys of the subscriber-controlled wallet as part of the attribute bundle signed by the CSP, presented in the federation transaction. The RP trusts the CSP's onboarding process of the wallet to provide assurance that the public key being presented can be trusted to present the attribute bundle in question.

The RP also needs to register with the subscriber-controlled wallet. In most cases, this is expected to be a dynamic process in which the RP introduces its properties during the federation transaction. The nature of a subscriber-controlled wallet makes it difficult for any specific RP to pre-register with an instance of the wallet, but this use case can be facilitated through the use of a trusted third party stipulated in the trust agreement. For example, an ecosystem has a centralized service for managing discovery and registration. When an RP joins the ecosystem, it registers itself with the trusted service, downloads the CSP's public keys, and receives an identifier to use with wallets. When the wallet is onboarded by the CSP, the wallet is informed where it can find the list of valid RP identifiers within the ecosystem. When the RP connects to the wallet, the wallet can verify the RP's identifier without the RP having to register itself directly with the wallet. Likewise, the RP can verify the wallet's signing keys by the fact they are presented in an attribute bundle signed by the CSP's public key, which had in turn been retrieved from the trusted third party.

5.6. Authentication and Attribute Disclosure

The decision of whether a federated authentication can occur or attributes may be passed **SHALL** be determined by the subscriber, acting in the role of the authorized party.

The subscriber-controlled wallet **SHOULD** provide a means to selectively disclose a subset of the attributes in the attribute bundle from the CSP.

The CSP **SHALL** provide effective mechanisms for redress of subscriber complaints or problems (e.g., subscriber identifies an inaccurate attribute value, or the need to deprovision a subscriber-controlled wallet). See Sec. 3.4.3 for additional requirements and considerations for redress mechanisms.

5.7. Assertion Requests

When the federation transactions are initiated by the RP, the RP's request for an assertion **SHALL** contain:

1. An identifier for the RP
2. A cryptographic nonce
3. The set of identity attributes requested by the RP and their purpose of use at the RP

Note that federation transactions are always initiated by the RP at FAL2 or higher.

5.8. Assertion Contents

Assertions from a subscriber-controlled wallet **SHALL** contain:

1. A signed attribute bundle from the CSP.
2. Subject identifier: An identifier for the party to which the assertion applies (i.e., the subscriber).
3. Issuer identifier: An identifier for the issuer of the assertion (i.e., the subscriber-controlled wallet).
4. Audience identifier: An identifier for the party intended to consume the assertion (i.e., the RP).
5. Issuance time: A timestamp indicating when the wallet issued the assertion.
6. Validity time window: A period of time outside of which the assertion **SHALL NOT** be accepted as valid by the RP for the purposes of authenticating the subscriber and starting an authenticated session at the RP. This is usually communicated by means of an expiration timestamp for the assertion in addition to the issuance timestamp.

7. Assertion identifier: A value uniquely identifying this assertion, used to prevent attackers from replaying prior assertions.
8. Authentication time: A timestamp indicating when the subscriber last used the wallet's activation factor.
9. Nonce: A cryptographic nonce, if one is provided by the RP.
10. Signature: Digital signature using asymmetric cryptography, covering the entire assertion.

All assertions **SHALL** contain sufficient information to determine the following aspects of the federation transaction:

1. The IAL of the subscriber account being represented in the assertion, or an indication that no IAL is asserted.
2. The wallet's intended FAL of the federation process represented by the assertion.

At FAL3, the assertion **SHALL** include one of the following:

- The public key, key identifier, or other identifier for a holder-of-key assertion. This **MAY** be the same key that the subscriber-controlled wallet uses to sign the assertion.
- An indicator that verification of a bound authenticator is required to process this assertion.

The signed attribute bundle from the CSP **SHALL** contain:

1. A public key or key identifier for the key used by the subscriber-controlled wallet to sign the assertion
2. Issuance time: A timestamp indicating when the CSP issued the attribute bundle.
3. Validity time window: A period of time outside of which the attribute bundle **SHALL NOT** be accepted as valid by the RP for the purposes of authenticating the subscriber and starting an authenticated session at the RP. This is usually communicated by means of an expiration timestamp for the assertion in addition to the issuance timestamp.
4. IAL: Indicator of the IAL of the subscriber account being represented in the attribute bundle, or an indication that no IAL is asserted.
5. Signature: Digital signature using asymmetric cryptography, covering the entire attribute bundle.

Additional identity attributes and derived attribute values **MAY** be included in the attribute bundle. These attributes **SHOULD** be made available using a selective disclosure method, whereby the subscriber can, through their wallet software, determine which parts of the bundle to disclose to the RP.

Identity attributes in the assertion but outside of a signed attribute bundle **SHALL** be considered self-asserted. The RP **MAY** validate these additional attributes out of band.

Subscriber-controlled wallets **SHOULD** use non-exportable key storage as discussed in Sec. 3.5.2.

5.9. Assertion Presentation

Assertions **SHALL** be presented to the RP through an authenticated protected channel.

The presentation **SHALL** include the cryptographic nonce from the RP's request, if present. The RP **SHALL** verify the nonce in accordance with the federation protocol.

If the assertion contains PII, and the presentation mechanism passes the assertion through a component other than the wallet or RP, the assertion **SHOULD** be encrypted.

The RP **SHALL** protect itself against injection of manufactured or captured assertions by the use of cross-site scripting protection, rejecting assertions outside of the correct stage of a federation transaction, or other accepted techniques discussed in Sec. 3.10.1. When possible, the IdP **SHOULD** use platform APIs instead of HTTP redirects when delivering an assertion to the RP.

Since assertions from a subscriber-controlled wallet always contain a reference to the wallet's signing key inside the signed attribute bundle from the CSP, assertions from subscriber-controlled wallets **MAY** be used as holder-of-key assertions to reach FAL3, as long as all other requirements in these guidelines are met. For additional requirements for holder-of-key assertions, see Sec. 3.14.

5.10. Assertion Validation

The RP **SHALL** validate the signature on all signed attribute bundles in the assertion, using the cryptographic key from the CSP issuing the signed attribute bundle. The RP **SHALL** validate the signature of the assertion using the identified cryptographic key in the signed attribute bundle.

The RP **SHALL** validate the assertion by checking that all the following are true:

- *Issuer verification*: ensuring that the assertion was issued by the wallet the RP expects it to be from.
- *Time validation*: ensuring that the expiration and issue times are within acceptable limits of the current timestamp.

- *Audience restriction*: ensuring that this RP is the intended recipient of the assertion.
- *Nonce*: ensuring that the cryptographic nonce included in the RP's request is included in the presentation.
- *Transaction terms*: ensuring that the IAL, AAL, and FAL represented by the assertion are allowable under the applicable trust agreement.

Additionally, the issuer **MAY** make available an online mechanism to determine the validity of a given attribute bundle, such as a status list queryable by the RP.

5.11. RP Subscriber Accounts

RP subscriber accounts **SHALL** be managed using a just-in-time or ephemeral provisioning model only (see Sec. 4.6.3). In each of these cases, the RP creates the RP subscriber account and associates it with the federated identifier only after successful validation of the assertion from the wallet.

The RP **SHALL** disclose its practices for management of subscriber information as part of the trust agreement. The RP **SHALL** provide effective means of redress to the subscriber for correcting and removing information from the RP subscriber account. See Sec. 3.4.3 for additional requirements and considerations for redress mechanisms.

6. Security

This section is informative.

Since the federated authentication process involves coordination between multiple components, including the CSP, IdP, and RP, there are additional opportunities for attackers to compromise federated identity transactions and additional ramifications for successful attacks. This section summarizes many of the attacks and mitigations applicable to federation.

6.1. Federation Threats

As in non-federated authentication, attackers' motivations are typically to gain access (or a greater level of access) to a resource or service provided by an RP. Attackers may also attempt to impersonate a subscriber. Rogue or compromised IdPs, RPs, user agents (e.g., browsers), and parties outside of a typical federation transaction are potential attackers. To accomplish their attack, they might intercept or modify assertions and assertion references. Furthermore, two or more entities may attempt to subvert federation protocols by directly compromising the integrity or confidentiality of the assertion data. For the purpose of these types of threats, any authorized parties who attempt to exceed their privileges are considered attackers.

In federated systems, successful attacks on the IdP can propagate through to the RPs that rely on that IdP for identity and security information. As a consequence, an attack against the IdP targeting one agency's RP could potentially proliferate to another agency's RP. Additionally, since a single subscriber account is made available to multiple RPs in a federated system, there are potential limitations on the tailoring to proofing strategies and the visibility into the proofing process that an IdP can offer to different RPs. However, these terms can vary in the trust agreements with each RP, if the IdP is able to support different use cases for different subscriber account populations. Furthermore, while the IdP can disclose different attributes to each RP, the subscriber account will need to contain the union of all attributes available to all RPs. This practice limits the damage of attacks against RPs but in turn makes the IdP a more compelling target for attackers.

Table 2. Federation Threats

Federation Threats/Attacks	Description	Examples
Assertion Manufacture or Modification	The attacker generates a false assertion	Compromised IdP asserts identity of a claimant who has not properly authenticated
	The attacker modifies an existing assertion	Compromised proxy that changes AAL of an authentication assertion
Assertion Disclosure	Assertion visible to third party	Network monitoring reveals subscriber address of record to an outside party
Assertion Repudiation by the IdP	IdP later claims not to have signed transaction	User engages in fraudulent credit card transaction at RP, IdP claims not to have logged them in
Assertion Repudiation by the Subscriber	Subscriber claims not to have performed transaction	User agreement (e.g., contract) cannot be enforced
Assertion Redirect	Assertion can be used in unintended context	Compromised user agent passes assertion to attacker who uses it elsewhere
Assertion Reuse	Assertion can be used more than once with same RP	Intercepted assertion used by attacker to authenticate their own session
Assertion Substitution	Attacker uses an assertion intended for a different subscriber	Session hijacking attack between IdP and RP

6.2. Federation Threat Mitigation Strategies

Mechanisms that assist in mitigating the above threats are identified in Table 3.

Table 3. Mitigating Federation Threats

Federation Threat/Attack	**Threat Mitigation Mechanisms**	**Normative Reference(s)**
Assertion Manufacture or Modification	Cryptographically sign the assertion at IdP and verify at RP	3.5, 3.12.2
	Send assertion over an authenticated protected channel authenticating the IdP	4.11
	Include a non-guessable random identifier in the assertion	3.12.1
Assertion Disclosure	Send assertion over an authenticated protected channel authenticating the RP	4.9, 5.8
	Encrypt assertion for a specific RP (may be accomplished by use of a mutually authenticated protected channel)	3.12.3
Assertion Repudiation by the IdP	Cryptographically sign the assertion at the IdP with a key that supports non-repudiation; verify signature at RP	3.12.2
Assertion Repudiation by the Subscriber	Issue holder-of-key assertions or assertions with bound authenticators; proof of possession of authenticator verifies subscriber's participation to the RP	3.14 3.15
Assertion Redirect	Include identity of the RP ("audience") for which the assertion is issued in its signed content; RP verifies that they are intended recipient	
Assertion Reuse	Include an issuance timestamp with short validity period in the signed content of the assertion; RP verifies validity	4.9, 5.8
	RP keeps track of assertions consumed within a configurable time window to ensure that a given assertion is not used more than once.	3.12.1
Assertion Substitution	Ensure that assertions contain a reference to the assertion request or some other nonce that was cryptographically bound to the request by the RP	4.9, 5.8
	Send assertions in the same authenticated protected channel as the request, such as in the back-channel model	4.11.1

7. Privacy Considerations

This section is informative.

7.1. Minimizing Tracking and Profiling

Federation offers numerous benefits to RPs and subscribers, but it requires subscribers to have trust in the federation participants. Sec. 3 and Sec. 3.3.1 cover a number of technical requirements, the objective of which is to minimize privacy risks arising from increased capabilities to track and profile subscribers. For example, a subscriber using the same IdP to authenticate to multiple RPs allows the IdP to build a profile of subscriber transactions that would not have existed absent federation. The availability of such data makes it vulnerable to uses that may not be anticipated or desired by the subscriber and may inhibit subscriber adoption of federated services.

Section 3.9 requires IdPs to use measures to maintain the objectives of predictability (enabling reliable assumptions by individuals, owners, and operators about PII and its processing by an information system) and manageability (providing the capability for granular administration of PII, including alteration, deletion, and selective disclosure) commensurate with privacy risks that can arise from the processing of attributes for purposes other than those listed in Sec. 3.9.1.

IdPs may have various business purposes for processing attributes, including providing non-identity services to subscribers. However, processing attributes for different purposes from the original collection purpose can create privacy risks when individuals are not expecting or comfortable with the additional processing. IdPs can determine appropriate measures commensurate with the privacy risk arising from the additional processing. For example, absent applicable law, regulation, or policy, it may not be necessary to get consent when processing attributes to provide non-identity services requested by subscribers, although notices may help subscribers maintain reliable assumptions about the processing (e.g., predictability). Other processing of attributes may carry different privacy risks that call for obtaining consent or allowing subscribers more control over the use or disclosure of specific attributes (manageability). Subscriber consent needs to be meaningful; therefore, when IdPs do use consent measures, they cannot make acceptance by the subscriber of additional uses a condition of providing the identity service.

When holder-of-key assertions are used at FAL3, the same authenticator is usually used at both the IdP and RP. With authenticators that can fulfill this technical requirement, it is likely that the same authenticator would further be used at multiple RPs. Furthermore, an unrelated RP could use the same authenticator for direct authentication. All such RPs would potentially be able to collude and disclose the use of the same authenticator across all parties in order to effect tracking of the subscriber through the network. This is true even if per-provider identifiers are used, as the bound authenticator is recognizable apart from the assertion. Additionally, many authenticators suitable for holder-of-key assertions contain identity attributes which are sent apart from the assertion or an identity API. These additional attributes have to be covered by the privacy risk assessment.

Consult the SAOP if there are questions about whether the proposed processing falls outside the scope of the permitted processing or the appropriate privacy risk mitigation measures.

Section 3.9 also encourages the use of technical measures to provide disassociability (enabling the processing of PII or events without association to individuals or devices beyond the operational requirements of the system) and prevent subscriber activity tracking and profiling [NISTIR8062]. Technical measures, such as those outlined in Sec. 3.2.3 for proxied federation and Sec. 3.3.1 for pairwise pseudonymous identifiers, can increase the effectiveness of policies by making it more difficult to track or profile subscribers beyond operational requirements. However, even these measures have their limitations and tracking can still occur based on subscriber attributes, statistical demographics, and other kinds of information shared between the IdP and RP.

In some use cases, especially at higher xALs, tracking the real-world identity of the subscriber is expected as a means of securing the system. It is the responsibility of the IdP and RP to inform and educate the subscriber about which pieces of information are transmitted, and allow the subscriber to review this information.

7.2. Notice and Consent

To build subscriber trust in federation, subscribers need to be able to develop reliable assumptions about how their information is being processed. For instance, it can be helpful for subscribers to understand what information will be transmitted, which attributes for the transaction are required versus optional, and to have the ability to decide whether to transmit optional attributes to the RP. Accordingly, Sec. 3.4 requires that positive confirmation be obtained from the authorized party before any attributes about the subscriber are transmitted to any RP.

In determining when a set of RPs should share a shared pairwise pseudonymous identifier as in Sec. 3.3.1.3, the trust agreement considers the subscriber’s understanding of such a grouping of RPs and provides a means for effective notice to the subscriber in assisting such understanding. An effective notice will take into account user experience design standards and research, as well as an assessment of privacy risks that may arise from the information processing. There are various factors to be considered, including the reliability of the assumptions subscribers may have about the processing and the role of different entities involved in federation. However, a link to a complex, legalistic privacy policy or general terms and conditions that a substantial number of subscribers do not read or understand is never an effective notice.

Sec. 3.4 does not specify which party should provide the notice. In some cases, a party in a federation may not have a direct connection to the subscriber in order to provide notice and obtain consent. Although multiple parties may elect to provide notice, it is permissible for parties to determine in advance, either contractually or through trust framework policies, which party will provide the notice and obtain confirmation, as long as the determination is being based upon factors that center on enabling the subscriber to pay attention to the notice and make an informed choice.

The IdP is required to inform subscribers of all RPs that might access the subscriber's attributes. If an RP is on an IdP's allowlist as described in Sec. 4.6.1.1, the subscriber will not be prompted at runtime to consent to the release of their attributes. This single-sign-on scenario allows for a more seamless login experience for the subscriber, who might not even realize they are participating in a federation transaction. The IdP makes its list of allowlisted RPs available to the subscriber as part of the terms of the trust agreement. This information allows the subscriber to see which RPs might have access to their attributes, under what circumstances, and for what purposes.

If a subscriber's runtime decisions at the IdP were stored in the subscriber account by the IdP to facilitate future transactions, the IdP also needs to allow the subscriber to view and revoke any RPs that were previously approved during a runtime decision. This list includes information on which attributes were approved and when the approval was recorded. Similarly, if a subscriber's runtime decisions at the RP are stored in some fashion, the RP also needs to allow the subscriber to view and revoke any IdPs that were approved during a runtime decision.

7.3. Data Minimization

Federation enables the data exposed to an RP to be minimized, which can yield privacy protections for subscribers. Although an IdP may collect additional attributes beyond what the RP requires for its use case, only those attributes that were explicitly requested by the RP are to be transmitted by the IdP. In some instances, an RP does not require a full value of an attribute. For example, an RP may need to know whether the subscriber is over 13 years old, but has no need for the full date of birth. To minimize collection of potentially sensitive PII, the RP may request a derived attribute value (e.g., Question: Is the subscriber over 13 years old? Response: Y/N or Pass/Fail). This minimizes the RP's collection of potentially sensitive and unnecessary PII. Accordingly, Sec. 3.10.2 recommends the RP to, where feasible, request derived attribute values rather than full attribute values. To support this RP requirement IdPs are, in turn, required to support a derived attribute value.

7.4. Agency-Specific Privacy Compliance

Section 3.9 identifies agency requirements to consult their SAOP to determine privacy compliance requirements. It is critical to involve the agency's SAOP in the earliest stages of digital authentication system development to assess and mitigate privacy risks and advise the agency on compliance obligations such as whether the federation triggers the Privacy Act of 1974 or the E-Government Act of 2002 requirement to conduct a PIA. For example, if the agency is serving as an IdP in a federation, it is likely that the Privacy Act requirements will be triggered and require coverage by either a new or existing Privacy Act System of Records Notice since credentials would be maintained at the IdP on behalf of any RP it federates with. If, however, the agency is an RP and using a third-party IdP, digital authentication may not trigger the requirements of the Privacy Act, depending on what data passed from the RP is maintained by the agency at the RP (in such instances the agency may have a broader programmatic SORN that covers such data).

The SAOP can similarly assist the agency in determining whether a PIA is required. These considerations should not be read as a requirement to develop a Privacy Act SORN or PIA for use of a federated credential alone. In many cases it will make the most sense to draft a PIA and SORN that encompasses the entire digital authentication process or includes the digital authentication process as part of a larger programmatic PIA that discusses the program or benefit the agency is establishing online access.

Due to the many components of digital authentication, it is important for the SAOP to have an awareness and understanding of each individual component. For example, other privacy artifacts may be applicable to an agency offering or using federated IdP or RP services, such as Data Use Agreements, Computer Matching Agreements, etc. The SAOP can assist the agency in determining what additional requirements apply. Moreover, a thorough understanding of the individual components of digital authentication will enable the SAOP to thoroughly assess and mitigate privacy risks either through compliance processes or by other means.

7.5. Blinding in Proxied Federation

While some proxy structures — typically those that exist primarily to simplify integration — may not offer additional subscriber privacy protection, others offer varying levels of privacy to the subscriber through a range of blinding technologies. Privacy policies may dictate appropriate use of the subscriber attributes and authentication transaction data (e.g., identities of the ultimate IdP and RP) by the IdP, RP, and the federation proxy.

Technical means such as blinding can increase effectiveness of these policies by making the data more difficult to obtain. A proxy-based system has three parties, and the proxy can be used to hide information from one or more of the parties, including itself. In a double-blind proxy, the IdP and RP do not know each other’s identities, and their relationship is only with the proxy. In a triple-blind proxy, the proxy additionally does not have insight into the data being passed through it. As the level of blinding increases, the technical and operational implementation complexity may increase. Since proxies need to map transactions to the appropriate parties on either side as well as manage the keys for all parties in the transaction, fully triple-blind proxies are very difficult to implement in practice.

Even with the use of blinding technologies, a blinded party may still infer protected subscriber information through released attribute data or metadata, such as by analysis of timestamps, attribute bundle sizes, or attribute signer information. The IdP could consider additional privacy-enhancing approaches to reduce the risk of revealing identifying information of the entities participating in the federation.

The following table illustrates a spectrum of blinding implementations used in proxied federation. This table is intended to be illustrative, and is neither comprehensive nor technology-specific.

Table 4. Proxy Characteristics

Proxy Type	**RP knows IdP**	**IdP knows RP**	**Proxy can track subscriptions between RP and IdP**	**Proxy can see attributes of Subscriber**
Non-Blinding Proxy with Attributes	Yes	Yes	Yes	Yes
Non-Blinding Proxy	Yes	Yes	Yes	N/A
Double Blind Proxy with Attributes	No	No	Yes	Yes
Double Blind Proxy	No	No	Yes	N/A
Triple Blind Proxy with or without Attributes	No	No	No	No

8. Usability Considerations

This section is informative.

> In order to align with the standard terminology of user-centered design and usability, the term "user" is used throughout this section to refer to the human party. In most cases, the user in question will be the subject (in the role of applicant, claimant, or subscriber) as described elsewhere in these guidelines.

Ergonomic of Human-System Interaction — Part 11: Usability: Definitions and Concepts [ISO/IEC9241-11] defines usability as the "extent to which a system, product or service can be used by specified users to achieve specified goals with effectiveness, efficiency and satisfaction in a specified context of use." This definition focuses on users, goals, and context of use as key elements necessary for achieving effectiveness, efficiency and satisfaction. A holistic approach considering these key elements is necessary to achieve usability.

From the usability perspective, one of the major potential benefits of federated identity systems is to address the problem of user fatigue associated with managing multiple authenticators. While this has historically been a problem with usernames and passwords, the increasing need for users to manage many authenticators — whether physical or digital — presents a usability challenge.

As stated in Sec. 8 of [SP800-63A] and Sec. 8 of [SP800-63B], overall user experience is critical to the success of digital identity systems. This is especially true for federated identity systems, as federation is a less familiar user interaction paradigm for many users. Users' prior authentication experiences may influence their expectations.

The overall user experience with federated identity systems should be as smooth and easy as possible. This can be accomplished by following usability standards (such as the ISO 25060 series of standards) and established best practices for user interaction design.

Guidelines and considerations are described from the users' perspective.

Section 508 of the Rehabilitation Act of 1973 [Section508] was enacted to eliminate barriers in information technology and require federal agencies to make electronic and information technology accessible to people with disabilities. While these guidelines do not directly assert requirements from Section 508, identity service providers are expected to comply with Section 508 provisions. Beyond compliance with Section 508, Federal Agencies and their service providers are generally expected to design services and systems with the experiences of people with disabilities in mind to ensure that accessibility is prioritized throughout identity system lifecycles.

8.1. General Usability Considerations

Federated identity systems should:

- Minimize user burden (e.g., frustration, learning curve)
  - Minimize the number of user actions required.
  - Allow users to quickly and easily select among multiple subscriber accounts with a single IdP. For example, approaches such as Account Chooser allow users to select from a list of subscriber accounts they have accessed in the recent past, rather than start the federation process by selecting their IdP from a list of potential IdPs.
  - Balance minimizing user burden with the need to provide sufficient information to enable users to make informed decisions.
- Minimize the use of unfamiliar technical jargon and details (e.g., users do not need to know the terms IdP and RP if the basic concepts are clearly explained).
- Strive for a consistent and integrated user experience across the IdP and RP.
- Help users establish an understanding of identity by providing resources to users such as graphics, illustrations, FAQs, tutorials and examples. Resources should explain how users' information is treated and how transacting parties (e.g., RPs, IdPs, and brokers) relate to each other.
- Provide clear, honest, and meaningful communications to users (i.e., communications should be explicit and easy to understand).
- Provide users online services independent of location and device.
- Make trust relationships explicit to users to facilitate informed trust decisions. Trust relationships are often dynamic and context dependent. Users may be more likely to trust some IdPs and RPs with certain attributes or transactions more than others. For example, users may be more hesitant to use federated identity systems on websites that contain valuable personal information (such as financial or health). Depending on the perceived sensitivity of users' personal information, users may be less comfortable with commercial as IdPs since people often have concerns about advertising and data-usage of such companies. Conversely, some may have more confidence in the commercial IdPs than government IdPs based on their historical interactions with government services. Either way, it is critical to be clear to end-users on the entities involved in a federation transaction and, ideally, provide options that support the broadest set of stakeholder perceptions possible.
- Follow the usability considerations specified in [SP800-63A] Sec. 8 for any user-facing information.

- Clearly communicate how and where to acquire technical assistance. For example, provide users with information such as a link to an online self-service feature, chat sessions or a phone number for help desk support. Avoid redirecting users back and forth among transacting parties (e.g., RPs, IdPs, and brokers) to receive technical assistance.

- Perform integrative and continuous usability evaluations with representative users and realistic tasks in an appropriate context to ensure success of federated identity systems from the users' perspectives.

8.2. Specific Usability Considerations

This section addresses the specific usability considerations that have been identified with federated identity systems. This section does not attempt to present exhaustive coverage of all usability factors related to federated identity systems. Rather, it is focused on the larger, more pervasive themes in the usability literature, primarily users' perspectives on identity, user adoption, trust, and perceptions of federated identity space. In some cases, implementation examples are provided. However, specific solutions are not prescribed. The implementations mentioned are examples to encourage innovative technological approaches to address specific usability needs. See standards for system design and coding, specifications, APIs, and current best practices (such as OpenID and OAuth) for additional examples. Implementations are sensitive to many factors that prevent a one-size-fits-all solution.

8.2.1. User Perspectives on Online Identity

Even when users are familiar with federated identity systems, there are different approaches to federated identity (especially in terms of privacy and the sharing of information) that make it necessary to establish reliable expectations for how users' data are treated. Users and implementers have different concepts of identity. Users think of identity as logging in and gaining access to their own private space. Implementers think of identity in terms of authenticators and assertions, assurance levels, and the necessary set of identity attributes to provide a service. Given this disconnect between users' and implementers' concepts of identity, it is essential to help users form an accurate concept of identity as it applies to federated identity systems. A good model of identity provides users a foundation for understanding the benefits and risks of federated systems and encourage user adoption and trust of these systems.

To minimize the personal information collected and protect privacy, IdPs ought to provide users with pseudonymous options for providing data to RPs, where possible, and inform users of the benefits and drawbacks of pseudonymous identification. Likewise, RPs ought to request pseudonymous options for users when pseudonymity is possible for the RP's policy. Both IdPs and RPs need to seek to minimize unnecessary data transmission and inform users of which information is transmitted and for what purpose.

Many properties of identity have implications for how users manage identities, both within and among federations. Just as users manage multiple identities based on context outside of cyberspace, users must learn to manage their identity in a federated environment. Therefore, it must be clear to users how identity and context are used. The following factors should be considered:

- Provide users the requisite context and scope in order to distinguish among different user roles. For example, whether the user is acting on their own behalf or on behalf of another, such as their employer.
- Provide users unique, meaningful, and descriptive identifiers to distinguish among entities such as IdPs, RPs, and accounts. Any such user-facing identifiers are likely to be in addition to identifiers used by the underlying protocols, which are not normally exposed to the user.
- Provide users with information on data ownership and those authorized to make changes. Identities, and the data associated with them, can sometimes be updated and changed by multiple actors. For example, some healthcare data is updated and owned by the patient, while some data is only updated by a hospital or doctor’s practice.
- Provide users with the ability to easily verify, view, and update attributes. Identities and user roles are dynamic and not static; they change over time (e.g., age, health, and financial data). The ability to update attributes or make attribute release decisions may or may not be offered at the same time. Ensure the process for how users can change attributes is well known, documented, and easy to perform.
- Provide users means for updating data, even if the associated subscriber account or RP subscriber account no longer exists. Consider applicable audit, legal, or policy constraints for needs to track updated data.
- Provide users means to delete their identities completely, removing all information about themselves, including transaction history. Consider applicable audit, legal, or policy constraints that may preclude such action. In certain cases, full deactivation is more appropriate than deletion.
- Provide users with clear, easy-to-find, site/application data retention policy information.
- Provide users with appropriate anonymity and pseudonymity options, and the ability to switch among such identity options as desired, in accordance with an organization’s data access policies.
- Provide a means for users to manage each IdP to RP connection, including complete separation as well as the removal of RP access to one or more attributes.

8.2.2. User Perspectives of Trust and Benefits

Many factors can influence user adoption of federated identity systems. As with any technology, users may value some factors more than others. Users often weigh perceived benefits versus risks before making technology adoption decisions. It is critical that IdPs and RPs provide users with sufficient information to enable them to make informed decisions. The concepts of trust and tiers of trust — fundamental principles in federated identity systems — can drive user adoption. Finally, a positive user experience may also result in increased user demand for federation, triggering increased adoption by RPs.

This sub-section is focused primarily on user trust and user perceptions of benefits versus risks.

To encourage user adoption, IdPs and RPs need to establish and build trust with users and provide them with an understanding of the benefits and risks of adoption. The following factors should be considered:

- Allow users to control their information disclosure and provide explicit consent through the appropriate use of interactive user interfaces and notifications (see Sec. 7.2). Considerations such as balancing the content, size, and frequency of notifications as well as tailoring notifications to specific communities are necessary to avoid thoughtless user click-through.
- For attribute sharing, consider the following:
  - Provide a means for users to verify those attributes and attribute values that will be shared. Follow good security practices (see Sec. 3.10.2 and Sec. 6).
  - Enable users to consent to a partial list of attributes, rather than an all-or-nothing approach. Allow users some degree of online access, even if the user does not consent to share all information.
  - Allow users to update their consent to their list of shared attributes.
  - Minimize unnecessary information presented to users. For example, do not display system generated attributes (such as pairwise pseudonymous identifiers) even if they are shared with the RP as part of the authentication response.
  - Minimize user steps and navigation. For example, build attribute consent into the protocols so they’re not a feature external to the federation transaction. Examples can be found in standards such as OAuth or OpenID Connect.
  - Provide effective redress methods such that a user can recover from invalid attribute information claimed by the IdP or collected by the RP. See Sec 3.6 of [SP800-63] for more requirements on providing redress.

  - Minimize the number of times a user is required to consent to attribute sharing. Limiting the frequency of consent requests avoids user frustration from multiple requests to share the same attribute.

- Collect information for constrained usage only and minimize information disclosure (see Sec. 7.3). User trust is eroded by unnecessary and superfluous information collection and disclosure or user tracking without explicit user consent. For example, only request attributes from the user that are relevant to the current transaction, not for all possible transactions a user may or may not access at the RP.

- Clearly and honestly communicate potential benefits and risks of using federated identity to users. Benefits that users value include time savings, ease of use, reduced number of passwords to manage, and increased convenience.

User concern over risk can negatively influence willingness to adopt federated identity systems. Users may have trust concerns, privacy concerns, security concerns, and single-point-of-failure concerns. For example, users may be fearful of losing access to multiple RPs if a single IdP is unavailable, either temporarily or permanently. Additionally, users may be concerned or confused about learning a new authentication process. In order to foster the adoption of federated identity systems, the perceived benefits must outweigh the perceived risks.

8.2.3. User Mental Models and Beliefs

Users’ beliefs and perceptions predispose them to expect certain results and to behave in certain ways. Such beliefs, perceptions, and predispositions are referred to in the social sciences as mental models. For example, people have a mental model of dining out that guides their behavior and expectations at each establishment, such as fast food restaurants, cafeterias, and more formal restaurants. Thus, it is not necessary to be familiar with every establishment to understand how to interact appropriately at each one.

Assisting users in establishing good and complete mental models of federation allows users to generalize beyond a single specific implementation. If federated identity systems are not designed from users’ perspectives, users may form incorrect or incomplete mental models that impact their willingness to adopt these systems. The following factors should be considered:

- Clearly explain the working relationship and information flow among the transacting parties (e.g., RPs, IdPs, and proxies) to avoid user misconceptions. Use the actual names of the entities in the explanation rather than using the generic terms IdPs and RPs.

  - Provide prominent visual cues and information so that users understand why seemingly unrelated entities have a working relationship. For example, users may be concerned with mixing online personal activities with government services due to a lack of understanding of the information flow in federated identity systems.

  - Provide prominent visual cues and information to users about redirection when an RP needs to redirect control from their site to an IdP. For example, display RP branding within the IdP user interface to inform users when they are logging in with their IdP for access to the destination RP.

- Provide users with clear and usable ways (e.g., visual assurance) to determine the authenticity of the transacting parties (e.g., RPs, IdPs, and proxies). This will also help to alleviate user concern over leaving one domain for another, especially if the root domain changes (e.g., .gov to .com). For example, display the URL of the IdP so that the user can verify that they are not being phished by a malicious site.

- Provide users with clear information, including visual cues, regarding logins and logouts. Depending on the implementation, logging into an RP with a federated account can create long-running sessions for the user at both the IdP and RP. Users may not realize that ending their session with the RP will not necessarily end their session with the IdP; users will need to explicitly “log out” of the IdP. Users require clear information to remind them if explicit logouts are required to end their IdP sessions. Both the IdP and RP could also have automated logout features, based on time since authentication or an activity timeout. Users require clear information about when their session might end without any action on their part, in order to avoid frustration, lost work, or insecure workarounds like copying data out of a secure site in order to avoid an unexpected session timeout.

9. Equity Considerations

This section is informative.

Equitable access to the functions of IdPs and RPs is an essential element of a federated identity system. The ability for all subscribers to authenticate reliably is required to provide equitable access to government services, even when using federation technology, as specified in Executive Order 13985, *Advancing Racial Equity and Support for Underserved Communities Through the Federal Government* [EO13985]. In assessing equity risks, IdPs and RPs should consider the overall user population served by their federated identity service. Additionally, IdPs and RPs further identify groups of users within the population whose shared characteristics can cause them to be subject to inequitable access, treatment, or outcomes when using that service. The Usability Considerations provided in Sec. 8 should also be considered to help ensure the overall usability and equity for all persons using federated identity services.

In its role as the verifier, the IdP needs to be aware of equity considerations related to identity proofing, attribute validation, and enrollment as enumerated in [SP800-63A] Sec. 9 and equity considerations concerning authenticators as enumerated in [SP800-63B] Sec. 9. An RP offering FAL3 will also need to be aware of these same authenticator considerations when processing bound authenticators and holder-of-key assertions.

Since the federation process takes place over a network protocol between multiple active parties, the experience of authenticating using the federation system may present equity problems, such as the following examples:

- Completing the entire federation transaction without timing out may be difficult for subscribers without a reliable network connection, such as those in rural areas.
- It may be difficult to provide informed consent for a runtime decision regarding the release of attributes for subscribers with intellectual, developmental, learning, or neurocognitive difficulties.
- Systems with sufficient processing power, network access, and other features required to interact with both the IdP and the RP simultaneously may be too costly or beyond some subscribers’ technological skill to access or use.
- Subscribers that share devices may find allowlist-based systems difficult to manage securely, as other users of the device could silently gain unintended access to an RP through a session still active at the IdP.
- It could be prohibitively difficult to re-establish an account at the RP for subscribers who lose access to their IdP for any of a variety of reasons.

Additionally, subscribers in disadvantaged populations could be more susceptible to monitoring and tracking through federation systems, as discussed in Sec. 7. If the IdP knows the subscriber is part of a disadvantaged population, the IdP could specifically target the subscriber by profiling them and their access to the set of RPs, and use the data gathered against the subscriber. Alternatively, the IdP could learn that that the subscriber is part of a disadvantaged population by watching the RP connections. For example, if the IdP sees that the subscriber logs into social services, the IdP has learned things about the subscriber’s socioeconomic status that were not disclosed to the IdP. The IdP could then use this to unfairly target the subscriber and provide a lower quality of service. Additionally, subscribers in disadvantaged populations are at a greater risk of having their data correlated between a set of colluding RPs. For example, a set of RPs could share subscriber attributes and behavior among them in order to justify denial of the RP’s services to the subscriber. As such, IdPs and RPs are encouraged to use privacy-enhancing techniques equally across subscriber populations.

When consent dialogs and notifications are sent to users, the content of these should be tailored to different subscriber populations in order to facilitate subscriber understanding and avoid thoughtless click-through.

IdPs are required to disclose the method of proofing used for each subscriber as recorded in the subscriber account. This includes all available forms of proofing and exception processes, and possibly compensating controls, as defined in the trust agreement. IdPs and CSPs should not single out subscribers who have had to make use of exception handling or compensating controls beyond the proofing information contained in their subscriber account to avoid bias processing against certain subscriber populations.

Since federation transactions are intended to cross security domain boundaries, discrepancies between the interests of the IdP and the RP could pose additional considerations. This difference in requirements has to be addressed in the trust agreement that governs the connection between these parties, and practices such as transparent reporting can help address some forms of disparities. Furthermore, the availability of alternative IdPs (for the RP) and RPs (for the IdP) for a given service can help enhance the equity of the system overall. For example in a public-private partnership, if a private IdP is used to access a federal RP, or a federal IdP is used to access a private RP, the public and private systems could be driven by different motivations and bound by different requirements in terms of equity, accessibility, and transparency.

Normative requirements have been established requiring IdPs and RPs to mitigate the problems in this area that are expected to be most common. However, normative requirements are unlikely to have anticipated all potential equity problems. Potential equity problems also will vary for different applications. Accordingly, IdPs and RPs need to provide mechanisms for subscribers to report inequitable authentication requirements and to advise them on potential alternative authentication strategies.

This guideline allows the binding of additional federated identifiers to an RP subscriber account to minimize the risk of IdP access loss (see Sec. 3.7). However, a subscriber might find it difficult to have multiple IdP accounts that are acceptable to the RP at the same time. This inequity can be addressed by having the RP having its own account recovery process that allows for the secure linking of multiple federated identifiers to the RP subscriber account.

RPs need to be aware that not all subscribers will necessarily have access to the same IdPs. The RPs can institute locally authenticated accounts for such subscribers, and later allow binding of those accounts to federated identifiers.

10. Examples

This section is informative.

This appendix contains several example scenarios of federation used in conjunction with the requirements in these guidelines.

The scenarios in this section are for illustrative purposes and do not convey additional requirements beyond those imposed by these guidelines.

10.1. Mapping FALs to Common Federation Protocols

Of protocols commonly in use today, OpenID Connect [OIDC] and SAML [SAML] both provide a variety of capabilities that can be leveraged to reach the requirements at different FALs. Table 5 provides examples of specific options in these protocols that could be deployed to reach a given FAL. It’s important to note that these guidelines do not represent a normative mapping to the given FALs and the entirety of the federation process has to be considered when establishing an FAL. Additionally, each FAL could be reached by processes, deployments, and procedures that are not listed in this table.

Table 5. FAL Protocol Examples

	OIDC	**SAML**
FAL1	All core flows in [OIDC] (Authorization Code, Implicit, and Hybrid) can all be configured to require signing of the assertion (the ID Token) using JSON Web Signatures. Assertions are presented in a variety of front and back channel methods. Each of these flows can be built using both static and dynamic client registration. Profiles such as [OIDC-Basic] and [OIDC-Implicit] can provide additional guidance for interoperable deployments.	The [SAML-WebSSO] profile allows for the signing of assertions using XML D-Sig and presentation of the assertion using the front channel. SAML deployments are generally set up with a static registration, sometimes managed through a federation authority, which can meet the requirements at this FAL and above.
FAL2	Flows that present the ID Token in the back channel (such as Authorization Code and Hybrid) can provide a level of injection protection.	The Artifact Binding of SAML defined in [SAML-Bindings] allows for a back-channel presentation of SAML assertions that can provide a level of injection protection.
FAL3	The ID Token can include the claims necessary for Holder-of-Key and Bound Authenticator assertion presentations, though to date there are not industry standard profiles for doing so.	The SAML Holder-of-Key profile can fit the assertion requirements at this level, if combined with other deployment choices.

For OpenID Connect in particular, it is common practice to give access to both an identity API (the UserInfo Endpoint) as well as additional APIs. While the security of API access is outside the scope of these guidelines (which are concerned with the identity assertion primarily), it is sensible for an OpenID Connect implementation to want to increase the security of all API calls in tandem with the FAL. For example, in addition to requiring a Holder-of-Key assertion at FAL3, which requires verification of a subscriber-held key, an OpenID Connect system might also require sender-constrained access tokens for API access, which require the verification of a key held by the RP for each API call.

10.2. Direct Connection to an Agency’s IdP

Agency A, which issues and manages subscriber accounts, sets up and operates an OpenID Connect IdP in order to make these subscriber accounts available online through a federation process.

The RP enters into a pairwise trust agreement with the IdP to accept assertions for subscribers from Agency A. The RP declares the set of attributes that it needs from the IdP as part of this agreement. The trust agreement stipulates that the subscriber is the authorized party for determining the release of attributes in the federation transaction.

The IdP generates a federated identifier for the subscriber account by taking the unique internal identifier for the subscriber account (such as an employee record number) and passing it through a one-way cryptographic function to create a unique identifier for the subscriber account. Such an identifier does not allow an RP to calculate the internal identifier but will be stable across attribute changes.

Per the terms of the trust agreement, the subscriber is prompted by the IdP the first time they log on to the RP. The IdP asks for the subscriber's consent at runtime to share their attributes with the RP, displaying to the subscriber the RP's requested uses for these attributes on the consent screen. The IdP also prompts the subscriber to allow the IdP to remember this consent decision. This stored decision causes the IdP to act on the stored consent in a future request and not prompt the subscriber if the same RP requests the same attributes.

The assertion, formatted as an OpenID Connect ID Token, contains the minimum set of attributes to facilitate the federated log in. Apart from the federated identifier, the assertion contains no identifying information about the subscriber. In addition to the assertion, the RP is given an OAuth 2.0 access token that allows the RP to access the identity API hosted by the IdP, the OpenID Connect UserInfo Endpoint. The RP can choose to call this API to get additional attributes as needed, such as the first time the subscriber uses the RP. Since this RP follows a just-in-time provisioning model, when the RP sees the subscriber's federated identifier for the first time, the RP creates an RP subscriber account for that federated identifier and calls the identity API to populate the RP subscriber account with the subscriber's attributes. For future authentications with this subscriber, the RP can decide if its cache of attributes is reasonably recent enough or if it should be refreshed by calling the identity API.

10.3. Multilateral Federation Network

Agencies A, B, and C each have an IdP running OpenID Connect for their subscriber accounts. All three agencies join a multilateral federation run by an independent agency set up to provide inter-agency connections. The federation authority independently verifies that each IdP represents the agency in question. The federation authority publishes the discovery records of the IdPs for all agencies that are part of the multilateral federation. This publication allows RPs within the federation to discover which IdP is to be used to access accounts for a given agency under the rules of the federation agreement.

RPs X and Y wish to allow logins from agencies A, B, and C, and the RPs declare their intent and a list of required attributes to the federation authority. The federation authority assesses both RP requests and adds them to the multilateral federation's trust agreement. This allows both RPs to register at each of the three separate IdPs as needed for each agency.

Both RPs interface directly with each of the three IdPs and not through a federation proxy. When a new IdP or RP is added to the multilateral federation agreement, the existing IdPs and RPs are notified of the new component and its parameters.

The IdPs and RPs establish a shared signaling channel under the auspices of the federation authority. This allows any IdP and any RP to report suspicious or malicious behavior that involves a specific account to the rest of the members under the federation authority.

10.4. Issuance of a Credential to a Digital Wallet

Agency B makes its subscriber accounts available for federation through the use of digital wallet technology. The agency's agreement for issuing credentials into wallets is facilitated by a federation authority that is set up to manage digital wallets across the federal government. The federation authority establishes the identity of the CSP for each agency under the multilateral agreement, and it ensures that only the CSP for Agency B can onboard subscriber-controlled wallets for Agency B within the multilateral trust agreement.

A subscriber has a digital wallet running on their device that they want to use with their subscriber account from Agency B. Within these guidelines, the digital subscriber-controlled wallet needs to be onboarded by the CSP before it can act as an IdP. To begin this process, the subscriber directs their digital wallet software to Agency B's CSP. The subscriber uses a biometric factor to activate their digital wallet, and the digital wallet makes an onboarding request to the CSP for the subscriber account. This onboarding request includes proof of a key held by the digital wallet. The CSP verifies the wallet's proof and processes any additional attestations from the wallet device.

The subscriber authenticates to the CSP during the onboarding process. The CSP prompts the subscriber with the terms of the trust agreement from the federation authority, and asks the subscriber to confirm that they wish to issue an identity to the digital wallet in question. The subscriber is informed of the sets of attributes that are made available to the wallet.

The CSP creates an attribute bundle that includes the subscriber's attributes as well as a reference to the digital wallet's key. The CSP signs this attribute bundle with its own key and returns the bundle to the digital wallet.

When the subscriber needs to authenticate to an RP, the RP sends a query to the subscriber's wallet for a credential that fits the RP's needs. The RP has a trust agreement with the same federation authority, agreeing to trust identities issued under the multilateral trust agreement’s rules. The digital wallet, acting as an IdP, identifies that the RP’s request can be fulfilled by the attribute bundle issued from Agency B’s CSP. The digital wallet prompts the subscriber to activate the IdP function of the digital wallet software using a local biometric factor. The digital wallet prompts the subscriber to confirm that they want to present the requested attributes to the RP in question. When the subscriber accepts, the IdP function of the digital wallet creates an assertion for the RP that is signed with the digital wallet’s keys. The assertion includes the attribute bundle from the CSP, which itself is covered by the signature from the IdP function. The IdP delivers the assertion to the RP.

The RP receives the signed assertion and validates the signature of the attribute bundle from the CSP, using the CSP’s keys identified by the federation authority. The RP then validates the signature of the assertion using the key identified in the assertion. When these checks pass successfully, the RP creates an RP subscriber account to represent the subscriber at the RP, based on the information in the assertion.

10.5. Enterprise Application Single-Sign-On

For enterprise applications, it is a common pattern for the organization to make the application available to all potential subscribers within the agency, through the use of an allowlist and pre-provisioned accounts.

In this scenario, Agency E establishes a pairwise agreement with an RP to provide an enterprise-class service to all employees of Agency E through the agency’s OpenID Connect IdP. As part of this trust agreement, the IdP allows access to a SCIM-based provisioning API for the RP. The IdP creates a federated identifier for each subscriber account and uses the provisioning API to push the federated identifiers and their associated attributes to the RP. In this way, the RP can pre-provision an RP subscriber account for every subscriber in the IdP’s system, allowing the RP to offer functions like access rights, data sharing, and messaging to all accounts on the system, whether or not a specific account has logged in to the RP yet.

Under the terms of the trust agreement, the RP is placed on an allowlist with the IdP. The allowlist entry states that:

- The subscriber has an active subscriber account at Agency E
- The subscriber has authenticated with the IdP at AAL2 or greater
- The RP is allowed to request only the federated identifier and basic authentication event information, since all other necessary attributes will be available through the provisioning API
- The federation transaction is at FAL2

Consequently, subscribers are not prompted for consent at runtime because the agency consented to use the service on behalf of all accounts at the time the RP was onboarded. This gives subscribers a seamless single sign-on experience, even though a federation protocol is being used across security domain boundaries. Since the IdP does not use any runtime decisions, any deviation from the allowlist parameters causes the federation transaction to fail.

The RP subscriber accounts are synchronized using the provisioning API. When a new subscriber account is created, modified, or deleted at the IdP, the IdP updates the status of the RP subscriber account using the provisioning API. This allows the RP to always have an up-to-date status for each subscriber account. For example, when the subscriber account is terminated at the IdP, the provisioning API signals to the RP that the corresponding RP subscriber account is to be terminated immediately. The RP removes all locally cached attributes for the account in question, except for the identifiers and references in audit and access logs.

10.6. FAL3 With a Smart Card

A subscriber has a cryptographic authenticator on a smart card. The certificate on this smart card can be verified independently by both the IdP and RP thanks to the use of a shared PKI system stipulated by the trust agreement. This type of authenticator can be used in a holder-of-key assertion at FAL3.

The subscriber starts the federation process and authenticates to the IdP using their authenticator. The IdP creates an assertion that includes a flag indicating that the assertion is intended for use at FAL3. The assertion also contains the certificate common name (CN) and thumbprint of the certificate to be used as a bound authenticator.

When the RP receives the assertion, the RP processes the assertion as usual and sees the FAL3 flag and the certificate attributes. The subscriber authenticates to the RP using their authenticator, and the RP verifies that the certificate presented by the subscriber matches the certificate in the assertion from the IdP. When these match, the RP creates a secure session with the subscriber at FAL3.

10.7. FAL3 With a non-PKI Authenticator

A subscriber has a hardware cryptographic authenticator that speaks the WebAuthn protocol. This authenticator is not tied to any PKI system, and in fact the authenticator device presents completely different and unlinked keys to both the IdP and RP during its normal authentication process. This kind of authenticator can still be used at FAL3 if the RP manages the bound authenticator.

In this example, when the subscriber uses this authentication device at the IdP, it presents proof of Key1. When the subscriber uses the same device at the RP, it presents proof of Key2. These are logically two separate authenticators, but from the perspective of the subscriber, they are using the same device in multiple places.

To start a federation transaction, the subscriber authenticates to the IdP using Key1. The IdP then creates an assertion that is flagged as FAL3. Since the IdP has no visibility into the existence and use of Key2, the assertion says that the subscriber is using a bound authenticator to reach FAL3. When the RP processes this assertion, the RP checks the RP subscriber account associated with the federated identifier in the assertion to find an RP bound authenticator for that account using Key2. The RP prompts the subscriber to authenticate using Key2. When that key is verified, the RP creates a secure session with the subscriber at FAL3.

10.8. FAL3 With Referred Token Binding

A subscriber authenticates to their IdP using a certificate that is trusted by the IdP but not known to the RP, since the IdP and RP are not in a shared PKI environment. However, the IdP and RP support the referred token binding extension of TLS. When the subscriber presents their certificate to the IdP, the IdP creates an assertion with the CN and thumbprint of the subscriber’s certificate. Along with the assertion or assertion reference, the IdP returns token binding headers. When these headers are presented to the RP, the RP can use them to associate the contents of the assertion with the subscriber’s bound authenticator. The RP still has to verify the certificate, but the token binding allows the RP to do so without having to separately trust the certificate chain of the authenticator’s certificate.

10.9. Ephemeral Federated Attribute Exchange

An RP needs to access a specific attribute for a subscriber, such as proof of age or affiliation with a known entity like a specific agency, without needing to know the identity of the subscriber. The RP requests only the derived attribute values that it needs in order to process its transaction, in this case a simple boolean of whether the subscriber is of age or is affiliated with the entity. The federation process creates an authenticated session between the RP and the subscriber. However, the RP uses an ephemeral provisioning mechanism, retaining only a record of the transaction and no further identifying attributes of the subscriber. The IdP provides a pairwise pseudonymous identifier to the RP. Since the IdP knows of the ephemeral nature of the RP subscriber account, the IdP can provide a distinct PPI to the RP on each request without affecting the subscriber’s usage of the RP. The IdP prompts the subscriber at runtime to release the derived attributes, preventing the RP from silently polling subscriber accounts against changes in information over time.

10.10. Multiple Different Authorized Parties and Trust Agreements

As a subscriber uses services at multiple RPs, different trust agreements can come into play, and those agreements can have different requirements and experiences. In this scenario, the subscriber has an account through a single IdP which they use at three different RPs, each with a different kind of trust agreement and different requirements for consent and notification.

Organizational Authorized Party:

An apriori trust agreement is established for an agency connecting to an enterprise service (the RP) to be made available to all subscribers at the agency. The authorized party for this trust agreement is the agency, and the IdP is configured with an allowlist entry for the RP with the set of common attributes requested by the RP for its use. When a subscriber logs in to the enterprise service, they are not prompted with any runtime decisions regarding the service, since the trust agreement establishes this connection as trusted. The details of this trust agreement are available to the subscriber from the IdP, including the list of attributes that are released to the RP and for what purpose.

Individual Authorized Party:

A separate a priori trust agreement is established by the agency for another service (a different RP), and this service is made available to all subscribers at the same agency. This trust agreement stipulates that the subscriber is the authorized party for release of attribute information to the RP. When logging in to the service, each subscriber is prompted for their consent to release their attributes to the RP. The prompt includes the context for the subscriber to make an appropriate security decision, including a link to the details of the trust agreement and a list of attributes being released and their purpose of use. The IdP allows the subscriber to save this consent decision so that when this subscriber logs in to this same RP in the future, the subscriber is not prompted again for their consent so long as the trust agreement and the request from the RP have not changed.

Subscriber-driven Service Access:

A subscriber-driven trust agreement is established when the subscriber goes to access an RP that is otherwise unknown by their IdP. The RP informs the subscriber about the uses of all attributes being requested from the IdP, and the IdP prompts the subscriber for consent to release their attributes to the RP. The IdP also warns the subscriber that the RP is unknown to the agency, and provides the subscriber with information received by the RP to help the subscriber make a secure decision.

All of these scenarios are involve the same subscriber account.

10.11. Shared Pairwise Pseudonymous Identifiers for Multiple RPs

A group of three applications is deployed in support of a specific mission, giving collaboration, document storage, and calendar capabilities. Due to the nature of the separate applications, they are deployed as separate RPs, but all are bound to the same IdP using a common trust agreement. The trust agreement stipulates that the three RPs are to be issued a shared PPI, so that the applications can coordinate individual subscriber accounts with each other but not with any other applications in the deployed environment. The IdP uses an algorithm to generate a shared PPI that incorporates a randomized identifier for the set of applications as well as a unique identifier for each subscriber accounts. As a result, all three RPs get the same PPI for each subscriber, but no other RP is issued that same identifier.

10.12. RP Authentication to an IdP

A federation transaction typically takes place over multiple network calls. Throughout this process, it is important for the IdP and RP to know that they are talking to the same party that they were in a previous step, and ultimately to the party that they expect to be in the transaction with in the first place.

Different techniques exist that provide different degrees of assurance, depending on the federation protocol in use and the needs of the system. For example, the Authorization Code Flow of [OIDC] allows the RP to register a shared secret or private key with the IdP prior to the transaction, allowing the IdP to strongly authenticate the RP’s request in the back channel to retrieve the assertion. In addition, the Proof Key for Code Exchange protocol in [RFC7636] allows the RP to dynamically create an unguessable secret that is transmitted in hashed form in the front channel and then transmitted in full in the back channel along with the assertion reference. These techniques can of course be combined for even greater assurance.

Federation authorities can also facilitate the authentication process. If the RP registers its public key and identifier with the federation authority, the IdP needs only to retrieve the appropriate keys from the federation authority instead of requiring the RP to register itself ahead of time.

Technical profiles of specific federation protocols are out of scope of these guidelines, but high security profiles such as [FAPI] provide extensive guidelines for implementers to deploy secure federation protocols.

References

This section is informative.

[A-130] Office of Management and Budget (2016) Managing Information as a Strategic Resource. (The White House, Washington, DC), OMB Circular A-130, July 28, 2016. Available at https://obamawhitehouse.archives.gov/sites/default/files/omb/assets/OMB/circulars/a130/a130revised.pdf

[EO13985] Biden J (2021) Advancing Racial Equity and Support for Underserved Communities Through the Federal Government. (The White House, Washington, DC), Executive Order 13985, January 25, 2021. https://www.federalregister.gov/documents/2021/01/25/2021-01753/advancing-racial-equity-and-support-for-underserved-communities-through-the-federal-government

[FAPI] Fett D, Bradley J, Heenan J (2024), *FAPI 2.0 Security Profile (draft)*. (OpenID Foundation, San Ramon, CA). https://openid.bitbucket.io/fapi/fapi-2_0-security-profile.html

[FEDRAMP] General Services Administration (2022), *How to Become FedRAMP Authorized*. Available at https://www.fedramp.gov/

[FIPS140] National Institute of Standards and Technology (2019) Security Requirements for Cryptographic Modules. (U.S. Department of Commerce, Washington, DC), Federal Information Processing Standards Publication (FIPS) 140-3. https://doi.org/10.6028/NIST.FIPS.140-3

[ISO/IEC9241-11] International Standards Organization (2018) *ISO/IEC 9241-11 Ergonomics of human-system interaction – Part 11: Usability: Definitions and concepts* (ISO, Geneva, Switzerland). Available at https://www.iso.org/standard/63500.html

[ISO/IEC18013-5] International Standards Organization (2021) *ISO/IEC 18013-5 Personal identification — ISO-compliant driving licence — Part 5: Mobile driving licence (mDL) application* (ISO, Geneva, Switzerland). Available at https://www.iso.org/obp/ui/en/#iso:std:iso-iec:18013:-5:ed-1:v1:en

[NISTIR8062] Brooks S, Garcia M, Lefkovitz N, Lightman S, Nadeau E (2017) An Introduction to Privacy Engineering and Risk Management in Federal Systems. (National Institute of Standards and Technology, Gaithersburg, MD), NIST Interagency or Internal Report (IR) 8062, January 2017. https://doi.org/10.6028/NIST.IR.8062

[NISTIR8112] Grassi PA, Lefkovitz NB, Nadeau EM, Galluzzo RJ, Dinh AT (2018) Attribute Metadata: A proposed Schema for Evaluating Federated Attributes. (National Institute of Standards and Technology, Gaithersburg, MD), NIST Interagency or Internal Report (IR) 8112. https://pages.nist.gov/NISTIR-8112/NISTIR-8112.html

[OIDC] Sakimura N, Bradley J, Jones M, de Medeiros B, Mortimore C (2014) OpenID Connect Core 1.0 incorporating errata set 1 (OpenID Foundation, San Ramon, CA). https://openid.net/specs/openid-connect-core-1_0.html

[OIDC-Basic] Sakimura N, Bradley J, Jones M, de Medeiros B, Mortimore C (2022) OpenID Connect Basic Client Implementer's Guide 1.0 (OpenID Foundation, San Ramon, CA). https://openid.net/specs/openid-connect-basic-1_0.html

[OIDC-Implicit] Sakimura N, Bradley J, Jones M, de Medeiros B, Mortimore C (2022) OpenID Connect Implicit Client Implementer's Guide 1.0 (OpenID Foundation, San Ramon, CA). https://openid.net/specs/openid-connect-implicit-1_0.html

[OIDC-Registration] Sakimura N, Bradley J, Jones M (2023) OpenID Connect Dynamic Client Registration 1.0 incorporating errata set 2 (OpenID Foundation, San Ramon, CA). https://openid.net/specs/openid-connect-registration-1_0.html

[RFC5246] Rescorla E, Dierks T (2008) The Transport Layer Security (TLS) Protocol Version 1.2. (Internet Engineering Task Force (IETF)), IETF Request for Comments (RFC) 5246. https://doi.org/10.17487/RFC5246

[RFC5280] Cooper D, Santesson S, Farrell S, Boeyen S, Housley R, Polk W (2008) Internet X.509 Public Key Infrastructure Certification and Certificate Revocation List (CRL) Profile. (Internet Engineering Task Force (IETF)), IETF Request for Comments (RFC) 5280. https://doi.org/10.17487/RFC5280

[RFC7591] Richer J, Jones M, Bradley J, Machulak M, Hunt P (2015) OAuth 2.0 Dynamic Client Registration Protocol. (Internet Engineering Task Force, Reston, VA), RFC 7591. https://doi.org/10.17487/RFC7591

[RFC7636] Sakimura N, Bradley J, Agarwal N (2015) Proof Key For Code Exchange by OAuth Public Clients. (Internet Engineering Task Force, Reston, VA), RFC 7636. https://doi.org/10.17487/RFC7636

[RFC9325] Sheffer Y, Saint-Andre P, Fossati T (2022) Recommendations for Secure Use of Transport Layer Security (TLS) and Datagram Transport Layer Security (DTLS). (Internet Engineering Task Force (IETF)), IETF Request for Comments (RFC) 9325. https://doi.org/10.17487/RFC9325

[SAML] Ragouzis N, Hughes J, Philpott R, Maler E, Madsen P, Scavo T (2008) Security Assertion Markup Language (SAML) V2.0 Technical Overview. (Organization for Advancement of Structured Information Standards (OASIS) Open, Woburn, MA), SAML 2.0. https://docs.oasis-open.org/security/saml/Post2.0/sstc-saml-tech-overview-2.0.html

[SAML-Bindings] Cantor S, Frederick H, Kemp J, Philpott R, Maler M (2005) Bindings for the OASIS Security Assertion Markup Language (SAML) V2.0. (Organization for

Advancement of Structured Information Standards (OASIS) Open, Woburn, MA), SAML 2.0. https://docs.oasis-open.org/security/saml/v2.0/saml-bindings-2.0-os.pdf

[SAML-WebSSO] Hughes J, Cantor S, Hodges J, Hirsch F, Mishra P, Philpott R, Maler E (2005) Profiles for the OASIS Security Assertion Markup Language (SAML) V2.0. (Organization for Advancement of Structured Information Standards (OASIS) Open, Woburn, MA), SAML Profiles 2.0. https://docs.oasis-open.org/security/saml/v2.0/saml-profiles-2.0-os.pdf

[Section508] General Services Administration (2022) *IT Accessibility Laws and Policies*. Available at https://www.section508.gov/manage/laws-and-policies/

[SD-JWT] Fett D, Yasuda K, Campbell B (2024) Selective Disclosure for JWTs (SD-JWT). (Internet Engineering Task Force, Reston, VA). Active Internet Draft. https://datatracker.ietf.org/doc/draft-ietf-oauth-selective-disclosure-jwt/

[SP800-52] McKay K, Cooper D (2019) Guidelines for the Selection, Configuration, and Use of Transport Layer Security (TLS) Implementations. (National Institute of Standards and Technology), NIST Special Publication (SP) 800-52 Rev. 2. https://doi.org/10.6028/NIST.SP.800-52r2

[SP800-53] Joint Task Force (2020) Security and Privacy Controls for Information Systems and Organizations. (National Institute of Standards and Technology, Gaithersburg, MD), NIST Special Publication (SP) 800-53 Rev. 5, Includes updates as of December 10, 2020. https://doi.org/10.6028/NIST.SP.800-53r5

[SP800-63] Temoshok D, Proud-Madruga D, Choong YY, Galluzzo R, Gupta S, LaSalle C, Lefkovitz N, Regenscheid A (2024) Digital Identity Guidelines. (National Institute of Standards and Technology, Gaithersburg, MD), NIST Special Publication (SP) 800-63-4 2pd. https://doi.org/10.6028/NIST.SP.800-63-4.2pd

[SP800-63A] Temoshok D, Abruzzi C, Choong YY, Fenton JL, Galluzzo R, LaSalle C, Lefkovitz N, Regenscheid A (2024) Digital Identity Guidelines: Identity Proofing and Enrollment. (National Institute of Standards and Technology, Gaithersburg, MD), NIST Special Publication (SP) 800-63A-4 2pd. https://doi.org/10.6028/NIST.SP.800-63a-4.2pd

[SP800-63B] Temoshok D, Fenton JL, Choong YY, Lefkovitz N, Regenscheid A, Galluzzo R, Richer JP (2024) Digital Identity Guidelines: Authentication and Authenticator Management. (National Institute of Standards and Technology, Gaithersburg, MD), NIST Special Publication (SP) 800-63B-4 ipd. https://doi.org/10.6028/NIST.SP.800-63b-4.2pd

[SP800-131A] Barker E, Roginsky A (2019) Transitioning the Use of Cryptographic Algorithms and Key Lengths. (National Institute of Standards and Technology, Gaithersburg, MD), NIST Special Publication (SP) 800-131Ar2. https://doi.org/10.6028/NIST.SP.800-131Ar2

Appendix A. List of Symbols, Abbreviations, and Acronyms

1:1 Comparison
One-to-One Comparison

ABAC
Attribute-Based Access Control

AAL
Authentication Assurance Level

CAPTCHA
Completely Automated Public Turing test to tell Computer and Humans Apart

CSP
Credential Service Provider

CSRF
Cross-Site Request Forgery

DNS
Domain Name System

FAL
Federation Assurance Level

FEDRAMP
Federal Risk and Authorization Management Program

IAL
Identity Assurance Level

IdP
Identity Provider

JOSE
JSON Object Signing and Encryption

JWT
JSON Web Token

MAC
Message Authentication Code

PIA
Privacy Impact Assessment

PII
Personally Identifiable Information

PIN
Personal Identification Number

PKI
Public Key Infrastructure

PPI
Pairwise Pseudonymous Identifier

RMF
Risk Management Framework

RP
Relying Party

SAML
Security Assertion Markup Language

SAOP
Senior Agency Official for Privacy

SCIM
System for Cross-domain Identity Management

SORN
System of Records Notice

TLS
Transport Layer Security

XSS
Cross-Site Scripting

Appendix B. Glossary

A wide variety of terms are used in the realm of digital identity. While many definitions are consistent with earlier versions of SP 800-63, some have changed in this revision. Many of these terms lack a single, consistent definition, warranting careful attention to how the terms are defined here.

account linking
The association of multiple *federated identifiers* with a single *RP subscriber account*, or the management of those associations.

account resolution
The association of an *RP subscriber account* with information already held by the *RP* prior to the *federation transaction* and outside of a *trust agreement*.

activation factor
An additional *authentication factor* that is used to enable successful *authentication* with a *multi-factor authenticator*.

allowlist
A documented list of specific elements that are allowed, per policy decision. In *federation* contexts, this is most commonly used to refer to the list of *RPs* allowed to connect to an *IdP* without subscriber intervention. This concept has historically been known as a *whitelist*.

approved cryptography
An encryption algorithm, *hash function*, random bit generator, or similar technique that is *Federal Information Processing Standard* (FIPS)-approved or NIST-recommended. Approved algorithms and techniques are either specified or adopted in a FIPS or NIST recommendation.

assertion
A statement from an *IdP* to an *RP* that contains information about an authentication event for a subscriber. Assertions can also contain identity *attributes* for the subscriber.

assertion reference
A data object, created in conjunction with an *assertion*, that is used by the *RP* to retrieve an assertion over an *authenticated* protected channel.

assertion presentation
The method by which an *assertion* is transmitted to the *RP*.

asymmetric keys
Two related keys, comprised of a *public key* and a *private key*, that are used to perform complementary operations such as encryption and decryption or signature *verification* and generation.

attribute
A quality or characteristic ascribed to someone or something. An identity attribute is an attribute about the identity of a subscriber.

attribute bundle
A package of *attribute values* and *derived attribute values* from a *CSP*. The package has necessary cryptographic protection to allow *validation* of the bundle independent from interaction with the CSP or *IdP*. Attribute bundles are often used with subscriber-controlled wallets.

attribute provider
The provider of an *identity API* that provides access to a subscriber’s attributes without necessarily asserting that the subscriber is present to the *RP*.

attribute value
A complete statement that asserts an identity attribute of a subscriber, independent of format. For example, for the *attribute* “birthday,” a value could be “12/1/1980” or “December 1, 1980.”

audience restriction
The restriction of a message to a specific target audience to prevent a receiver from unknowingly *processing* a message intended for another recipient. In *federation protocols*, *assertions* are audience *restricted* to specific *RPs* to prevent an RP from accepting an assertion generated for a different RP.

authenticate
See *authentication*.

authenticated protected channel
An encrypted communication channel that uses *approved cryptography* where the connection initiator (client) has authenticated the recipient (server). Authenticated protected channels are encrypted to provide confidentiality and protection against active intermediaries and are frequently used in the user *authentication* process. *Transport Layer Security* (TLS) and Datagram Transport Layer Security (DTLS) [RFC9325] are examples of authenticated protected channels in which the certificate presented by the recipient is verified by the initiator. Unless otherwise specified, authenticated protected channels do not require the server to authenticate the client. Authentication

of the server is often accomplished through a certificate chain that leads to a trusted root rather than individually with each server.

authenticated session
See *protected session*.

authentication
The process by which a *claimant* proves possession and control of one or more *authenticators* bound to a *subscriber account* to demonstrate that they are the subscriber associated with that account.

Authentication Assurance Level (AAL)
A category describing the strength of the authentication process.

authenticator
Something that the subscriber possesses and controls (e.g., a *cryptographic module* or *password*) and that is used to *authenticate* a *claimant's* identity. See *authenticator type* and *multi-factor authenticator*.

authenticator binding
The establishment of an association between a specific *authenticator* and a *subscriber account* that allows the *authenticator* to be used to *authenticate* for that subscriber account, possibly in conjunction with other authenticators.

authorize
A decision to grant access, typically automated by evaluating a *subject*'s *attributes*.

authorized party
In *federation*, the organization, person, or entity that is responsible for making decisions regarding the release of information within the *federation transaction*, most notably subscriber *attributes*. This is often the subscriber (when runtime decisions are used) or the party operating the *IdP* (when *allowlists* are used).

back-channel communication
Communication between two systems that relies on a direct connection without using redirects through an intermediary such as a browser.

bearer assertion
An *assertion* that can be presented on its own as proof of the identity of the presenter.

blocklist
A documented list of specific elements that are blocked, per policy decision. This concept has historically been known as a *blacklist*.

challenge-response protocol
An *authentication protocol* in which the *verifier* sends the *claimant* a challenge (e.g., a random value or *nonce*) that the claimant combines with a secret (e.g., by hashing the challenge and a *shared secret* together or by applying a *private-key* operation to the challenge) to generate a response that is sent to the verifier. The verifier can independently verify the response generated by the claimant (e.g., by re-computing the hash of the challenge and the shared secret and comparing to the response or performing a public-key operation on the response) and establish that the claimant possesses and controls the secret.

core attributes
The set of identity *attributes* that the *CSP* has determined and documented to be required for *identity proofing*.

credential service provider (CSP)
A trusted entity whose functions include *identity proofing applicants* to the identity service and registering *authenticators* to *subscriber accounts*. A CSP may be an independent third party.

cross-site request forgery (CSRF)
An attack in which a subscriber who is currently *authenticated* to an *RP* and connected through a secure session browses an attacker’s website, causing the subscriber to unknowingly invoke unwanted actions at the RP.

For example, if a bank website is vulnerable to a CSRF attack, it may be possible for a subscriber to unintentionally *authorize* a large money transfer by clicking on a malicious link in an email while a connection to the bank is open in another browser window.

cross-site scripting (XSS)
A vulnerability that allows attackers to inject malicious code into an otherwise benign website. These scripts acquire the permissions of scripts generated by the target website to compromise the confidentiality and integrity of data transfers between the website and clients. Websites are vulnerable if they display user-supplied data from requests or forms without sanitizing the data so that it is not executable.

derived attribute value
A statement that asserts a limited identity *attribute* of a subscriber without containing the attribute value from which it is derived, independent of format. For example, instead of requesting the attribute “birthday,” a derived value could be “older than 18”. Instead of requesting the attribute for “physical address,” a derived value could be “currently residing in this district.” Previous versions of these guidelines referred to this construct as an “attribute reference.”

digital identity

An *attribute* or set of attributes that uniquely describes a *subject* within a given context.

digital signature

An *asymmetric key* operation in which the *private key* is used to digitally sign data and the *public key* is used to verify the signature. Digital signatures provide *authenticity* protection, integrity protection, and *non-repudiation* support but not confidentiality or *replay attack* protection.

disassociability

Enabling the *processing* of PII or events without association to individuals or devices beyond the operational requirements of the system. [NISTIR8062]

entropy

The amount of uncertainty that an attacker faces to determine the value of a secret. Entropy is usually stated in bits. A value with *n* bits of entropy has the same degree of uncertainty as a uniformly distributed *n*-bit random value.

equity

The consistent and systematic fair, just, and impartial treatment of all individuals, including individuals who belong to underserved communities that have been denied such treatment, such as Black, Latino, and Indigenous and Native American persons, Asian Americans and Pacific Islanders, and other persons of color; members of religious minorities; lesbian, gay, bisexual, transgender, and queer (LGBTQ+) persons; persons with disabilities; persons who live in rural areas; and persons otherwise adversely affected by persistent poverty or inequality. [EO13985]

Federal Information Processing Standard (FIPS)

Under the Information Technology Management Reform Act (Public Law 104-106), the Secretary of Commerce approves the standards and guidelines that the National Institute of Standards and Technology (NIST) develops for federal computer systems. NIST issues these standards and guidelines as Federal Information Processing Standards (FIPS) for government-wide use. NIST develops FIPS when there are compelling federal government requirements, such as for security and interoperability, and there are no acceptable industry standards or solutions. See background information for more details.

FIPS documents are available online on the FIPS home page: https://www.nist.gov/itl/fips.cfm

federated identifier

The combination of a *subject identifier* within an *assertion* and an *identifier* for the *IdP* that issued that assertion. When combined, these pieces of information uniquely identify the *subscriber* in the context of a *federation transaction*.

federation
A process that allows for the conveyance of identity and authentication information across a set of *networked* systems.

Federation Assurance Level (FAL)
A category that describes the process used in a *federation transaction* to communicate authentication events and subscriber *attributes* to an *RP*.

federation protocol
A technical protocol that is used in a *federation transaction* between *networked* systems.

federation proxy
A component that acts as a logical *RP* to a set of *IdPs* and a logical IdP to a set of RPs, bridging the two systems with a single component. These are sometimes referred to as “brokers.”

federation transaction
A specific instance of *processing* an authentication using a *federation* process for a specific *subscriber* by conveying an *assertion* from an *IdP* to an *RP*.

front-channel communication
Communication between two systems that relies on passing messages through an intermediary, such as using redirects through the subscriber’s browser.

hash function
A function that maps a bit string of arbitrary length to a fixed-length bit string. Approved hash functions satisfy the following properties:

1. One-way — It is computationally infeasible to find any input that maps to any pre-specified output.
2. Collision-resistant — It is computationally infeasible to find any two distinct inputs that map to the same output.

identifier
A data object that is associated with a single, unique entity (e.g., individual, device, or *session*) within a given context and is never assigned to any other entity within that context.

identity
See *digital identity*

identity API
A protected API accessed by an *RP* to access the *attributes* of a specific subscriber.

Identity Assurance Level (IAL)
A category that conveys the degree of confidence that the *subject*’s *claimed identity* is their real identity.

identity provider (IdP)
The party in a *federation transaction* that creates an *assertion* for the subscriber and transmits the assertion to the *RP*.

injection attack
An attack in which an attacker supplies untrusted input to a program. In the context of federation, the attacker presents an untrusted *assertion* or *assertion reference* to the *RP* in order to create an *authenticated session* with the RP.

login
Establishment of an *authenticated session* between a person and a system. Also known as *“sign in”*, *“log on”*, and *“sign on.”*

message authentication code (MAC)
A cryptographic checksum on data that uses a *symmetric key* to detect both accidental and intentional modifications of the data. MACs provide *authenticity* and integrity protection, but not *non-repudiation* protection.

network
An open communications medium, typically the Internet, used to transport messages between the *claimant* and other parties. Unless otherwise stated, no assumptions are made about the network’s security; it is assumed to be open and subject to active (e.g., impersonation, *session* hijacking) and passive (e.g., eavesdropping) attacks at any point between the parties (e.g., claimant, *verifier*, *CSP*, *RP*).

nonce
A value used in security protocols that is never repeated with the same key. For example, nonces used as challenges in *challenge-response authentication protocols* must not be repeated until authentication keys are changed. Otherwise, there is a possibility of a *replay attack*. Using a nonce as a challenge is a different requirement than a random challenge, because a nonce is not necessarily unpredictable.

pairwise pseudonymous identifier
A *pseudonymous identifier* generated by an IdP for use at a specific *RP*.

personal information
See *personally identifiable information*.

personally identifiable information (PII)
Information that can be used to distinguish or trace an individual's identity, either alone or when combined with other information that is linked or linkable to a specific individual. [A-130]

predictability
Enabling reliable assumptions by individuals, owners, and operators about PII and its *processing* by an information system. [NISTIR8062]

private key
In *asymmetric key* cryptography, the private key (i.e., a secret key) is a mathematical key used to create *digital signatures* and, depending on the algorithm, decrypt messages or files that are encrypted with the corresponding *public key*. In *symmetric key* cryptography, the same private key is used for both encryption and decryption.

processing
Operation or set of operations performed upon PII that can include, but is not limited to, the collection, retention, logging, generation, transformation, use, disclosure, transfer, and disposal of PII. [NISTIR8062]

protected session
A *session* in which messages between two participants are encrypted and integrity is protected using a set of *shared secrets* called "session keys."

A protected session is said to be *authenticated* if — during the session — one participant proves possession of one or more *authenticators* in addition to the session keys, and if the other party can verify the identity associated with the authenticators. If both participants are authenticated, the protected session is said to be *mutually authenticated*.

Provisioning API
A protected API that allows an *RP* to access identity *attributes* for multiple subscribers for the purposes of provisioning and managing RP *subscriber accounts*.

pseudonymous identifier
A meaningless but unique *identifier* that does not allow the *RP* to infer anything regarding the subscriber but that does permit the RP to associate multiple interactions with a single subscriber.

public key
The public part of an *asymmetric key* pair that is used to verify signatures or encrypt data.

public key certificate
A digital document issued and digitally signed by the *private key* of a certificate authority that binds an *identifier* to a subscriber's *public key*. The certificate indicates that the subscriber identified in the certificate has sole control of and access to the private key. See also [RFC5280].

public key infrastructure (PKI)
A set of policies, processes, server platforms, software, and workstations used to administer certificates and public-_private key_ pairs, including the ability to issue, maintain, and revoke *public key certificates*.

reauthentication
The process of confirming the subscriber's continued presence and intent to be *authenticated* during an extended usage *session*.

relying party (RP)
An entity that relies upon a *verifier*'s *assertion* of a subscriber's identity, typically to process a transaction or grant access to information or a system.

replay attack
An attack in which the attacker is able to replay previously captured messages (between a legitimate *claimant* and a *verifier*) to masquerade as that claimant to the verifier or vice versa.

risk assessment
The process of identifying, estimating, and prioritizing risks to organizational operations (i.e., mission, functions, image, or reputation), organizational assets, individuals, and other organizations that result from the operation of a system. A risk assessment is part of *risk management*, incorporates threat and vulnerability analyses, and considers mitigations provided by security *controls* that are planned or in-place. It is synonymous with "risk analysis."

risk management
The program and supporting processes that manage information security risk to organizational operations (including mission, functions, image, reputation), organizational assets, individuals, and other organizations and includes (i) establishing the context for risk-related activities, (ii) assessing risk, (iii) responding to risk once determined, and (iv) monitoring risk over time.

RP subscriber account
An account established and managed by the *RP* in a federated system based on the RP's view of the *subscriber account* from the *IdP*. An RP subscriber account is associated with one or more *federated identifiers* and allows the subscriber to access the account through a *federation transaction* with the IdP.

security domain
A set of systems under a common administrative and access control.

session
A persistent interaction between a subscriber and an *endpoint*, either an *RP* or a *CSP*. A session begins with an authentication event and ends with a session termination event. A session is bound by the use of a session secret that the subscriber's software (e.g., a browser, application, or OS) can present to the RP to prove association of the session with the authentication event.

session hijack attack
An attack in which the attacker is able to insert themselves between a *claimant* and a *verifier* subsequent to a successful authentication exchange between the latter two parties. The attacker is able to pose as a subscriber to the verifier or vice versa to control *session* data exchange. Sessions between the claimant and the *RP* can be similarly compromised.

single sign-on (SSO)
An authentication process by which one account and its *authenticators* are used to access multiple applications in a seamless manner, generally implemented with a *federation protocol*.

subject
A person, organization, device, hardware, *network*, software, or service. In these guidelines, a subject is a *natural person*.

subscriber
An individual enrolled in the *CSP* identity service.

subscriber account
An account established by the *CSP* containing information and *authenticators* registered for each subscriber enrolled in the CSP identity service.

symmetric key
A *cryptographic key* used to perform both the cryptographic operation and its inverse. (e.g., to encrypt and decrypt or create a *message authentication code* and to verify the code).

Transport Layer Security (TLS)

An authentication and security protocol widely implemented in browsers and web servers. TLS is defined by [RFC5246]. TLS is similar to the older SSL protocol, and TLS 1.0 is effectively SSL version 3.1. SP 800-52, Guidelines for the Selection and Use of Transport Layer Security (TLS) Implementations [SP800-52], specifies how TLS is to be used in government applications.

trust agreement

A set of conditions under which a *CSP*, *IdP*, and *RP* are allowed to participate in a *federation transaction* for the purposes of establishing an authentication *session* between the subscriber and the RP.

usability

The extent to which a product can be used by specified users to achieve specified goals with effectiveness, efficiency, and satisfaction in a specified context of use. [ISO/IEC9241-11]

verifier

An entity that verifies the *claimant's* identity by verifying the claimant's possession and control of one or more *authenticators* using an *authentication protocol*. To do this, the verifier needs to confirm the binding of the authenticators with the *subscriber account* and check that the subscriber account is active.

Appendix C. Changelog

This appendix is informative. It provides an overview of the changes to SP 800-63C since its initial release.

- Added discussion of equity considerations and requirements.
- Established trust agreements and registration/discovery (key establishment) as discrete steps in the federation process.
- All FALs have requirements around establishment of trust agreements and registration.
- FAL definitions no longer have encryption requirements; encryption is triggered by passing PII in an assertion through an untrusted party regardless of FAL.
- FAL2 requires injection protection.
- FAL3 allows more general bound authenticators including RP-managed authenticators, in addition to classical holder-of-key assertions.
- Communication of IAL/AAL/FAL required.
- Updated language to be more inclusive.
- Added definition and discussion of RP subscriber accounts.
- Added attribute provisioning models and discussion.
- Subscriber-controlled wallet model added, with specific requirements separated from general-purpose IdPs.
- Restructured core document sections to address common, general-purpose, and subscriber-controlled wallet requirements in separate sections.
- Redress requirements for IdPs and RPs added.
- Enterprise and dynamic use cases added throughout, with explicit examples.